TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY
Tampere International Center for Signal Processing
TICSP Series #41

Miika Ahdesmäki, Korbinian Strimmer, Nicole Radde, Jörg Rahnenführer, Konstantin Klemm, Harri Lähdesmäki & Olli Yli-Harja

Fifth International Workshop on Computational Systems Biology, WCSB 2008, June 11-13, 2008 Leipzig, Germany

Tampere 2008

Tampere International Center for Signal Processing.
TICSP series # 41

Miika Ahdesmäki, Korbinian Strimmer, Nicole Radde, Jörg Rahnenführer,
Konstantin Klemm, Harri Lähdesmäki & Olli Yli-Harja

Fifth International Workshop on Computational Systems Biology, WCSB
2008, June 11-13, 2008 Leipzig, Germany

Tampere International Center for Signal Processing
Tampere 2008

ISBN 978-952-15-1988-8
ISSN 1456-2774

PREFACE

The Workshop on Computational Systems Biology has been organized annually by the Computational Systems Biology research group in the Department of Signal Processing at Tampere University of Technology (TUT). The history of the workshop traces back to 2003, when it was organized for the first time as an internal meeting with some invited international collaborators. Since then the meeting has grown each year witnessing a rapid development in experimental biosciences and growth in the research of computational methods in systems biology.

This year the program committee set the target to making the event more international, as well as emphasizing the quality and significance of the research papers published in this proceedings book. Therefore in 2008 the workshop is organized in Leipzig, Institute for Medical Informatics, Statistics and Epidemiology (IMISE), Germany, together with collaborators from University of Leipzig and Dortmund University of Technology. The joint organization has proved to be smooth and successful - we are having a record number of international participants. We have brought together the various communities involved in the different aspects of computational systems biology research, e.g. experimental biology, machine learning, signal processing, statistics and theoretical physics. The workshop program together with the range of published papers demonstrate an increasing sphere of influence of WCSB.

This volume is the collection of the research papers and short abstracts submitted to WCSB2008. We would like to thank the authors and the reviewers for their contribution to this workshop. We are also grateful for the contribution of organizers in Finland and Germany for their efforts. We would also like to thank Finnish Academy of Sciences, Tampere Graduate School in Information Science and Engineering (TISE), Tampere International Center for Signal Processing (TICSP), Research Training Group "Statistical Modelling", Department of Statistics, Dortmund University of Technology, and Max Planck Institute for Evolutionary Anthropology for their support.

On behalf of the WCSB 2008 Scientific committee,

Olli Yli-Harja

WCSB 2008 ORGANISATION

Scientific Committee

Jaakko Astola (TUT)
Marja-Leena Linne (TUT)
Harri Lähdesmäki (TUT)
Ioan Tabus (TUT)
Olli Yli-Harja (TUT, workshop chair)
Hans Binder (UL)
Konstantin Klemm (UL)
Nicole Radde (UL)
Korbinian Strimmer (UL)
Samuel Kaski (Helsinki University of Technology)
Jörg Rahnenführer (Technische Universität Dortmund)
Ilya Shmulevich (Institute for Systems Biology)

Workshop Organisation

Miika Ahdesmäki (TUT)
Xiaofeng Dai (TUT)
Virve Larmila (TUT)
Pirkko Ruotsalainen (TUT)
Pekka Ruusuvuori (TUT)

TABLE OF CONTENTS

Abstracts

Regular papers

ABSTRACTS

Qualitative Modeling and Simulation of Bacterial Regulatory Networks

Hidde de Jong

INRIA Grenoble-Rhône-Alpes, France
Hidde.de-Jong@inrialpes.fr

The adaptation of living organisms to their environment is controlled at the molecular level by large and complex networks of genes, mRNAs, proteins, metabolites, and their mutual interactions. We have analyzed the network of global transcription regulators controlling the adaptation of the bacterium Escherichia coli to environmental stress conditions. Even though E. coli is one of the best studied model organisms, it is currently little understood how a stress signal is sensed and propagated through the network of global regulators, and leads the cell to respond in an adequateway. We have modeled the carbon starvation network of E. coli and applied various model reduction methods in order to overcome the current lack of quantitative data on kinetic parameters. The qualitative dynamics of the resulting simplified piecewise-affine differential equation model can be studied using discrete abstraction approaches from hybrid systems theory. This has allowed us to identify essential features of the transition between exponential and stationary phase of the bacteria and to make new predictions on the qualitative system behavior following a carbon upshift.

EVOLUTION OF MOLECULAR NETWORKS

Martin Lercher

Heinrich-Heine-Universität, Düsseldorf, Germany
lercher@cs.uni-duesseldorf.de

How do complex molecular systems evolve? Which types of genes are gained and lost in response to environmental changes? How are such new genes integrated into the regulatory circuits of an organism? Horizontal gene transfer among bacteria provides a rich data source for the study of these phenomena. Genes gained by (or lost from) metabolic networks act mostly at the cell´s interface to the environment, and are generally environment specific. Constraining evolution to gene losses, as has happened in endosymbionts, even allows the model-based prediction of evolutionary outcomes. While newly added components need to be active immediately to provide selective advantages, their regulatory fine-tuning proceeds surprisingly slowly, often spanning millions of years.

UNCOVERING STRUCTURE AND MOTIFS IN BIOLOGICAL NETWORKS

Stéphane Robin

UMR 518 AgroParisTech / INRA Appl. Math. Comput. Sc., France
Stephane.Robin@agroparistech.fr

Getting and analysing biological interaction networks is at the core of systems biology. To help understanding these complex networks, many recent works have suggested to focus (i) the global topology of the network and (ii) on motifs which occur more frequently than expected in random.

Looking for a latent structure is one of the many strategies used to better understand the behaviour of a network. Several methods already exist for the binary case. We present a model-based strategy to uncover groups of nodes in both binary and valued graphs. This framework can be used for a wide span of parametric random graphs models. Variational tools allow us to achieve approximate maximum likelihood estimation of the parameters of these models. We provide several examples, showing that the proposed methodology can be applied to a broad range of biological networks.

To identify exceptional motifs in a given network, we propose a statistical and analytical method which does not require any simulation. For this, we first provide an analytical expression of the mean and variance of the count under any stationary random graph model. Then we approximate the motif count distribution by a compound Poisson distribution whose parameters are derived from the mean and variance of the count. Thanks to simulations, we show that the quality of our compound Poisson approximation is very good and highly better than a Gaussian or a Poisson one. The compound Poisson distribution can then be used to get an approximate p-value and to decide if an observed count is significantly high or not.

We compare our method to the Mfinder software on PPI data and discuss the choice of a relevant random graph model to detect over-represented motifs.

Joint work with J.-J. Daudin, M. Koskas, M. Mariadassou, F. Picard and S. Schbath.

BAYESIAN INFERENCE FOR STOCHASTIC MODELS OF INTRACELLULAR REACTION NETWORKS

Darren Wilkinson

School of Mathematics and Statistics, Newcastle University, UK
d.j.wilkinson@ncl.ac.uk

This talk will provide an overview of computationally intensive methods for conducting Bayesian inference for the rate constants of stochastic kinetic models of reaction networks using single-cell time course data. Inference for the true Markov jump process is extremely challenging in realistic scenarios, so it is sometimes useful to replace the "true" model with a diffusion approximation, known in this context as the Chemical Langevin Equation (CLE). Inference for the CLE is also challenging, but the development of effective algorithms is possible, and turns out to be extremely effective, even in scenarios where one would expect the diffusion approximation to break down.

REGULAR PAPERS

COMPUTATIONAL TOOL FOR STRAIN DESIGN: MAXIMIZING YIELDS IN METABOLIC SYSTEMS

Tommi Aho[1], Roger Mallol Parera[1,2*], Antti Larjo[1], Olli Yli-Harja[1]*

[1]Institute of Signal Processing, Tampere University of Technology,
P.O. Box 553, FI-33101 Tampere, Finland
[2]Escola Tècnica Superior d'Enginyeria,
Edifici Q, Campus de la UAB, 08193 Bellaterra (Cerdanyola del Vallès), Spain
tommi.aho@tut.fi, roger.mallol@campus.uab.cat, antti.larjo@tut.fi,
olli.yli-harja@tut.fi
[*]Authors with equal contribution

ABSTRACT

In bioprocesses, bacteria can be used as cell factories for producing small molecules. In order to maximize the yield of the product of interest, cultivation conditions can be optimized and bacteria can be genetically modified. Metabolic capabilities of a bacterium can be computationally approximated using a reconstruction of its metabolic network. Here, we present a computational tool called Bioptima which predicts metabolic modifications needed to improve the yield of a user-defined compound. The tool uses optimization algorithms that inactivate reactions in the given metabolic reconstruction and, on the other hand, insert new reactions from the KEGG metabolic database. The metabolic yields are calculated using flux balance analysis implemented in COBRAToolbox, and the KEGG database is accessed using Medicel Integrator platform.

1. INTRODUCTION

During evolution, micro-organisms have specialized to grow in their specific environment. They maintain metabolism that produces specific metabolites which they need for growth and survival. The ability of a cell to produce various substances can be exploited in bioprocesses which aim at producing a specific compound for the needs within medicine, bioenergy production, etc. Because the cell adjusts its metabolic fluxes on the basis of its own needs, it may not produce the desired compound as much as it could. Therefore, the metabolic fluxes may need to be redirected in order to maximize the yield of the compound of interest.

Metabolic processes of a cell may be controlled to a certain degree using various factors during cultivation, such as pH, nutrients, oxygen, and loading rate of a bioreactor. In addition to these environmental factors, genetic modifications can be used to direct metabolic fluxes towards the desired pathways. New metabolic routes can be introduced by addition of genes which products are metabolic enzymes. On the other hand, gene knockouts can block the production of the respective metabolic enzymes and, therefore, result to inactivation of metabolic pathways.

A central issue in genetic modifications is their targeting. One choice is not to address this question at all. In that case, genetic mutations are produced randomly, and the best mutations are chosen by a screening method. Another choice is to design modifications rationally. The advances in high-throughput genome sequencing techniques, efficient database-based annotation tools, and the wealth of biochemical and cell physiological literature have made it possible to produce genome-scale metabolic reconstructions for several micro-organisms (see, e.g., [1, 2, 3, 4, 5]). These reconstructions can be used in system-level modeling approaches to facilitate strain design where the effects of genetic modifications are predicted *in silico*.

This work presents a computational tool for rational strain design. The approach has similarities with [6] and [7] since it includes a bi-level optimization strategy, and it uses a large reaction repository. The main differences are in the optimization methods, in the use of the computational platform, and in the data import methods. The Bioptima tool is provided for public use under GNU GPL license. It can be downloaded from http://www.cs.tut.fi/sgn/csb/.

2. MATERIALS AND METHODS

Figure 1 illustrates the main steps of the computational strain design procedure: import of a metabolic model, modifying the model using a reaction repository, and two optimization tasks. These steps are discussed next.

2.1. Metabolic network model

Systems biology markup language (SBML) [8] is a widely used exchange format for biochemical network

models. A network model is imported to the tool as an SBML file that contains lower and upper bounds for reaction rates together with structural information of the network.

2.2. Additional metabolic reactions

During the optimization procedure, the tool is able to add such metabolic reactions to the model which do not exist there originally. These reactions are chosen from a reaction repository which is built using Medicel Integrator platform [9]. The repository contains all those metabolic reactions from Kyoto Encyclopedia of Genes and Genomes (KEGG) [10] which conserve matter. The total number of different biochemical transformations in the repository is about 5000.

2.3. Bi-level optimization strategy

The optimization contains two nested optimization problems, as presented in Figure 1.

The outer optimization introduces additions and deletions of metabolic reactions into the model, checks the production of a user-defined compound, and aims to improve its yield. The first option to resolve this optimization problem is a greedy algorithm. The greedy algorithm tests different reaction additions and deletions, and accepts only those modifications which improve the value of the objective function (i.e., the production of the compound of interest). The second option to resolve the outer optimization problem is a simulated annealing algorithm [11]. In this case, the algorithm tests additions and deletions, but their acceptance is based on a probability function. In the beginning, the algorithm may accept such modifications which worsen the value of the objective function. During its execution, the probability to accept these modifications gets gradually smaller and, finally, the algorithm becomes greedy. Both optimization methods are iterative. Their execution is stopped if the maximum number of iterations is obtained, the maximum number of modifications is implemented, or if the theoretical maximum yield for the compound of interest is obtained.

The inner optimization procedure predicts the response of a cell to the modifications in its metabolic network. A common assumption in metabolic modeling is that micro-organisms aim at maximizing their biomass production and, when the metabolic model is accurate enough, the respective metabolic flux distribution can be estimated using flux balance analysis (FBA) [1]. In FBA, the fluxes through reactions are resolved with the help of linear programming optimization. An optimization task is set up using the knowledge of the structure of the metabolic network, stoichiometric coefficients of reactions, lower and upper bounds of reaction rates, and a steady-state assumption for compounds which cannot be freely exchanged with the extracellular environment. Bioptima uses COBRAToolbox [12] which implements the needed functionality related to FBA. The optimization procedures of Bioptima are implemented using Matlab programming environment [13].

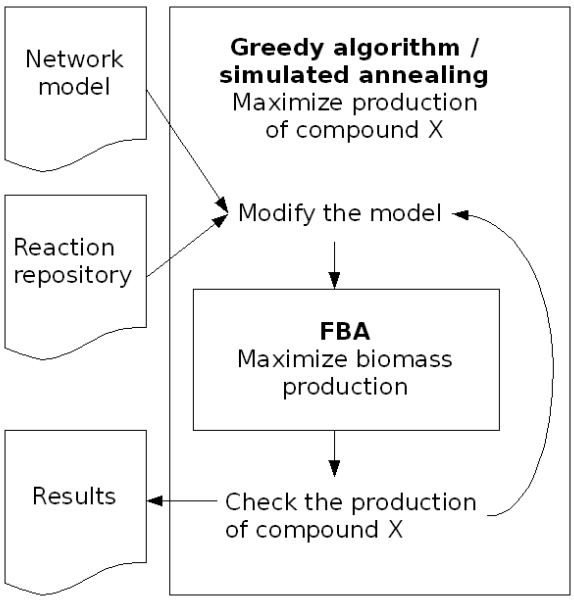


Figure 1: Overview to the optimization procedure. Compound X is a user-defined compound which yield is to be maximized.

3. RESULTS

As a usage example of the tool, we searched for such reaction deletions and additions which predict improved hydrogen production for *Escherichia coli*.

The model of the metabolic network of *E. coli* was obtained from [4]. The rate constraints of the exchange reactions of the model were set to simulate anaerobic cultivation conditions. In order to make hydrogen production possible by the model, the rate constraints of one reaction had to be loosened. Maximization of hydrogen secretion was set as the objective of the outer optimization procedure, and maximization of biomass production was set as the cellular objective in the inner optimization.

Simulated annealing algorithm was set to perform at most 2000 iterations, to make at most 50 modifications to the network, and to keep biomass production rate always greater than 0.1 1/h. After 26 implemented deletions and 24 implemented additions the initial hydrogen production rate 14.1 mmol/gDW/h was improved to the rate 17.2 mmol/gDW/h. Because of the nature of the simulated annealing algorithm, all the made modifications did not improve the value of the objective function. After the algorithm stopped, these modifications were

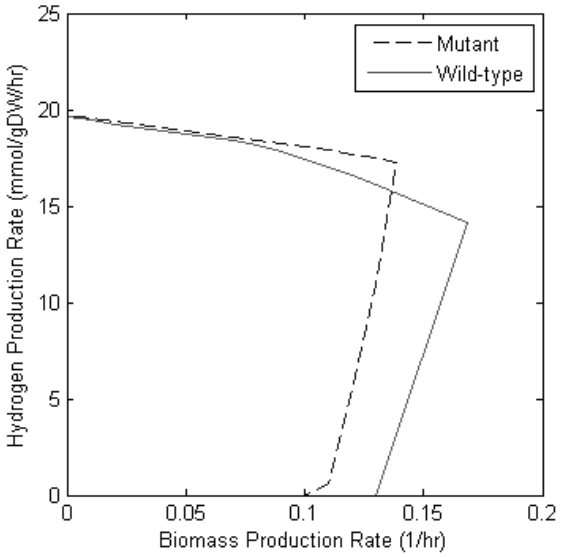


Figure 2. Hydrogen production limits for wild-type and mutant strains of *E. coli* under anaerobic conditions. In both cases, a line limits an area that describes the possibilities of a cell to produce hydrogen while maintaining different growing rates. The upper right corners are the points where the wild-type and mutant cells meet their growth objectives.

searched and removed from the list of modifications. The final mutant network, containing three additions and three deletions, had hydrogen production rate of 17.3 mmol/gDW/h. Figure 2 illustrates the characteristics of the wild-type *E. coli* metabolic network and the mutated network. The mutant is unable to grow as fast as the wild-type, but it produces more hydrogen when it maximizes its biomass production. This indicates that the mutations were able to redirect the metabolic fluxes to the desired direction where the cell overproduces hydrogen while pursuing its own objective.

4. CONCLUSION

The number of metabolic reconstructions increases along with the advances in molecular and cellular biology and information technology infrastructure. In addition, the sizes of the reconstructions grow, and they become more detailed. The reconstructions can be used to facilitate rational strain design where microbes are used as cell factories, and their cultivation conditions and metabolic capabilities are optimized for this purpose.

In this work a computational tool Bioptima was developed. Given a metabolic reconstruction, the tool suggests modifications which are needed to improve the yield of a user-defined compound. The use of the tool was demonstrated using hydrogen production in *E. coli* as an example.

5. ACKNOWLEDGMENTS

This work was supported by the Academy of Finland, (application number 213462, Finnish Programme for Centres of Excellence in Research 2006-2011).

6. REFERENCES

[1] J. Edwards, B. Palsson, "Systems properties of the *Haemophilus influenzae* Rd metabolic genotype," *J. Biol. Chem.*, vol. 274, pp. 17410-17416, 1999.

[2] N. Duarte, M. Herrgård, and B. Palsson, "Reconstruction and validation of *Saccharomyces cerevisiae* iND750, a fully compartmentalized genome-scale metabolic model," *Gen. Res.*, vol. 14, pp. 1298-1309, 2004.

[3] M. Herrgård, B.-S. Lee, V. Portnoy, and B. Palsson, "Integrated analysis of regulatory and metabolic networks reveals novel regulatory mechanisms in *Saccharomyces cerevisiae*," *Gen. Res.*, vol. 16, pp. 627-635, 2006.

[4] A. Feist, C. Henry, J. Reed, M. Krummenacker, A. Joyce, P. Karp, L. Broadbelt, V. Hatzimanikatis, and B. Palsson, "A genome-scale metabolic reconstruction for *Escherichia coli* K-12 MG1655 that accounts for 1260 ORFs and thermodynamic information," *Mol. Syst. Biol.*, 3:121, 2007.

[5] I. Borodina, P. Krabben, and Jens Nielsen, "Genome-scale analysis of *Streptomyces coelicolor* A3(2) metabolism," *Gen. Res.*, vol. 15, pp. 820-829, 2005.

[6] A. Burgard, P. Pharkya, and Costas D. Maranas, "OptKnock: A bilevel programming framework for identifying gene knockout strategies for microbial strain optimization," *Biotech. Bioeng.*, vol. 84, pp. 647-657, 2003.

[7] P. Pharkya, A. Burgard, and C. Maranas, "OptStrain: a computational framework for redesign of microbial production systems," *Gen. Res.*, vol. 14, pp. 2367-2376, 2004.

[8] M. Hucka, A. Finney, H. Sauro, H. Bolouri, J. Doyle, H. Kitano, A. Arkin, B. Bornstein, and the rest of SBML community, "The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models," *Bioinformatics*, vol. 19, pp. 524-531, 2003.

[9] Medicel Ltd., http://www.medicel.com.

[10] M. Kanehisa, S. Goto, S. Kawashima, and A. Nakaya, "The KEGG databases at GenomeNet," *Nucl. Acid. Res.*, vol. 30, pp. 42-46, 2002.

[11] O. Nelles, *Non-linear system identification: from classical approaches to neural networks and fuzzy models*, Springer, 2001.

[12] S. Becker, A. Feist, M. Mo, G. Hannum, B. Palsson, and M. Herrgard, "Quantitative prediction of cellular metabolism with constraint-based models: the COBRA Toolbox," *Nat. Protoc.*, vol. 2, pp. 727-738, 2007.

[13] The Mathworks Inc., http://www.mathworks.com.

AN EXPERT-BASED APPROACH FOR THE IDENTIFICATION OF REMOTE HOMOLOGS

Nicolas Beaume[1,2]*, Gerard Ramstein*[2] *and Yannick Jacques*[1]

[1]INSERM, U893, groupe de recherche cytokines et recepteurs, Nantes, France,
[2]LINA, KOD team, EPUN,
rue Christian Pauc, 44306 Nantes, France
nicolas.beaume@univ-nantes.fr, gerard.ramstein@univ-nantes.fr, yannick.jacques@univ-nantes.fr

ABSTRACT

As remote homology detection remains a challenging task, even for powerful methods like Support Vector Machines (SVMs), we suggest adding biological information to improve classification performances. Contrary to SVMs involving large feature spaces, an expert focuses only on a specific clue. Following a Bayesian approach, an expert associates to each candidate a membership probability estimate that can be combined to an SVM classifier. This hybrid machine learning system is composed of an SVM classifier and experts has been tested on a particular family, namely the cytokines. Our use of four SVM methods and four different experts demonstrated that this technique greatly improves the classification performances.

1. INTRODUCTION

Homologs are proteins which arise from a common ancestor. This definition leads to the notion of protein family where each member is homolog to the others. The research of all its members is a primary step to understand a protein family, and many methods of homolog detection have been developed to this purpose. Common strategies include similarity-search [1], structure-based alignments [2] or supervised classification methods [3]. Among them, machine learning approaches and especially the Support Vector Machine (SVM) [4] outperform the others. Several SVM classifiers , [5], , [6] were developed, but despite their efficiency, the research of homologs remains a challenge for some families. An example of such a family is the four-helix bundle cytokines (refered to as the four-helix cytokines in the following). These proteins are involved in intercellular signal transmission and are particulary known for their roles in inflammation and immune response.

To solve this problem, we propose a new strategy of homolog detection, based on an SVM classifier combined with the specific knowledge of the family studied. This approach associates the efficiency of a generic classifier with the specificity of particular features selected by the biologist to characterize the family.

In this paper, we will first describe how to design experts based on specific criteria and show four examples of experts for the four-helix cytokines. Then, we will demonstrate that the association of these experts with an SVM classifier taken from the literature can improve the identification capacity of the latter.

2. EXPERTS

2.1. What is an expert ?

An expert can be defined as a tool which adds a biological knowledge to a classification system. An expert does not need to have its own discriminating power, but consistute an auxiliary support to impove the classification performances. Multiple experts can be designed for the same problem, and combined to provide more information to the system.

Practically, an expert uses a criterion c to estimate the probability p that a candidate belongs to the investigated family. Let F represent the classification variable. In our case, there are only two classes: f (the positive class, i.e. the four-helix cytokines) or $\neg f$ (the negative class). A Naive Bayes classifier is a function that assigns a class label to an example e. According to Bayes'rule, the probability of an example e being class f is :

$$p(f|e) = \frac{p(e|f)p(f)}{p(e)} \quad (1)$$

e is classified as the class $F = f$ if and only if

$$B(e) = \frac{p(F = f|e)}{p(F = \neg f|e)} \geq 1$$

In our approach, we only use the probability estimate given by Eq.1, since the expert is designed to be associated with another classifier and different experts.

2.2. Exemple of experts

To illustrate this concept, four automated experts where designed for the four-helix cytokines using the criteria described below.

2.2.1. Length of sequence

Most of the four-helix cytokines have a length between 76 and 232 amino acids, which is a narrow range in comparison with that of the human genome proteins. Thus,

length-based expert (L-expert) can partially discriminate candidates, especially those with an irrelevant length.

2.2.2. Molecular weight

The molecular weight of a substance is defined as the mass of one molecule of that substance, relative to the unified atomic mass unit u (equal to $1/12^{th}$ the mass of one atom of C^{12}). This feature is commonly used by biologists to characterize the size of a protein and is easy to compute from the protein sequence, using tools like the BIOJAVA API. [7]. Four-helix cytokines often use the same cellular components, suggesting they share physico-chemical features to interact with them. Following this idea, a molecular weight expert (MW-expert) can underline candidates having comparable features with the four-helix cytokines.

2.2.3. Isoelectrical point

The isoelectrical point is the pH where a given molecule is in a zwitterionic state, *i.e.* it carries no net electrical charge. Conveniently, isoelectrical point can be computed from the protein sequence itself using the BIOJAVA API. Like with the molecular weight, an isoelectrical point expert (IP-expert) may be used to highlight candidates displaying physico-chemical features common with the four-helix cytokine.

2.2.4. Secondary structure

More than the sequence, the protein structure (or tertiary structure) is a well conserved feature in protein families. Due to the low number of tertiary structures resolved and the difficulty to predict them, it is impossible to directly use this criterion to evaluate candidates. The secondary structure defined as the sequencial chaining of local structures of amino acids is, in the case of the four-helix cytokines, as well conserved and far more computationally tractable than the tertiary one. The secondary structure can be predicted from the protein sequence by using software like PSIPRED [8]. To compare the secondary structure of a candidate with that of four-helix cytokines ones, we propose to use the SOV criterion [9]. The SOV computes a similarity score between two structures. This score ranges from 0 to 1. A score of 0 means that the structures have no residue in the same local structure state and a score of 1 means that the two structures are identical. Thus the secondary structure expert (SS-expert) can reveal candidates sharing structural similarities with the four-helix cytokines, which is a strong evidence that they may belong to that family.

3. EXPERT EVALUATION AND DISCUSSION

To demonstrate the interest of the experts, we have analyzed the combination of the four experts described above with an SVM classifier for the detection of four-helix cytokines.

3.1. SVM classifier

Our approach involves a classifier associated with experts. As noted above, SVM classifiers [4] are the most efficient classifiers for homolog detection.

SVMs operate a mapping of a training set into a high dimensional feature space. A hyperplane boundary is defined between the positive and the negative classes so that the separation plane maximizes a margin from any point of the training set. An unlabelled point can then be predicted by simply considering the space region where it lies. The SVM technique has been applied with the LIBSVM API [10] allowing to compute a membership probability instead of a simple boolean value.

As SVM requires that the input be fixed-length numeric vectors, a feature extraction technique is generally used to transcribe the variable-length strings representing the sequences into real vectors. Several SVM classifiers were developed especially for biomedical data. They differ only by their feature extraction technique.

The SVM classifier, we used, called LA kernel[6], is known to be very efficient at detecting remote homologs. The classifier compares a sequence s with a collection of known proteins. This learning set is composed of k positive and negative examples. The sequence is transformed into a numeric vector of length k for which the i^{th} component represents a similarity measure between s and the i^{th} sequence of the learning set. This value is given by the Smith-Waterman algorithm (SW), which is a widely-used local alignment method. Yet, instead of considering only the best local alignment, LA kernel performs a summation over all the possible ones. The classifier is trained by considering each sequence of the training set like an unknown sequence and using the $k-1$ other to compute the SW-score. The highest score obtained this way is also used as the k^{th} score to avoid bias due to alignment of a sequence against itself.

3.1.1. Methods

To combined LA kernel with the experts, we used the unweighted arithmetic mean. Although this choice of agregation operator seems naive, it can be justified by the fact that this operator gives each of its components the same weight, allowing us to equally appreciate the contribution of each expert. Combination of the expert membership probabilities into the SVM classifier together with the sequence-based features was tested. This aggregation method gives classification results similar to LA kernel alone, probably because of a low influence of the experts in the choice of the support vectors due to their weak discriminating power.

To classify the sequence, we have ranked them according to their average membership probability over LA kernel (see 3.1) and the experts used in the agregation. As a measure of performance, we used the Area Under ROC Curve (AUC)[11]. An example of ROC curve is given in fig.1 The AUC is defined as:

$$AUC = \frac{\Sigma_{i=1}^{n^+}\Sigma_{j=1}^{n^-}\mathbf{1}_{g(x_i^+)>g(x_j^-)}}{n^+n^-} \qquad (2)$$

where $g(.)$ is the scoring function used for the ranking (in our case, the natural ordering), x^+ (resp. x^-) represents a positive (resp. negative) example, n^+ (resp. n^-) their

number, and 1_π is the indicator function (equals to 1 if the predicate π holds and 0 otherwise). This indice returns a score between 0 and 1 which can be interpreted as the probability that a positive example (*i.e.* a cytokine) will achieve a higher score than a negative one (*i.e.* a negative example), when both examples are selected at random. An AUC of 1 means that all positives are ranked before the negatives, an AUC of 0 means the contrary and an AUC of 0.5 means that the positives and the negatives are randomly ranked.
This approach is more interesting than assigning a label to each sequence and calculating the accuracy of classification because we are directly working on rank without having to define an arbitrary threshold.

3.1.2. Dataset

We have used a dataset of 30 four-helix cytokines and 6493 negative examples from the data base SCOP [12]. This dataset was split into two subsets : a learning set and an evaluation set. Experiments demonstrate that classification accuracy saturates when using more than 100 negative examples in the learning set (data not shown). Thus, we have designed learning sets with 15 four-helix cytokines and 100 negative examples, randomly drawn from the dataset. From the remaining data (15 four-helix cytokine and 6393 negative examples), we kept the four-helix cytokine and the 200 negative examples with the highest membership probability obtained from LA kernel. This selection was performed in order to keep only the highest-ranked candidates, as would do a human expert. These 215 candidates consitute the evaluation set. This operation was repeated 100 times to obtain 100 learning set/evaluation set couples. LA kernel and the experts were trained on learning sets and are used to classify the associated evaluation sets.
For each candidate, the unweighted mean of the membership probabilities given by LA kernel and the four experts was taken and used to rank the candidates, then the AUC was computed to assess the quality of the final ranking.

3.2. Results

Table 1 summarizes results from 100 different data sets. The ΔAUC indicates the gain of performance achieved by using experts. The gain is the difference of AUC between LA kernel alone and the agregation of LA kernel and various combinations of experts.

The average AUC of LA kernel alone over the 100 trials is 0.695 with a standard deviation of 0.02.

These results show that all combination of experts, except one, have a positive ΔAUC, indicating that the associations of LA kernel with experts outperforms LA kernel alone. Furthermore, it must be noted that, in general, the more experts we add, the more the classification performance increases, suggesting that all experts have a positive influence on the classification accuracy.
In fact, all combinations of experts are not equivalent. The IP-expert seems less efficient than the others as demonstrate the negative ΔAUC when associated with LA kernel alone. Likewise, associations containing the IP-experts have the lowest ΔAUC among the associations with the same number of experts. For example, when considering the association of two experts with LA kernel, the associations LA+L+IP, LA+SS+IP and LA+IP+MW have the lowest ΔAUC. The same observation can be made for the association of three experts with LA kernel. This strongly suggests that the isoelectrical point adds less information than the other experts to the classification. Yet, in the four-expert combination, ΔAUC is higher than in every three-experts ones, suggesting that the IP-expert adds a useful information to the classification.
Combinations involving either L-expert or the MW-expert show comparable ΔAUC. However, the L-expert seems to slightly outperform the MW-expert in the three-expert association (0.146 for the LA+SS+PI+MW association vs 0.156 for the LA+L+SS+IP one). Like with the IP-expert, the comparison between the four-expert association and every three-expert associations shows that the combination of the two is required to achieve the highest ΔAUC. Finally these results demonstrate that the SS-expert adds the most useful information, as all combination containing it have the highest ΔAUC among the combinations with the same number of experts.

Figure 1 shows a representative example of ROC curve of LA kernel alone (dashed line) and LA kernel combined with the four experts (plain line). One can note a significant increase of the ROC curve. This is especially true for the results of fig.1 : one observes that LA kernel has placed a set of negative candidates at the middle of the ranking (at true positive rate 0.6 on the curve). The agregation of experts clearly correct this misranking.

L	SS	IP	MW	ΔAUC
⋆				0.043
	⋆			0.068
		⋆		-0.019
			⋆	0.05
⋆	⋆			0.132
⋆		⋆		0.067
⋆			⋆	0.107
	⋆	⋆		0.092
	⋆		⋆	0.134
		⋆	⋆	0.062
⋆	⋆	⋆		0.156
⋆	⋆		⋆	0.169
⋆		⋆	⋆	0.117
	⋆	⋆	⋆	0.146
⋆	⋆	⋆	⋆	0.174

Table 1. ΔAUC obtained by the association of LA kernel with different combination of experts LA = LA kernel, L = length (L-) expert, SS = Secondary Structure (SS-) expert, IP = Isolectrical Point (IP-) expert, MW = Molecular Weight (MW-) expert

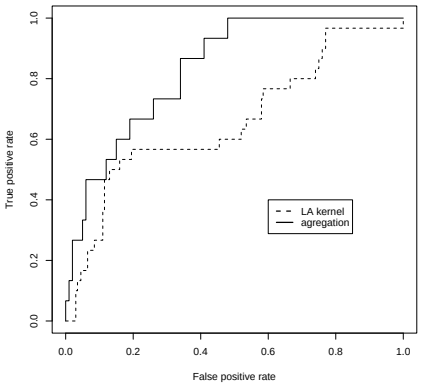


Figure 1. An example of ROC curves. A binary classifier defines a membership estimate and uses a decision value to separate two classes. This cut point determines the individuals considered as positives. Among these, actual positives are called true positives (TPs) whereas actual negatives are called false positives (FPs). Each point of the ROC curve depicts a cut point which determines the rate of TPs and FPs. The closer the curve follows the left-hand border and then the top border of the ROC space, the more accurate the classifier is.

3.3. Discussion

In this article, we have proposed a new strategy of homolog detection, based on a standard classifier of the literature, combined with information specific to the family studied. We have explained how this information can be exploited through experts and shown four examples of experts in the case of the four-helix cytokines. We have tested this strategy by combining the experts previously discribed with LA kernel, an SVM classifier known to be one of the best for homolog detection. Our results allow us to draw several conclusions.
First, the experiments clearly demonstrate the validity of our approach, as almost every combination of experts with LA kernel achieve better performances than LA kernel alone. The results also point out that some experts are more interesting than others. This is especially true for the SS-expert, which is not surprising as secondary structure is a well conserved feature among the four-helix cytokines. However, all the experts bring valuable information and thus are needed to obtain the best performances.
Although this work was dedicated to the four-helix cytokines, the strategy of combining classifiers with specific experts, can be generalized to any homolog detection problem.

Perspectives of this work are two-fold. First, the possibility of adding other experts and classifiers can be investigated, in order to determine the optimal association that maximize the classification performances. Secondly, our agregation strategy is very simple. The combination of classifiers has long been proposed as a method to improve the accuracy achieved in isolation by a single classifer. An evaluation of other agregation methods can be conducted to suggest different means of combining experts and classifiers. A comparison of different approaches, based on metaheuristics, fuzzy techniques and meta-classification are currently investigated.

4. REFERENCES

[1] S. F. Altschul, T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman, "Gapped blast and psi-blast: a new generation of protein database search programs," *Nucleic Acids Res*, vol. 25, no. 17, pp. 3389–3402, Sep 1997.

[2] S. A. Cammer, B. T. Hoffman, J. A. Speir, M. A. Canady, M. R. Nelson, S. Knutson, M. Gallina, S. M. Baxter, and J. S. Fetrow, "Structure-based active site profiles for genome analysis and functional family subclassification," *J Mol Biol*, vol. 334, no. 3, pp. 387–401, Nov 2003.

[3] C. Leslie, E. Eskin, and W. S. Noble, "The spectrum kernel: a string kernel for svm protein classification," *Pac Symp Biocomput*, pp. 564–575, 2002.

[4] V. Vapnik, *The nature of statistical learning theory*, Springer-Verlag, 1998.

[5] C. S. Leslie, E. Eskin, A. Cohen, J. Weston, and W. S. Noble, "Mismatch string kernels for discriminative protein classification," *Bioinformatics*, vol. 20, no. 4, pp. 467–476, Mar 2004.

[6] H. Saigo, J. P. Vert, N. Ueda, and T. Akutsu, "Protein homology detection using string alignment kernels," *Bioinformatics*, vol. 20, no. 11, pp. 1682–1689, Jul 2004.

[7] M. Pocock, T. Down, and T. Hubbard, "Biojava: open source components for bioinformatics," *SIGBIO Newsl.*, vol. 20, no. 2, pp. 10–12, 2000.

[8] D. T. Jones, "Protein secondary structure prediction based on position-specific scoring matrices," *J Mol Biol*, vol. 292, no. 2, pp. 195–202, Sep 1999.

[9] A. Zemla, C. Venclovas, K. Fidelis, and B. Rost, "A modified definition of sov, a segment-based measure for protein secondary structure prediction assessment," *Proteins*, vol. 34, no. 2, pp. 220–223, Feb 1999.

[10] H. Chih-Wei, C. Chih-Chung, and L. Chih-Sen, "A practical guide to support vector classification," http://www.csie.ntu.edu.tw/ cjlin/libsvm/.

[11] M. H. Zweig and G. Campbell, "Receiver-operating characteristic (roc) plots: a fundamental evaluation tool in clinical medicine," *Clin Chem*, vol. 39, no. 4, pp. 561–577, Apr 1993.

[12] L. L. Conte, B. Ailey, T. J. Hubbard, S. E. Brenner, A. G. Murzin, and C. Chothia, "Scop: a structural classification of proteins database," *Nucleic Acids Res*, vol. 28, no. 1, pp. 257–259, Jan 2000.

SLOPPY PARAMETERS IN OSCILLATORY SYSTEMS WITH UNOBSERVED SPECIES

Ben Calderhead and Mark Girolami

Department of Computing Science,
University of Glasgow, Scotland
bc@dcs.gla.ac.uk, girolami@dcs.gla.ac.uk

ABSTRACT

Nonlinear models based on systems of ordinary differential equations have been successfully employed to describe the behaviour of a wide range of complex biological processes. The main challenge when using such models lies in finding values for the free parameters which allow the model to reproduce the observed behaviour, and it has recently been shown [1] that often there are multiple sets of parameters which fulfil this requirement. Free parameters which admit a wide range of plausible values are termed to be "sloppy". In this paper we adopt a Bayesian inference framework to examine the Repressilator model [2]. In particular, we investigate how the number and choice of observed species affects the "sloppiness" of the inferred free parameters, which we measure in terms of the difference between the prior and the posterior distributions as measured by the Kullback-Leibler divergence. We observe that different species may have varying information content, which raises important questions regarding the optimisation of experimental protocol.

1. INTRODUCTION

The use of mechanistic models based on systems of ordinary differential equations (ODEs) is of great importance in systems biology and they have been successfully used to accurately describe the behaviour of a wide range of biological systems. Such models can be considered a codification of the underlying structure of the system, and individual terms in the equations correspond directly to the reactions believed to be taking place. This makes it relatively straightforward to develop and encode new hypotheses about complex biological systems in terms of (often nonlinear) systems of ODEs. The main challenge with this approach, however, lies in finding the parameter values and initial conditions for each system of ODEs that allow the model to reproduce the observed behaviour. This becomes especially challenging as the complexity of models increases, incorporating sometimes even hundreds of free parameters.

It is difficult to accurately measure the biochemical parameters directly as these are often reaction rates which may potentially take on a range of values according to the experimental conditions. Indeed, it has been observed that for a particular model, there are often multiple sets of parameters that yield an accurate fit to the observed data. This has been investigated recently in [1], where the authors have shown that many published models in the area of systems biology have "sloppy" parameter sensitivities. Even though individual rate parameters were often poorly constrained, it was found that tight uncertainties on the model responses were still possible. The authors suggest that uncertainty estimates should therefore be based on sampling from all parameter sets which produce model responses consistent with the available data. It makes sense then to adopt the Bayesian framework, which also allows model comparison through Bayes factors, which can be estimated using the inferred posterior distributions [3].

In reality not all chemical species may be measurable, due to either technical or cost issues, and it is often therefore necessary to infer parameter values and initial conditions based on observing only a subset of all the chemical species present in the model. In this paper we investigate the effect the number of observed species has on the "sloppiness" of the inferred free parameters, which we measure in terms of the difference between the prior and the posterior distributions as measured by the Kullback-Leibler divergence. This allows us to attempt to quantify the information content of particular observed species, which could potentially enable us to guide experimental protocol in the most efficient manner.

We examine the Repressilator model, a synthetic network of transcriptional regulators based on a cycle of repressors. This symmetrical mathematical model, although relatively simple may reproduce complex nonlinear oscillatory dynamics, and allows us to clearly see the extent to which information content varies depending on the number and choice of species observed for the purpose of parameter inference. Such oscillatory systems are of particular interest, since many fundamental biological processes exhibit this type of behaviour. Examples include the cell cycle, photosynthesis in plants and many other processes associated with circadian rhythms [4], the underlying design principles of which are still poorly understood despite attempts to characterise their behaviour through the use of complex ODE-based mathematical descriptions based on the underlying biology [5]. In contrast, the Repressilator was constructed with the emphasis on reproducing function, and can then be compared to naturally occurring networks to improve our understanding of them.

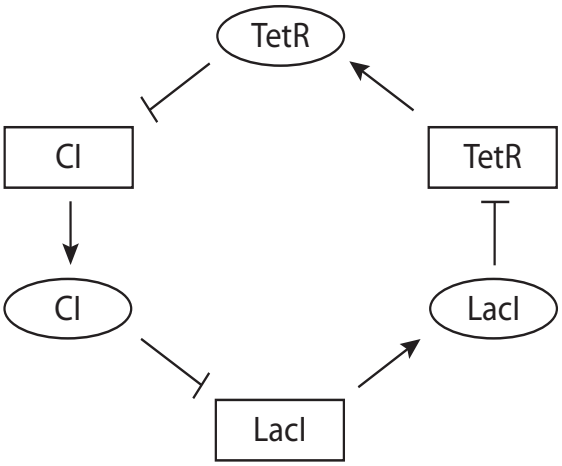


Figure 1. A circuit diagram of the synthetic Repressilator network. mRNA is represented by a rectangle and a protein by an oval. This loop of repressor proteins is capable of exhibiting oscillatory responses.

2. THE REPRESSILATOR

The Repressilator is a synthetic network which was created in *Escherichia coli* using a chain of three repressor genes and their corresponding proteins [2]. The system is capable of producing oscillatory responses, and is described by the following equations

$$\begin{aligned}
\frac{dm_l}{dt} &= -m_l + \frac{\alpha}{(1+p_c^n)} + \alpha_0 \\
\frac{dp_l}{dt} &= -\beta(p_l - m_l) \\
\frac{dm_t}{dt} &= -m_t + \frac{\alpha}{(1+p_l^n)} + \alpha_0 \\
\frac{dp_t}{dt} &= -\beta(p_t - m_t) \\
\frac{dm_c}{dt} &= -m_c + \frac{\alpha}{(1+p_t^n)} + \alpha_0 \\
\frac{dp_c}{dt} &= -\beta(p_c - m_c)
\end{aligned}$$

where $p_{l,t,c}$ are the protein levels for *LacI*, *TetR* and *CI* respectively, $m_{l,t,c}$ are the corresponding mRNA levels, α and α_0 are promoter strength parameters, β represents the ratio of protein decay to mRNA decay and n is a Hill coefficient, which is taken to be constant, $n = 2$, for this paper. There are therefore 3 rate parameters and 6 initial conditions which must be inferred. Stability analysis in [2] shows the sets of parameter values for which the Repressilator system exhibits stable and unstable steady states.

3. INFERENCE METHODS

The Bayesian framework is adopted, whereby parameter samples are generated from the posterior distribution of likely solutions. In addition to allowing for uncertainty over "sloppy" parameters to be taken into account, the Bayesian approach enables us to easily incorporate prior information or beliefs about the system under study in a principled and consistent manner. The posterior distribution provides us with an updated measure of our beliefs. This distribution can be calculated from the likelihood and prior distributions using Bayes' Theorem

$$p(\boldsymbol{\theta} \mid \mathbf{Y}, S) = \frac{\overbrace{p(\mathbf{Y} \mid \boldsymbol{\theta}, S)}^{Likelihood}\overbrace{p(\boldsymbol{\theta})}^{Prior}}{\underbrace{\int p(\mathbf{Y} \mid \boldsymbol{\theta}, S)p(\boldsymbol{\theta})d\boldsymbol{\theta}}_{MarginalLikelihood}} \tag{1}$$

where $\boldsymbol{\theta}$ is the set of model parameters, $\mathbf{Y}$ is the experimental data, and S is the model, in this case defined by a system of differential equations.

The likelihood probability distribution provides a measure of mismatch between the experimental data and the model response given a set of parameter values, and accounts for the many different types of error, both inherent and experimental. We assume independent normally distributed errors across our experimental data, with variance σ_m inferred for species m. In order to avoid numerical problems when dealing with the products of small probabilities, we work in log space when calculating likelihoods, and the overall log likelihood is therefore the sum of the logs of the likelihoods over all N data points for each of the M observed species

$$\log(p(\mathbf{Y} \mid \boldsymbol{\theta}, S)) = \sum_{m=1}^{M}\sum_{n=1}^{N} \log(\mathrm{N}_{\mathbf{Y}_{m,n}}(\boldsymbol{\varphi}_{m,n}(\boldsymbol{\theta}), \sigma_m)) \tag{2}$$

where $\boldsymbol{\varphi}_{m,n}(\boldsymbol{\theta})$ is the solution to the system of ODEs defining the model S.

We place gamma priors with large variance over the model parameters and the initial conditions of unobserved species to reflect our lack of knowledge regarding their "true" values. We employ lower variance Gaussian priors over the initial values of observed chemical species centered on the first experimental data point measured.

3.1. Markov Chain Monte Carlo

We obtain independent samples from the posterior distribution, $p(\boldsymbol{\theta}|\mathbf{Y}, S)$, by running Markov chains to convergence over a product target density indexed by a temperature parameter t such that

$$\widetilde{p}(\boldsymbol{\theta}|\mathbf{Y}, S, \mathbf{t}) = \prod_{n=1}^{N} p(\boldsymbol{\theta}|\mathbf{Y}, S, t_n) \tag{3}$$

where $t_n \in [0, 1]$ and $p(\boldsymbol{\theta}|t_N) = p(\boldsymbol{\theta} \mid \mathbf{Y}, S)$. A time homogeneous Markov transition kernel which has $p(\boldsymbol{\theta}|t_N)$ as its stationary distribution can be constructed from both local Metropolis-Hastings proposal moves and global exchange moves between the tempered chains [6] thus allowing freer movement within the parameter space. This

is important when investigating models with nonlinear responses since they often induce correspondingly nonlinear likelihood surfaces, which are very uneven and difficult to sample from using conventional methods. 3 populations of 10 Markov chains were run simultaneously for 25,000 iterations and their convergence was monitored using Gelman's $\hat{R}$ statistic, which compares within-chain and between-chain variance.

4. INFORMATION CONTENT ANALYSIS

We wish to investigate the effect the number of observed species has on the "sloppiness" of the inferred free parameters. Sharper posterior distributions across the free parameters mean a greater information content in the observed data used for the inference procedure, since the system has been better identified. We therefore use KL divergence, measuring the difference between priors and posteriors, to assess the "sloppiness" induced by the observed data used. KL divergence between a prior, P, and posterior, P_{post}, is given by

$$\begin{aligned} KL(P||P_{post}) &= E_P\left[\log \frac{P}{P_{post}}\right] \\ &= E_P\left[\log P\right] - E_P\left[\log P_{post}\right] \quad (4) \end{aligned}$$

where E_P is the expectation with respect to the prior. Conveniently, the first term in Equation 4 can be calculated analytically for the gamma priors employed in this paper, since this is just the information entropy of a gamma distribution.

The second term in Equation 4 must be estimated since the distribution of the posterior is not known analytically. Given S samples drawn from the posterior, P_{post}, using a Markov chain, we may estimate the probability that some value θ is drawn from the posterior by using a kernel density estimator. An approximation for $P_{post}(\theta)$ is given by

$$\hat{P}_{post}(\theta) = \frac{1}{S}\sum_{s=1}^{S} K_h(\theta, \theta_s) \quad (5)$$

where $\theta_s \sim P_{post}$ and K_h is some kernel density function with positive support and width h. Substituting this into Equation 4, written as an integral, we obtain the estimator

$$\begin{aligned} \hat{E}_P\left[\log P_{post}\right] &= \int P(\theta) \log\left[\frac{1}{S}\sum_{s=1}^{S} K_h(\theta, \theta_s)\right] d\theta \\ &\approx \frac{1}{N}\sum_{n=1}^{N} \log\left[\frac{1}{S}\sum_{s=1}^{S} K_h(\theta_n, \theta_s)\right] \quad (6) \end{aligned}$$

where $\theta_n \sim P$.

5. EXPERIMENTS AND RESULTS

For the Repressilator model in section 2, we sample from a 10 dimensional space consisting of 3 free rate parameters, 6 initial values and a variance parameter estimating the noise present in the data.

Oscillatory experimental data is generated by allowing the system to reach a limit cycle using the parameters $\alpha = 50$, $\alpha_0 = 0.05$ and $\beta = 5$, and then taking the initial conditions at the time when $LacI$ mRNA is at its maximum value in its cycle. 49 data points are observed for each species from $t = 0$ to $t = 24$ at intervals of 0.5. Gaussian noise is then added with variance set to 1 percent of the standard deviation of all the data points for each observed species.

All unknown parameters are inferred using the methods in section 3 based on the set of observed species. 5000 samples are drawn after convergence of the chains and these are used to approximate the posterior distribution. The KL divergence is estimated 10 times, each time using 100,000 samples drawn from the prior.

5.1. Experiment 1 - Number of Observed Species

We start with all 6 species observed and perform inference over the 10 unknown values, although it is mainly the 3 free rate parameters, α, α_0 and β, that we are interested in. We repeat this inference step 5 more times, each time reducing the number of observed species by 1. Figure 2 shows boxplots of the KL divergence estimates for each of the 3 free rate parameters.

We note that the information content, as quantified by the KL divergence, is least when observing just 1 species and increases monotonically as the number of observed species increases. The most drastic change in information content occurs when the number of species being observed is increased from 1 to 2.

5.2. Experiment 2 - Choice of Observed Species

We now perform inference with only 1 species observed at a time. Figure 3 shows a boxplot of the KL divergence estimates for parameter 1. We see that the information content changes depending on which species is observed.

The Repressilator is a symmetric system with the 3 mRNA and protein pairs forming a loop. However, the observed data used in the experiments is not symmetric in that it was generated with $LacI$ mRNA at the maximum level in its oscillatory cycle. We observe that the information content decreases as the observed species becomes further separated from the $LacI$ mRNA in the circuit. Since the genetic network forms a loop, the $TetR$ protein is the furthest distance away from the $LacI$ mRNA. It is interesting to note that it also has the lowest KL divergence, and therefore the most "sloppiness" in the inferred parameters. As we progress round the loop in both directions towards the $LacI$ mRNA we observe that the information content increases to a maximum level, which is when the level of $LacI$ mRNA itself is measured.

6. CONCLUSIONS

We have shown how both the number and choice of species observed affects the accuracy or "sloppiness" of the inferred parameter distributions, as measured by the KL divergence between the prior and posterior distributions over

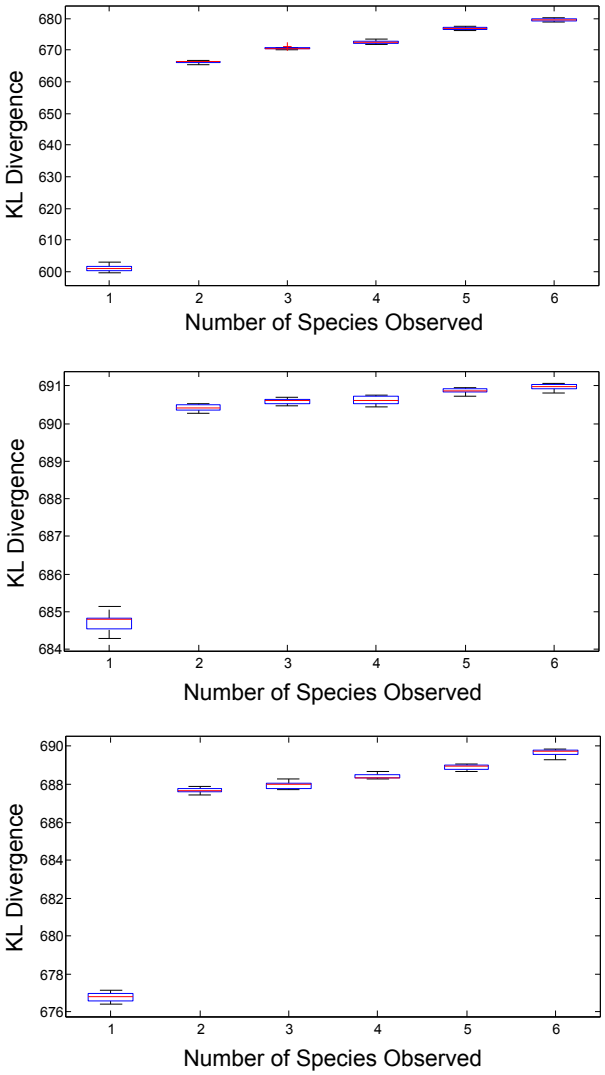


Figure 2. The boxplots in these figures show the KL divergence estimates given the number of species observed for the parameters α (top), α_0 (middle) and β (bottom). At first only the LacI mRNA level is measured. Additional species are then added sequentially, following the loop round anticlockwise, until all species are observed. So for 2 observed species, measurements of LacI protein are also used. For 3 observed species, measurements of TetR mRNA are added etc.

each parameter. This has important implications regarding the design of experiments in which only limited measurements may be taken due to perhaps technical or cost issues, and raises important questions regarding the optimisation of experimental protocol.

Our results suggest that the structure of the proposed model could be first analysed to determine the impact of each species on the "sloppiness" of the parameters being inferred. In our experiments we employed the symmetric and relatively simple Repressilator model. Since current working hypotheses of genetic networks describing, for example, circadian rhythms typically consist of 10 or more species with upwards of 30 free rate parameters to be inferred, further work is needed to examine how the information content of species varies in larger, more complex computational models.

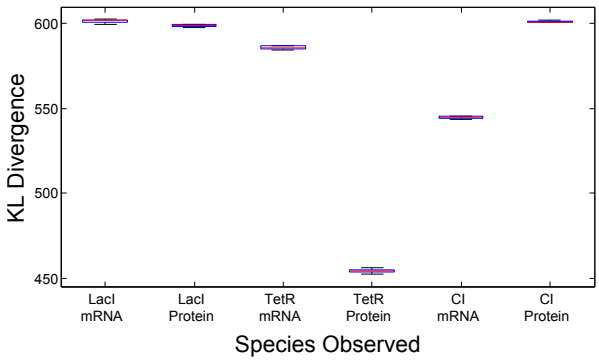


Figure 3. The boxplots in this figure show the KL divergence for parameter α when observations from each of the individual species are used one at a time. It is clear to see that as the species used gets further away from the LacI mRNA, which is at the start of its oscillatory cycle in the experimental data, the KL divergence decreases. It then increases again as the species used gets closer to the $LacI$ mRNA in the loop.

7. ACKNOWLEDGMENTS

Ben Calderhead is supported by Microsoft Research through its PhD Scholarship Programme. Mark Girolami is supported by an EPSRC Advanced Research Fellowship EP/EO52029.

8. REFERENCES

[1] R. N. Gutenkunst, J. J. Waterfall, F. P. Casey, K. S. Brown, C. R. Myers, and J. P. Sethna, "Universally sloppy parameter sensitivities in systems biology models," *PLoS Computational Biology*, vol. 3(10), pp. 1871–1878, 2007.

[2] M. B. Elowitz and S. Leibler, "A synthetic oscillatory network of transcriptional regulators," *Nature*, vol. 403, pp. 335–338, 2000.

[3] V. Vyshemirsky and M. A. Girolami, "Bayesian ranking of biochemical system models," *Bioinformatics*, vol. (Advance Access), 2007.

[4] J. Dunlap, J. Loros, and P. DeCoursey, *Chronobiology: Biological TimeKeeping*, Sinauer, Sunderland, 2003.

[5] G. Kurosawa, A. Mochizuki, and Y. Iwasa, "Comparative study of circadian clock models, in search of processes promoting oscillation," *Journal of Theoretical Biology*, vol. 216, pp. 193–208, 2002.

[6] E. Marinari and G. Parisi, "Simulated tempering: A new monte carlo scheme," *Europhysics Letters*, vol. 19, pp. 451, 1992.

BGMM: A BETA-GAUSSIAN MIXTURE MODEL FOR CLUSTERING GENES WITH MULTIPLE DATA SOURCES

Xiaofeng Dai, Harri Lähdesmäki, Olli Yli-Harja

Department of Signal Processing,
Tampere University of Technology, Tampere, Finland
{xiaofeng.dai,harri.lahdesmaki,olli.yli-harja}@tut.fi

ABSTRACT

This paper presents a novel Beta-Gaussian mixture model, BGMM, for clustering genes based on gene expression data and protein-DNA binding data. An expectation maximization (EM) type of algorithm for Beta mixture model is first developed and then combined with that of Gaussian mixture model. This combined algorithm can jointly estimate the parameters for both Beta and Gaussian distributions and is used as the core in the BGMM method. Four well-studied model selection methods, Akaike information criterion (AIC), modified AIC (AIC3), Bayesian information criterion (BIC), and integrated classification likelihood-BIC (ICL-BIC) are applied to estimate the number of clusters, and AIC3 works best for BGMM in our simulations. Simulations also indicate that combining two different data sources into a single mixture model can greatly improve the clustering accuracy and stability. The proposed BGMM method differs from other mixture model based methods in its integration of two different data types into a single and unified probabilistic modeling framework, which provides a more efficient use of multiple data sources than methods that analyze different data sources separately.

1. INTRODUCTION

It has become more and more acknowledged that different data sources offer information from different aspects, and their combination can make the prediction more robust. Thus how to integrate different data types to make the results more accurate has become one of the most challenging problems in the field of system biology. In the context of gene clustering, gene expression data has been widely used with the assumption that genes which have similar expression pattern under different conditions have similar cellular functions, are likely to be involved in the same cellular processes [6]. This assumption might be too ideal considering the complexity of real biological systems. However, if we could incorporate physical binding information, such as the probabilities of certain binding events occurring among gene products and genes (protein-DNA binding data), into expression data based clustering framework, the clustering results might be more trustable with respect to similar cellular functions, processes and co-regulation. In this study, we developed a clustering algorithm which can cluster genes based on their expression data and protein-DNA binding data.

Many unsupervised methods have been developed and widely used in gene clustering. They can be roughly classified into three categories, which are heuristic, iterative relocation and model-based methods [4]. The first two approaches have problems with solving some basic practical issues such as 'how to define the number of clusters' and 'how to handle outliers'. In model-based methods, the first question can be recasted as the model selection problem. For the second problem, the outliers can be handled by adding one or more components which represent a different distribution for them [4, 5]. Moreover, model-based clustering methods outweigh approaches within the other two categories in their statistical nature [4]. So in this study, we choose model-based clustering as the framework for unsupervised data fusion.

Expectation maximization (EM) algorithm is generally used to solve the problem of maximum likelihood estimation with incomplete data, and thus is commonly adopted in model-based clustering. Although EM algorithm for Gaussian distribution is well-known, less information is available about EM algorithm for other distributions, not mentioning combinations of different distributions. In our study, gene expression data and protein-DNA binding data are integrated into a combined mixture model. We first developed an EM type of algorithm for beta distribution, and then combined it with that for Gaussian distribution. Simulation results show that our joint mixture model can yield better results compared with either of its component models, which demonstrates the idea that the more data that are integrated the better the result turns out to be.

Criteria for model selection can be classified into likelihood-based methods and approximation-based methods, of which approximation-based methods are widely preferred by its simplicity and less computational cost [10]. These methods include penalized likelihood, closed-form approximations to the Bayesian solution, and Monte Carlo sampling of the Bayesian solution, among which penalized likelihood method is most prevalent. Four well-known penalized likelihood criteria, Akaike information criterion (AIC), modified AIC (AIC3), Bayesian information criterion (BIC), and integrated classification likelihood-BIC

(ICL-BIC) were tested in BGMM and its component models (Beta mixture model 'BMM', Gaussian mixture model 'GMM') in this study. AIC and BIC are commonly used as the criterion for GMM [2, 5], and ICL-BIC is reported to work better for BMM according to [5]. Our simulation results suggest using AIC and AIC3 in BMM and BGMM respectively and embrace the tradition of employing BIC in GMM.

The following sections are organized as 'Methods', 'Results', and 'Conclusions'. Section 'Methods' is divided into two parts. In the first part, mixture model based clustering and EM algorithm are discussed, where the classic EM for GMM, our EM for BMM, and the joint EM for BGMM are all introduced. The second part of this section introduces the formulation of four tested model selection criteria (AIC, AIC3, BIC, ICL-BIC), and how the optimal criteria for each model was chosen. In section 'Results', we evaluated and compared the performance of BGMM with BMM and GMM. In section 'Conclusions', we summarized this study and discuss its possible extension and applications to other problems, and mentioned the possible future work that is related to the proposed BGMM.

2. METHODS

2.1. Mixture model based clustering and EM algorithm

In model-based clustering method, each observation x is drawn from a finite mixture distributions with the prior probability π_i, component-specific distribution f_i and its parameters θ_i. The formula is given as

$$f(x;\Theta) = \sum_{i=1}^{g} \pi_i f_i(x;\theta_i), \tag{1}$$

where $\Theta = \{(\pi_i, \theta_i) : i = 1, \ldots, g\}$ is used to denote all unknown parameters, with the restriction that $0 \leq \pi_i \leq 1$ for any i and that $\sum_{i=1}^{g} \pi_i = 1$. Note that g is the number of components in this model.

EM algorithm is then derived for the above model-based clustering. The data log-likelihood can be written as

$$\log L(\Theta) = \sum_{j=1}^{n} \log\Big(\Big[\sum_{i=1}^{g} \pi_i f_i(x_j;\theta_i)\Big]\Big), \tag{2}$$

given $X = \{x_j : j = 1, ..., n\}$, whose direct maximization, however, is difficult.

In order to make the maximization of Equation 2 tractable, the problem is casted in the framework of incomplete data. Define z_{ji} as the indicator of whether x_j is from component i, i.e., $z_{ji} = 1$ if x_j is indeed from component i, and $z_{ji} = 0$ otherwise. Then the complete data log-likelihood becomes

$$\log L_c(\Theta) = \sum_{j=1}^{n} \sum_{i=1}^{g} z_{ji} \log\left(\pi_i f_i(x_j;\theta_i)\right). \tag{3}$$

In the EM algorithm, E step computes the expectation of the complete data log-likelihood which is denoted as Q

$$\begin{aligned} Q(\Theta;\Theta^{(m)}) &= E_{\Theta^{(m)}}(\log L_c|X) \\ &= \sum_{j=1}^{n} \sum_{i=1}^{g} \tau_{ji}^{(m)} \log\left(\pi_i f_i(x_j;\theta_i)\right), \end{aligned} \tag{4}$$

where $\Theta^{(m)}$ represents the parameter estimates at iteration m. M step updates the parameter estimates to maximize Q. The algorithm is iterated until convergence. Note that zs in Equation 3 are replaced with τs in Equation 4, and the relationship between these two parameters is $\tau_{ji} = E[z_{ji}|x_j, \hat{\theta}_1, ..., \hat{\theta}_g; \hat{\pi}_1, ..., \hat{\pi}_g]$. The set of parameter estimates $\left\{\hat{\theta}_1, ..., \hat{\theta}_g; \hat{\pi}_1, ..., \hat{\pi}_g\right\}$ is a maximizer of the expected log-likelihood for given τ_{ji}s, and we can assign each x_j to its component based on $\{i_0|\tau_{ji_0} = \max_i \tau_{ji}\}$.

2.1.1. GMM and its EM algorithm

The most widely used and well known model-based clustering method is finite GMM, in which each component is assumed to follow a Gaussian distribution. In this study we use the standard p dimensional normal distribution with mean μ_i and unconstrained covariance matrix V_i for each component in GMM [7]. We run the EM algorithm multiple times with different initial values, where fuzzy c-means clustering algorithm is used for initialization, to avoid possible local maxima.

2.1.2. BMM and its EM algorithm

In order to make the model-based method to work for data within boundaries $[0, 1]$, we developed a BMM with the assumption that each component is a product of independent beta distributions. The probability density function is defined as

$$f_i(x;\alpha_i,\beta_i) = \prod_{j=1}^{p} \frac{x^{\alpha_{ij}-1}(1-x)^{\beta_{ij}-1}}{B(\alpha_{ij},\beta_{ij})}. \tag{5}$$

The details of our EM type of algorithm for BMM is described below. First, initialize the parameters. αs and βs for each component beta distribution k ($k \in \{1, \ldots, p\}$) are initialized by method-of-moments so that their means are randomly distributed within the range of $x_{1k}, \ldots, x_{nk}$ and variances are equal for all clusters (g); and for π_is, they are initialized with the uniform probability $1/g$. Second, run E-step. Calculate τ_{ji} with current parameters, according to which x_js are clustered to their corresponding clusters using z_{ji_0}s (where $\{i_0|\tau_{ji_0} = \max_i \tau_{ji}\}$). Third, run M-step to maximize Equation 3. Given the hard clusters obtained in E-step, numerically estimate the new parameters $\hat{\alpha}$s and $\hat{\beta}$s using the maximum likelihood principle (matlab function 'betafit' is used here for this purpose), and calculate the new $\hat{\pi}$s by

$$\hat{\pi}_i^{(m+1)} = \sum_{j=1}^{n} \tau_{ji}^{(m)}/n, \tag{6}$$

$$\tau_{ji}^{(m)} = \frac{\pi_i^{(m)} f_i(x_j;\alpha_i^{(m)},\beta_i^{(m)})}{\sum_{i=1}^{g} \pi_i^{(m)} f_i(x_j;\alpha_i^{(m)},\beta_i^{(m)})}. \tag{7}$$

2.1.3. BGMM and its EM algorithm

EMs for BMM and GMM are combined into a single framework in BGMM with the assumption that, for each component i, the expression and binding data are independent. The procedures of parameter maximization for both data types are the same as those for BMM and GMM, except that the calculation of τs is the product of two distributions

$$\tau_{ji}^{(m)} = \frac{\pi_i^{(m)} f_i^G(x_j;\mu_i^{(m)},V_i^{(m)}) f_i^B(x_j;\alpha_i^{(m)},\beta_i^{(m)})}{\sum_{i=1}^g \pi_i^{(m)} f_i^G(x_j;\mu_i^{(m)},V_i^{(m)}) f_i^B(x_j;\alpha_i^{(m)},\beta_i^{(m)})}. \quad (8)$$

Note that the superscripts (G) and (B) of fs mean that the parameters they represented are from GMM and BMM respectively.

In this study, for each data set we run each EM algorithm 100 times with different initial values. The convergence threshold (where Q is used to monitor the convergence) and maximum number of iterations were set to 0.0001 and 100 respectively for all the tested models, and all the simulations have reached their convergences according to the statistics stored during the simulations.

2.2. Model Selection

Four well-known approximation-based model selection criteria, AIC [1, 2], AIC3 [2, 3], BIC [8, 9], and ICL-BIC [5] are compared in BGMM and its component models, according to which the optimal criterion for each model is chosen. Calculations for the above criteria are defined in

$$\begin{aligned} AIC &= -2\log L(\hat{\Theta}) + 2d, && (9)\\ AIC3 &= -2\log L(\hat{\Theta}) + 3d, && (10)\\ BIC &= -2\log L(\hat{\Theta}) + d\log(nM), && (11)\\ ICL-BIC &= -2\log L(\hat{\Theta}) + d\log(nM) \\ &\quad -2\sum_{j=1}^{n}\sum_{i=1}^{g}\tau_{ji}\log(\tau_{ji}), && (12) \end{aligned}$$

where d is the number of free parameters in its corresponding model, and M in equations 11 and 12 is the total dimension of the data ($M = \sum_{w=1}^{W} M_w$, M_w is the dimension of data set w and W is the number of input data sets). Note that $-2\sum_{j=1}^{n}\sum_{i=1}^{g}\tau_{ji}\log(\tau_{ji})$ is the estimated entropy of the fuzzy classification matrix $C_{ji} = (\tau_{ji})$ [5].

The number of free parameters d are different in different models. In GMM, we have $(p^2+p)g/2$ σs, pg μs, and $g-1$ free πs ($\sum_{i=1}^{g}\pi_i = 1$), so $d_G = (p^2+p)g/2 + pg + g - 1$. In BMM, as we have pg αs, pg βs, and also $g-1$ free πs, $d_B = 2gp + g - 1$. In the joint model, the number of free parameters is the sum of those in its parents' models minus one set of free πs, thus we have $d_{BG} = d_B + d_G - (g-1)$.

3. RESULTS

In this study, we compared the performance of BMM, GMM and BGMM using two artificial datasets, which are generated by a simplified model (we generate data from a diagonal covariance model although our model assumes unconstraint covariance). Both datasets are designed to have three clusters and 60 by 4 dimensions ($n = 60$, $p = 4$). Parameters for different dimensions within each cluster are the same in the first data set but different in the second one, called 'non-mixed' and 'mixed' cases respectively. We designed two kinds of data for each data type within each data set, namely 'gB', 'bB', 'gG' and 'bG', which are short for 'good Beta' (less noisy, Beta distribution), 'bad Beta' (more noisy, Beta distribution), 'good Gaussian' (less noisy, Gaussian distribution), and 'bad Gaussian' (more noisy, Gaussian distribution) respectively. We also designed two kinds of 'bG', 'bG$_m$' and 'bG$_v$', which are hard to be clustered compared to 'gG' with respect to means and variances respectively. Parameter settings for the datasets are listed in Table 1, where the combination of 'good Gaussian variance' and 'bad Gaussian mean' is 'bG$_m$', and the combination of 'good Gaussian mean' and 'bad Gaussian variance' is the case 'bG$_v$'. All the simulations are repeated 20 times with randomly generated data sets.

In order to choose the optimal model selection criterion (with the highest score) for each model, we summed up the number of hits of the correct number of clusters for each data combination in both simulations. The summation results for AIC, AIC3, BIC, and ICL are 93, 71, 16 and 10 respectively in BMM, 8, 54, 64, 58 respectively in GMM, and 35, 101, 43, 43 respectively in BGMM, according to which AIC, BIC and AIC3 are chosen as the criteria for BMM, GMM, and BGMM respectively.

We developed one scoring system for evaluating the clustering accuracy, which is denoted as 'E score'

$$\begin{aligned} e_j(r) &= \begin{cases} 1 & if\ \hat{z}_{ji} = 1\ and\ r_i = T_j \\ 0 & otherwise \end{cases} \\ E &= \max_{r\in R}\sum_{j=1}^{n} e_j(r)/n \quad (13)\\ R &= \big\{r = (r_1,\ldots,r_{\hat{g}}) : \forall i \neq j\ r_i \neq r_j; \\ &\quad r_i \in \{1,\ldots,\max\{\hat{g},g\}\}\big\}. \end{aligned}$$

In this scoring system, T_j denotes the ground truth clustering membership of data j; R stands for all possible associating ways between the estimated and the true clusters, where r_i is the label of data belonging to component i predicted by the clustering algorithm, and r is chosen from labels $1, 2, \ldots, \max\{\hat{g}, g\}$ ($\hat{g}$ and g are the largest labels in the estimated and ground truth clustering respectively); also note that e represents the individual score of each gene, E is the average score of all the genes for each repetition, 'E score' of each repetition is the one corresponding to the optimal Q, and the final 'E score' of each data set is the median of the 20 'E score's. This scoring system evaluates the overall performance of the model since it not only records the accuracy of the results but also reflects the influence of the criterion for model selection.

The comparison results of BGMM with its component models are shown in Fig. 1. For expression data whose variances are not too large, the joint model can improve

Data			Data set 1			Data set 2										
			c1	c2	c3	c1				c2				c3			
Beta	good	alpha	10	20	25	15	20	25	20	20	25	15	5	1	20	1	30
		beta	20	10	20	20	15	20	25	20	25	15	5	20	1	30	1
	bad	alpha	10	15	17	15	10	25	20	10	5	15	12	30	25	30	35
		beta	20	20	18	10	15	20	25	5	10	12	15	25	30	35	30
Gaussian	good	mean	7	8	9	9	-9	11	-11	10	-10	12	-12	11	-11	13	-13
		variance	0.3	0.4	0.2	0.7	0.2	0.7	0.2	0.8	0.3	0.8	0.3	0.9	0.4	0.9	0.4
	bad	mean	7.5	8	8.5	9.5	-9.5	10	-10	9	-9	9.5	-9.5	10	-10	9	-9
		variance	1	0.9	0.8	1	1	1.5	1.5	1.5	1.5	2	2	2	2	1	1

Table 1. Data sets designed for simulations

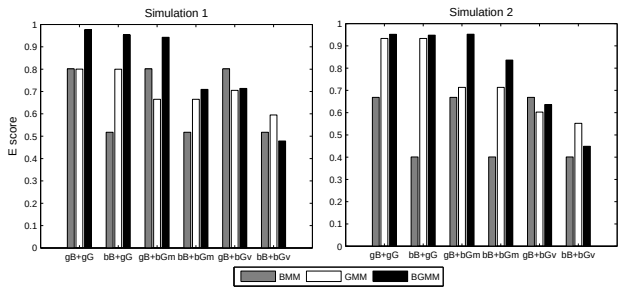


Figure 1. Performance test of BGMM.

the clustering accuracy regardless of the quality of the data compared with either of its component models (E scores for cases 'gB+gG', 'bB+gG', 'gB+bG$_m$' and 'bB+bG$_m$' in BGMM are higher than those in GMM or BMM). However, if the expression data contains too much noise with respect to large variances('gB+bG$_v$', 'bB+bG$_v$'), the joint model does not necessarily yield better results. These results indicate that BGMM has the power of reinforcing each component model with information from the other one in both mixed and non-mixed cases but is sensitive to the variances of the Gaussian distributed data.

4. CONCLUSIONS

This paper presents a novel method based on Beta-Gaussian mixture model, BGMM, for gene clustering from multiple data sources. In this study, we integrated gene expression data and protein-DNA binding data, where expression data and protein-DNA binding data are assumed to be of Gaussian and Beta distribution respectively. An EM type of algorithm for estimating parameters from beta distribution is developed and combined with the EM for Gaussian distribution into a single framework, which is used as the core of BGMM. In principle, this proposed BGMM is not limited to the data we have used here, and any data that can be modeled as Gaussian and Beta distribution could be integrated into this framework. This work demonstrates one approach of integrating information from multiple data sources. Data of other distributions can also be incorporated by joining EM algorithm of that particular distribution into this framework in a similar way. Therefore BGMM is applicable to many problems and not limited to the particular problem considered here.

For future work, we will first apply our method to real data, where a possible problem might be the computational complexity due to the large dimensions of the data. Many techniques might be used to handle these problems such as reducing the dimension of the data or employing a faster EM framework. Second, we will integrate more data types into the proposed mixture model framework, where the most obvious start is to develope a stratified BGMM [8] which could incorporate one more data source by constructing the priors from a third data type.

5. REFERENCES

[1] H. Akaike, "A new look at the statistical identification model", *IEEE Transactions on Automatic Control* vol. 19, pp. 716-723, 1974.

[2] C. Biernacki and G. Govaert, "Choosing models in model-based clustering and discriminant analysis", *J. Statis. Comput. Simul.*, vol. 64, pp. 49-71, 1999.

[3] H. Bozdogan, "Model Selection and Akaike Information Criterion (AIC): The General Theory and its Analytic Extensions" *Psychometrika* vol. 52, pp. 345-370, 1987.

[4] C. Fraley and A. E. Raftery, "Model-based clustering, discriminant analysis, and density estimation", *Journal of the American Statistical Association*, vol. 97, no. 458, pp. 611-631, 2002.

[5] Y. Ji, C. Wu, P. Liu, J. Wang, R. K. Coombes, "Applications of beta-mixture models in bioinformatics", *Bioinformatics*, vol. 21, no. 9, pp. 2118-2122, 2005.

[6] D. X. Jiang, C. Tang, A. D. Zhang, "Cluster analysis for gene expression data: a survey", *IEEE Transactions on knowledge and data engineering*, vol. 16, no. 11, pp. 1370-1386, 2004.

[7] G. Mclachlan and D. Peel, "Finite mixture models", *John Wiley & Sons*, 2000.

[8] W. Pan, "Incorporating gene functions as priors in model-based clustering of microarray gene expression data", *Bioinformatics*, vol. 22, no. 7, pp. 795-801, 2006.

[9] G. Schwarz, "Estimating the dimension of a model" *Annals of Statistics* vol. 6, pp. 461-464, 1978.

[10] P. Smyth, "Model selection for probabilistic clustering using cross-validated likelihood", *Statistics and Computing*, vol. 9, pp. 63-72, 2000.

FEATURE REPRESENTATION OF DNA SEQUENCES FOR MACHINE LEARNING TASKS

Robertas Damaševičius

Software Engineering Department, Kaunas University of Technology,
Studentų 50-415, LT-51368, Kaunas, Lithuania
robertas.damasevicius@ktu.lt

ABSTRACT

Recognition of specific functionally-important DNA sequence fragments is one of the most important problems in bioinformatics. Common sequence analysis methods such as pattern search can not solve this problem because of noisy data and variability of consensus sequences across different species. Machine learning methods such as Support Vector Machine (SVM) can be used for sequence classification, because they can learn useful descriptions of genetic concepts from data instances only rather than explicit definitions. Before applying SVM classification one must define a mapping of classified sequences into a feature space. We analyze binary and k-mer frequency-based feature mapping rules. The efficiency of such rules is demonstrated for the recognition of promoters and splice-junction sites.

1. INTRODUCTION

Biomolecular data mining is the activity of finding significant information in DNA, RNA and protein molecules. The significant information may refer to periodicities, motifs, clusters, genes, protein signatures, grammar rules and classification rules. Common methods are sequence alignment using dynamic programming (Needleman-Wunsch, Smith-Waterman) and substitution-matrix based methods (PAM, BLOSUM). However, such methods are not very effective for recognition of some types of DNA sequence such as *promoters* (short sequences that precede the beginnings of genes) or *splice-junction sites* (boundary points between exons and introns where splicing occurs), because of noisy data and large variability of consensus sequences across species. For such difficult cases machine learning techniques such as Support Vector Machine (SVM) are applied [1-6].

SVM [7] is a supervised learning method for creating binary classification functions from a set of labeled training data. The accuracy of a particular parametric classifier on a given dataset will depend on the relationship between the classifier and the available dataset [8].

There are two groups of approaches for improving the classification results. The *algorithmic approach* focuses on improving or proposing new classification algorithms or kernels, whereas the *data processing approach* focuses on modifying the dataset or extracting relevant features to improve the accuracy of classification. SVM requires that each data instance is represented as a vector of binary/real numbers in a *feature space*. Thus, if there are categorical attributes, we first have to convert them into numeric data. Good mapping of data into feature space can allow achieving better classification accuracy.

Feature mapping rules such as binary mapping [9], frequency-based encoding [10], determinative degree [11], and features constructed based on a combination of various k-mers (k-base long sequences) [12-14] have been an object of extensive research. Here we analyze different position-dependent (binary) and position-independent (frequency-based) data encoding schemes for nucleotides (A, C, G, T) and their groupings (S/W, K/M, R/Y), which can achieve optimal classification results for promoter and splice-site recognition problems.

2. DNA MAPPING RULES

Before applying SVM classification to the available dataset, first, one must define a mapping of classified objects to the feature space. This mapping is called a *feature vector representation* of the subject area. A feature space can be constructed using a) *position-dependent* information, b) *position-independent* information, or c) *both* position-dependent and position-independent information about the presence or absence of specific nucleotides or their k-mers.

A feature mapping rule can be described as a function $M : \hat{S} \to F$, where $\hat{S} = (s_1, s_2, ..., s_N), s_i \in \{A, C, G, T\}^k$ is a DNA sequence, N is the length of a DNA sequence, $k = \|s_i\|$ is the length of the mapped sequence, and $F = (f_1, f_2, ..., f_M)$, where $f_j \in \{0,1\}^l$ in case of a binary feature space, or $f_j \in \Re$ in case of a real number feature space, M is the length of a feature vector, $l = \|f_i\|$ is the length (dimension) of a feature.

Depending upon the values of k and l, we classify binary feature mapping rules as the binary $1 \to 4$, $1 \to 2$, $1 \to 1$ and $2 \to 1$ rules. Feature mapping rules based on k-mer frequency are categorized according to the DNA alphabet they are applied on.

2.1. Binary mapping rules

2.1.1. Binary $1 \to 4$ rule

An example of the $1 \to 4$ rule is *orthogonal encoding*, where the nucleotides in a DNA sequence are represented by 4-dimensional orthogonal binary vectors:

$$\langle A \to (0001), C \to (0010), G \to (0100), T \to (1000) \rangle \quad (1)$$

For this rule, feature vector size is $4N$.

This rule allows achieving better classification results than the mapping of the nucleotides into 2-dimensional binary vectors, due to the identical Hamming distances between the nucleotide encodings [15].

2.1.2. Binary $1 \to 2$ rule

There are several methods to represent DNA nucleotides using a binary 2-bit code. Jiménez-Montaño *et al.* [16] suggested the rule $\mathrm{A} = 00, \mathrm{G} = 01, \mathrm{T} = 10, \mathrm{C} = 11$. Stambuk [17] defined the rule $\mathrm{T} = 00, \mathrm{C} = 01, \mathrm{G} = 10, \mathrm{A} = 11$. Karasev and Stefanov [18] suggested the rule $\mathrm{C} = 00, \mathrm{T} = 01, \mathrm{G} = 10, \mathrm{A} = 11$. He *et al.* [19] used the rule $\mathrm{C} = 00, \mathrm{T} = 10, \mathrm{G} = 11, \mathrm{A} = 01$.

Actually, there are $4! = 24$ such rules; however, only 3 rules are essentially different, while the remaining rules can be obtained from these by inversion:

Binary 1: $\langle A \to (0,0), C \to (0,1), G \to (1,0), T \to (1,1) \rangle \quad (2)$

Binary 2: $\langle A \to (0,0), C \to (0,1), G \to (1,1), T \to (1,0) \rangle \quad (3)$

Binary 3: $\langle A \to (0,0), C \to (1,1), G \to (0,1), T \to (1,0) \rangle \quad (4)$

For these rules, feature vector size is $2N$.

2.1.3. Binary $1 \to 1$ rule

The $1 \to 1$ rules are unequal representation rules that map one nucleotide into 1 and the remaining nucleotides into 0. There are four such rules: A-rule, C-rule, G-rule and T-rule [20]. These rules reflect the distribution of a particular type of nucleotides along the DNA sequence:

A-rule: $\langle A \to 1, B \to 0 \rangle, B = \{C, G, T\} \quad (5)$

C-rule: $\langle C \to 1, D \to 0 \rangle, D = \{A, G, T\} \quad (6)$

G-rule: $\langle G \to 1, H \to 0 \rangle, H = \{A, C, T\} \quad (7)$

T-rule: $\langle T \to 1, V \to 0 \rangle, V = \{A, C, G\} \quad (8)$

For these rules, feature vector size is N.

2.1.4. Binary $2 \to 1$ rules

The $2 \to 1$ rules are based on the grouping of the 4-letter DNA alphabet into two subsets of two nucleotides each. There are 3 different such partitions, therefore there are 3 different binary mapping rules that map a nucleotide onto a binary number. Each of these rules represents a different aspect of the DNA molecule structure.

The SW mapping rule ($\{A, T\}$ vs. $\{G, C\}$) reflects the difference in the number of hydrogen bonds in the DNA molecule. Each strong (S) nucleotide (C or G) has 3 hydrogen bonds, and each weak (W) nucleotide (A or T) has only 2 hydrogen bonds. This rule is particularly appropriate to analyze genome-wide correlations [21].

SW rule: $\langle S \to 1, W \to 0 \rangle, S = \{A, T\}, W = \{C, G\} \quad (9)$

The RY rule ($\{A, G\}$ vs. $\{T, C\}$) describes how *purines* (R) and *pyrimidines* (Y) are distributed along the DNA sequence. This rule corresponds to the chemical composition bias in the DNA strand.

RY rule: $\langle R \to 1, Y \to 0 \rangle, R = \{A, G\}, Y = \{C, T\} \quad (10)$

The KM rule ($\{A, C\}$ vs. $\{T, G\}$) describes how *amines* (M) and *ketones* (K) are distributed along the DNA sequence.

KM rule: $\langle K \to 1, M \to 0 \rangle, K = \{A, C\}, M = \{G, T\} \quad (11)$

For these rules, feature vector size is N.

2.2. K-mer frequency rules

K-mers are lists or ordered sets of nucleotide sequence elements, which can be described as a k-tuple $(a_1, a_2, \ldots, a_k)$, where $a_i \in \hat{S}$ for all $\mathrm{i} = 1, 2, \ldots, \mathrm{k}$. Feature vector is constructed using a frequency (or probability) $p_j = \dfrac{n_j}{N - k + 1}$ of each k-mer in a N-length sequence $\hat{S}$, where n_j is the number of j-th k-mer in $\hat{S}$. Traditionally, k-mers have been used with 4-letter DNA alphabet $\{A, C, G, T\}$. The disadvantage of such mapping rule is its explosive feature space growth: there may be 4^k distinct k-mers in a nucleotide sequence (actually, there are $N - k + 1$ such k-mers in N-length sequence), and a feature vector is composed of 4^k elements:

ACGT: $\langle \hat{S} \to (p_j) \rangle, \hat{S} \in \{A, C, G, T\}^N, j = 1, \ldots, 4^k \quad (12)$

We can construct smaller feature vectors based on the grouping of the 4-letter DNA alphabet into two subsets of two nucleotides each. There are three different such partitions, therefore there are three different grouping-based k-mer frequency mapping rules [22]:

SW k-mer rule: $\left\langle \hat{S} \to \left(p_j\right) \right\rangle, \hat{S} \in \{S,W\}^N, j = 1,...,2^k$ (13)

RY k-mer rule: $\left\langle \hat{S} \to \left(p_j\right) \right\rangle, \hat{S} \in \{R,Y\}^N, j = 1,...,2^k$ (14)

KM k-mer rule: $\left\langle \hat{S} \to \left(p_j\right) \right\rangle, \hat{S} \in \{K,M\}^N, j = 1,...,2^k$ (15)

Note that using SW, RY or KM groupings, a feature vector is much smaller than in case of full nucleotide alphabet, and is composed of only 2^k elements.

3. CASE STUDY

3.1. Datasets

For promoter classification, we use the 2002 collection of data of drosophila (*D. melanogaster*) core promoter regions [23]. The training file contains 1260 examples (372 promoters, 361 introns, 527 coding sequences). The test file contains 6500 examples (1842 promoters, 1799 introns, 2859 coding sequences).

For splice site recognition, we use the dataset from the UCI repository [24] obtained from Genbank 64.1 primate data. The dataset contains 3175 sequences, each 60 bp length starting at position -30 bp and ending at position +30 bp with regard to splice site location, of which 767 (25%) sequences contain exon/intron (EI) sites (donors), 768 (25%) sequences contain intron/exon (IE) sites (acceptors), and 1655 (50%) sequences contain neither EI nor IE sites (negative, N).

3.2. Problem definition

Dataset sequences are mapped into a feature space using feature mapping rules described in Eq. (1-15). A training dataset is an ordered set of features $F = \left(f_1, f_2, ..., f_M\right)$ of the sequences and their assigned class:

$$LS_M = \{(F_i, c_i) \mid i = 1,...,M\} \quad (16)$$

The objective of the classification is to derive from LS_M a classifier $\hat{c}\left(F_j\right)$, which predicts the class of unseen sequence s_j as accurately as possible based on some selected classification accuracy metric. As a classifier, we use SVM$^{\text{light}}$ [25] with power series kernel [26].

3.3. Results

To represent the precision of promoter classification for binary mapping rules (Eq. 1-11) graphically, the *Receiver Operating Characteristic* (ROC) is used (see Figure 1). The perfect classification corresponds to the $(0,100)$ point in the ROC plot.

The best classification results are obtained using Binary 1, KM, A and T rules. This can be explained by the fact that drosophila promoter sequences are characterized by the repeating occurrences of the so called TATA box (TATAA or TATAAA) or the Pribnow box (TATAAT), thus the best results can be achieved using the rules, where A and T nucleotides are coded using different binary values (as, e.g., in the KM rule).

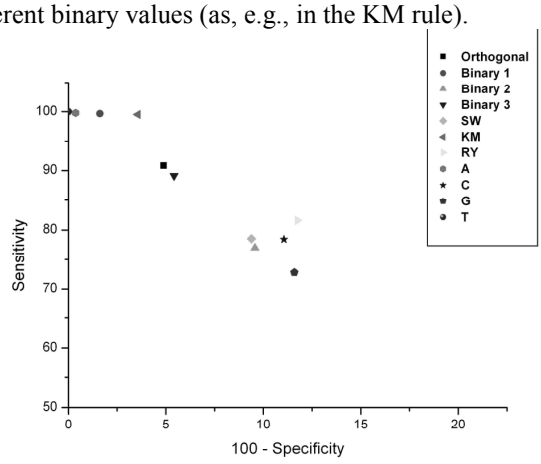


Figure 1. Comparison of promoter classification results for binary feature mapping methods

The splice site classification results for k-mer frequency rules (Eq. 12-15) are summarized in Figure 2.

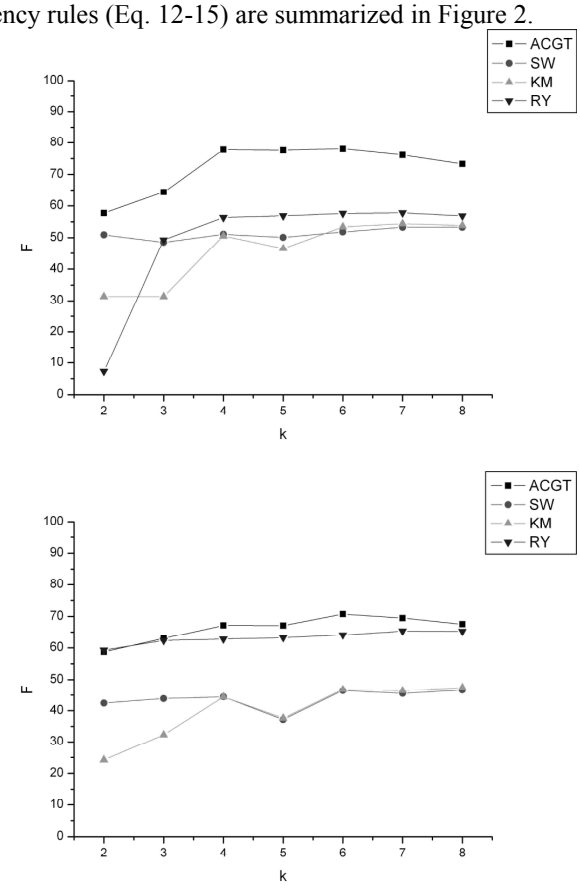


Figure 2. Comparison of splice-site sequence classification results for different k-mers using F-measure: EI vs. N (top), IE vs. N (bottom)

EI splice sites are best recognized with ACGT alphabet 4-mer frequencies (78.05%) and IE splice sites are best recognized using ACGT alphabet 6-mer frequencies (70.75%). Interestingly, 5-mers in both cases have worse recognition results for all types of frequency mapping rules than 4-mers or 6-mers. The classification accuracy for larger k-mers decreases. Out of the 2-nucleotide grouping based feature mapping rules, the RY frequency based feature mapping rule has the best results in both cases, and the results are only slightly worse than the results of the 4-nucleotide frequency mapping for IE splice site recognition. Therefore, if the classification speed is the issue, the RY frequency mapping for IE splice site recognition may yield nearly the same accuracy, though its feature space is significantly smaller (2^k instead of 4^k elements).

The k-mer frequency-based mapping rules perform worse than binary rules (though we solve different problems and use different accuracy metrics). This can be explained by the fact that in frequency based feature representation the nucleotide (k-mer) position information is lost. However, the advantage of frequency rules is smaller feature space, if long sequences are classified.

4. CONCLUSION

The selection of the appropriate feature mapping rule can greatly influence the DNA sequence classification results. The mapping rule should be selected based on the properties of the available data for a specific classification problem. The obtained classification results confirm that the mapping rule(s) with the best classification results correspond to the characteristics of the repeating subsequences ("boxes", consensus sequences) of the analyzed sequences. The selection between binary and frequency mapping rules can provide a trade-off between classification precision and speed.

Future work will focus on the feature space reduction problem to identify features that do not contribute to classification accuracy and can be discarded thus yielding higher recognition speed and better accuracy.

REFERENCES

[1] G. Rätsch, S. Sonnenburg and C. Schäfer, "Learning Interpretable SVMs for Biological Sequence Classification", *BMC Bioinformatics* 2006, 7(Suppl 1):S9.

[2] T. Werner, "The state of the art of mammalian promoter recognition",*Briefings in Bioinformatics* 4(1):22-30, 2003.

[3] A.C. Lorena, and A.C.P.L.F. de Carvalho, "Human Splice Site Identification with Multiclass Support Vector Machines and Bagging", in *Proc. of ICANN 2003*, Istanbul, Turkey, LNCS 2714, Springer, 234-244.

[4] S. Sonnenburg, G. Rätsch, A. Jagota and K. Müller, "New Methods for Splice Site Recognition", in *Proc. of ICANN 2002*, Madrid, Spain, LNCS 2415, Springer, 329-336.

[5] A. Baten, B. Chang, S. Halgamuge and J. Li, "Splice site identification using probabilistic parameters and SVM classification", *BMC Bioinformatics* 2006, 7:S15.

[6] S. Rampone, "Recognition of splice junctions on DNA", *Bioinformatics*, 14(8):676–684, 1998.

[7] V. Vapnik. *Statistical Learning Theory.* Wiley-Interscience, New York, 1998.

[8] C.M. van der Walt and E. Barnard, "Data characteristics that determine classifier performance", in *Proc. of the 16th Annual Symp. of the Pattern Recognition Association of South Africa*, pp. 160-165, 2006.

[9] B. Podobnik, J. Shao, N.V. Dokholyan, V. Zlatic, H.E. Stanley, I. Grosse, "Similarity and dissimilarity in correlations of genomic DNA", *Physica A,* 373:497-502, 2006.

[10] R. Ranawana, and V. Palade, "A neural network based multiclassifier system for gene identification in DNA sequences", *J. Neural Comput. Appl.* 2005, 14, 122–131.

[11] D. Duplij, and S. Duplij, "DNA sequence representation by trianders and determinative degree of nucleotides", *J Zhejiang Univ Sci B.* 2005 August; 6(8): 743–755.

[12] R. Islamaj, L. Getoor, and W.J. Wilbur, "A Feature Generation Algorithm for Sequences with Application to Splice-Site Prediction", in *Proc. of PKDD 2006,* Berlin, Germany, LNCS 4213, Springer, 553-560.

[13] Y. Saeys, S. Degroeve, D. Aeyels, P. Rouzé and Y. Van de Peer, "Feature selection for splice site prediction: A new method using EDA-based feature ranking", *BMC Bioinformatics* 2004, 5:64.

[14] T. Sobha Rani, S. Durga Bhavani and R.S. Bapi, "Analysis of E.coli promoter recognition problem in dinucleotide feature space", *Bioinformatics* 2007 23(5):582-588

[15] B. Demeler, and G.W. Zhou, "Neural network optimization for E. coli promoter prediction", *Nucleic Acids Res* 19:1593–9, 1991.

[16] M. Jimenez-Montano, C. Mora-Basanez, and T. Poschel, "The hypercube structure of the genetic code explains conservative and non-conservative amino acid substitutions in vivo and in vitro", *Biosystems* 1996, 39, 117-125.

[17] N. Stambuk, "Universal Metric Properties of the Genetic Code", *Croatica Chemica Acta* 2000, 73, 1123-1139.

[18] V.A. Karasev and V.E. Stefanov, "Topological Nature of the Genetic Code", *J. Theor. Biol.* 2001, 209, 303-317.

[19] M. He, S. Petoukhov, and P.E. Ricci, "Genetic Code, Hamming Distance and Stochastic Matrices", *Bull. Math. Biol.* 2004, 00, 1–17.

[20] R.F. Voss, "Evolution of long-range fractal correlations and 1/f noise in DNA base sequences", *Phy. Rev. Lett.* 1992, 68(25): 3805–3808.

[21] A. Arneodo, E. Bacry, P. Graves, and J.F. Muzy, "Characterizing long-range correlations in DNA sequences from wavelets analysis", *Phys. Rev. Lett.* 1995, 74, 3293–3296.

[22] R. Damaševičius, "Splice Site Recognition in DNA Sequences Using K-mer Frequency Based Mapping for Support Vector Machine with Power Series Kernel", *Proc. of Int. Conf. on Complex Software Intensive Systems (CISIS 2008)*, March 4–7, 2008, Barcelona, Spain, 687-692.

[23] Drosophila promoter dataset. http://www.fruitfly.org/seq_tools/datasets/Drosophila/

[24] The Machine Learning Database Repository. http://mlearn.ics.uci.edu/databases/molecular-biology/

[25] SVMlight. http://svmlight.joachims.org/

[26] R. Damaševičius, "Optimization of SVM Parameters for Promoter Recognition in DNA Sequences", *Int. Conf on Continuous Optimization and Knowledge-Based Technologies EurOPT-2008*, May 20-23, Neringa, Lithuania.

DECODING THE DYNAMICS OF GENE REGULATORY NETWORKS UNDER AN ALGEBRAIC EXPRESSION MODEL

Janis Dingel[1] *and Olgica Milenkovic*[2]

[1]Institute for Communications Engineering, Technische Universität München, Germany
[2]Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, USA

janis.dingel@tum.de, milenkov@uiuc.edu

ABSTRACT

Building upon a recently proposed algebraic framework for representing expression profiles, we propose a novel method for reverse engineering the dynamics of gene regulatory networks of known topology. The crux of our approach is to describe the update functions for gene expressions in terms of polynomials over finite fields. Establishing a connection to coding theory, we account for stochastic effects and small sample sizes of the measurements via decoding and iterative model refinement algorithms. We test the performance of the new method on synthetic data and apply it to a regulatory sub-net of the *E. coli* gene control network responsible for DNA repair.

1. INTRODUCTION

The problem of predictive statistical inference of gene regulatory networks (GRN) has recently attracted significant attention of the bioinformatics, systems biology, and signal processing research community. Modern high- throughput experimental systems, such as DNA microarrays, enable biologists to monitor whole-genome expression patterns, and many inference techniques have been proposed for describing the coupled dynamics of these patterns. Currently used modeling approaches, capable of capturing nonlinear interactions characteristic of biological control circuits, can broadly be classified into discrete or continuous, deterministic or stochastic models [1]. For example, Boolean network (BN) models are discrete deterministic models where genes are either switched "ON" or "OFF" over discrete steps of time according to a Boolean rule. Probabilistic extensions of Boolean networks account for stochastic effects by using a list of functions that are evaluated according to a probability distribution [2]. Early efforts for reverse engineering of Gene networks under the BN model were based on the assumption that a time series generated by the underlying model could be perfectly observed [3]. More recent approaches account for stochasticity of the system and noise in the data by performing model selection, e.g. based on the Minimum Description Length principle, and were shown to outperform purely deterministic models [4]. Algebraic expression models, as introduced by Laubenbacher et al. in 2005 [5], provide a generalization of Boolean networks. Laubenbacher et al. assume that expression levels can take values from a finite set that is given the structure of a finite field. They use tools from computer algebra to develop an algorithm that reconstructs the topology and the dynamics of the network assuming purely deterministic generated data. In this paper, we consider the easier problem of inferring the dynamics under the algebraic model when the network topology is given a priori. However, we describe a constructive approach for addressing the randomness and missing data issues in an algebraic framework making use of concepts developed in coding theory. We show that known decoding algorithms can be used to reverse engineer the dynamics of gene expression profiles from noisy data when the topology of the network is known, and there is no requirement for the logic to be Boolean. The remainder of the paper is organized as follows. Section 2 contains the description of algebraic GRN models, while Section 3 introduces the new reverse engineering framework. Section 4 presents performance results of decoding methods for synthetic data, and data for an emergency response control network of *E. coli*. In the same section, we also briefly address possible extensions of our work.

2. GENE NETWORKS AS POLYNOMIAL DYNAMICAL SYSTEMS

We formally define a gene regulatory network as a directed graph, $G = (V, E)$, in which the vertices $V = \{v_1, ..., v_n\}$ represent genes, while the edges in E describe regulatory relationships among genes. An edge (j, i) is drawn from gene v_j to gene v_i if gene v_j regulates the expression of gene v_i. Throughout the paper, it is assumed that the topology of the network, i.e. E, is known. Let $\breve{\boldsymbol{v}}_i = (v_{i_1}, .., v_{i_m})$, $i_1 < i_2 < .. < i_m$, $\forall v_{i_k} : (i_k, i) \in E$ be the vector of regulators of v_i. We consider discrete time, discrete value models, i.e. $t \in \mathbb{N}$, $v_i(t+1) \in \{0, 1, ...\}$. The expression of a gene v_i at time $t+1$ is determined by the expression pattern of its regulators at time t via the function f_i:

$$v_i(t+1) = f_i(\breve{\boldsymbol{v}}_i(t)). \tag{1}$$

Boolean network (BN) models allow genes to be in two different states - "ON" or "OFF". In this case, the functions $f_i : \mathbb{F}_2^n \to \mathbb{F}_2$, where $\mathbb{F}_2$ denotes the finite field of order two, describe a Boolean relationship among a gene and its regulators. The vector $\boldsymbol{v}(t) = (v_1(t), ..., v_n(t))$ denotes the state of the BN at time t. A global transition of the network is a pair of consecutive network states, $(\boldsymbol{v}(t), \boldsymbol{v}(t+1)) \in \mathbb{F}_2^n \times \mathbb{F}_2^n$. An alternative way to represent a transition of the network is to list the local transi-

tions at the nodes $(\breve{\boldsymbol{v}}_i(t), v_i(t+1)) \in \mathbb{F}_2^m \times \mathbb{F}_2$, $i = 1..n$. Clearly, a list of local transitions completely specifies a global transition and vice versa. In other words, a BN is a Boolean mapping

$$\mathcal{F} : \mathbb{F}_2^n \to \mathbb{F}_2^n, \boldsymbol{v}(t) \mapsto \boldsymbol{v}(t+1), \tag{2}$$

that can be decomposed into n different Boolean functions f_i. Recently, Laubenbacher et. al. introduced reverse engineering in an algebraic framework for modeling gene expression profiles which is a generalization of the two-state Boolean model [5]: a gene is allowed to take a finite number of states that represent different expression levels. The number of states, q, is chosen to be a power of a prime, and the state values are assumed to be elements of a finite field $\mathbb{F}_q$. Consequently, $v_i(t) \in \mathbb{F}_q$ and the resulting discrete network can be seen as a generalization of BNs, defined analogously to Eq. (2):

$$\mathcal{F}_q : \mathbb{F}_q^n \to \mathbb{F}_q^n, \boldsymbol{v}(t) \mapsto \boldsymbol{v}(t+1). \tag{3}$$

It is well known that over a finite field, any possible function $\mathbb{F}_q^n \to \mathbb{F}_q$ is an element of the polynomial ring $\mathbb{F}_q[x_1, x_2, ..., x_n]$, i.e. the set of all polynomials in variables $x_1, .., x_n$, and coefficients in $\mathbb{F}_q$. Hence, under this model, each node function f_i is a multivariate polynomial with the regulators $\breve{\boldsymbol{v}}_i$ of the gene as the variables. In a more general context, models of this form are also referred to as *polynomial dynamical systems* (PDS).

We describe next a class of algorithms for constructing PDS models for GRN under noisy conditions that use coding-theoretic ideas.

3. DECODING OF NOISY PDS

Early efforts on reverse engineering discrete models assumed a completely deterministic network and perfectly observed transitions[3]. As this is an unrealistic assumption, recent approaches for reverse engineering under the Boolean model account for randomness by means of model selection criteria [4]. No such method has yet been presented for the algebraic model. Even though the algorithm presented by Laubenbacher et al. selects a solution which is in a sense "minimal", it does not explicitly take noise into account and is therefore very sensitive to stochastic effects [6]. Here, we assume that, for each gene v_i, we are given time series data of transitions

$$\mathcal{V}_i = ((\breve{\boldsymbol{v}}_i(0), v_i(1)), .., (\breve{\boldsymbol{v}}_i(T-1), v_i(T))),$$

under the algebraic expression model, and we explicitly assume that the observations are noisy, i.e. $v_i(t+1) = f_i(\breve{\boldsymbol{v}}_i(t)) + \varepsilon_i(t)$ where $\varepsilon_i(t)$ is a random variable that can take values in $\mathbb{F}_q$ with nonzero probability $P(\varepsilon_i(t) \neq 0) = p_\varepsilon > 0$, $P(\varepsilon_i(t) = \ell) = (1-p_\varepsilon)/(q-1)$, $\ell \neq 0$. In the noiseless case ($p_\varepsilon = 0$), interpolating a polynomial through the observed points $\mathcal{V}_i$ will perfectly reconstruct the function f_i, provided enough transitions are available [7]. However, when $p_\varepsilon > 0$ is fairly large, interpolating through all transitions is obviously a bad strategy as one may expect to over-fit the data and reconstruct the wrong polynomial. In this case, one tractable solution is to perform specialized model selection and to find a solution that generates a time series that differs from the observed values in a fraction of points (approximation instead of interpolation - similar to a regression analysis). In what follows, we establish a connection of the approximation problem to coding theory and present an algorithm that efficiently solves this problem, provided certain conditions are met.

As shown above, we deal with approximating observations by functions f_i that are multivariate polynomials in $\mathbb{F}_q$. In Coding Theory, this is known as the problem of decoding q-ary Reed-Muller (RM) codes: let $f \in \mathbb{F}_q[x_1, ..., x_n]$, i.e.

$$f = \sum_{i_1,...,i_n} a_{i_1 i_2 ... i_n} x_1^{i_1} ... x_n^{i_n},$$

then the total degree of f, totdeg(f), is defined as

$$\text{totdeg}(f) = \max\{i_1 + i_2 + ... + i_n : a_{i_1 i_2 ... i_n} \neq 0\}.$$

A q-ary Reed-Muller (RM) code $\mathcal{RM}_q(u, m)$ is the set of all m-variate polynomials $f \in \mathbb{F}_q[x_1, .., x_m]$ of bounded total degree evaluated at q^m pairwise-distinct points $\alpha_k \in \mathbb{F}_{q^m}$. Formally,

$$\begin{aligned}\mathcal{RM}_q(u, m) =& \{(f(\alpha_1), ..., f(\alpha_{q^m})) : \\ & f \in \mathbb{F}_q[x_1, .., x_m], \text{totdeg}(f) \leq u\}. \end{aligned} \tag{4}$$

A code is used to encode messages allowing for their subsequent reconstruction from an observed noisy version of the codeword. The reconstruction process is called decoding. In an RM code, the encoded messages are multivariate polynomials of bounded degree. Given the noisy observations $(f(\alpha_1) + \varepsilon_1, .., f(\alpha_{q^m}) + \varepsilon_{q^m})$ an optimal RM decoder finds the polynomial f of bounded degree u that most likely led to the observation (Maximum Likelihood). However, this problem is NP hard and suboptimal decoders must be used. A set of powerful algorithms has been proposed in the coding literature that can closely approach optimal performance (see references in [6]).

We will shortly sketch our reverse engineering method and refer to [6] for a more detailed discussion: The number of regulators corresponds to the parameter m in Eq. (4). The message that is to be encoded in the codeword is the polynomial f_i. The input vector $\breve{\boldsymbol{v}}_i$ having m elements from the field $\mathbb{F}_q$ can be interpreted as an element from $\mathbb{F}_{q^m}$ and corresponds to the evaluation points α_k. In this framework, the genes' outputs $v_i(t+1)$ represent the codeword symbols $f(\alpha_k)$. The situation is depicted in Figure 1 as an example for a single gene with 3 regulators. We can then apply known RM decoding algorithms to find the approximating polynomials. Our model selection criteria here is to favor polynomials with low degree, this is done by exploring the solution space of polynomials of bounded degree u. We start with $u = 1$ and try to find a polynomial that approximates the time series using a decoding algorithm. If a solution is found, it is stored in the list $\mathcal{C}$, then we increase u by one and repeat the decoding step. The success of this procedure depends on the combined number of errors $|\{t : \varepsilon_i(t) \neq 0\}|$ and number of observations and the degree of the node function u. Bounds on the required number of transitions for unique reconstruction are easily derived by analyzing the properties of RM codes (for details see [7]). By increasing u to its maximum value, $\mathcal{C}$ will always contain the solution

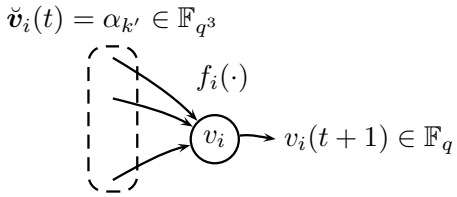


Figure 1. The function f_i is evaluated at the input vector resulting in $v_i(t+1)$ which can be regarded as the codewordsymbol of the RM-code arising from $\alpha_{k'}$.

that perfectly interpolates the time series. Note that our algorithm explicitly reconstructs the functions f_i meaning that it is able to predict outputs for input patterns $\breve{\boldsymbol{v}}_i$ that have not been observed in $\mathcal{V}_i$.

4. RESULTS

4.1. Synthetic Networks

We sampled 1000 PDS with random topologies, the nodes having in-degree 0 (30%), 2 (50%) or 3 (20%). Each time the different degrees u_i of the node polynomials f_i were randomly chosen from $u_i \in \{1, 2, 3, 4, 5\}$ with random coefficients from $\mathbb{F}_q$, $q = 5$ was used for the simulations. Nodes were initialized with random expression values and five transitions of the network were recorded. This was repeated 50 times, producing a set of 250 synthetic expression samples. Noise was added to the "measurements" by randomly replacing a fraction of values with different symbols. Simulations were implemented and run in MATLAB and the algorithm described in [8] was implemented for the decoding step. Figure 2 shows the percentage of correctly reconstructed node functions over an increasing noise level p_ε. Results show that reverse engineering is indeed possible in the presence of a significant noise component.

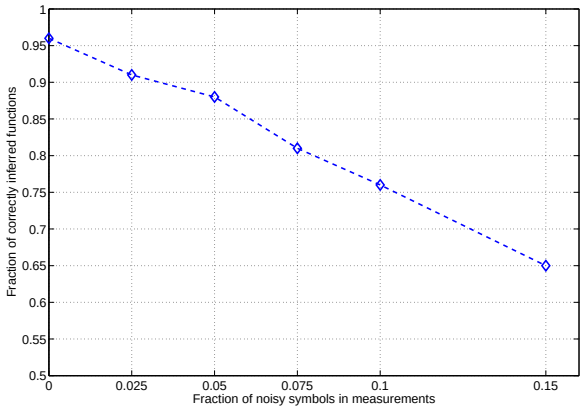


Figure 2. Performance on 1000 simulated 20-node PDS over $\mathbb{F}_5$ with random topologies and random node functions.

4.2. *E.Coli* SOS pathway

Microarray experiments are still very expensive, and often only extremely small datasets are available for a time-course analysis of expression profiles. The number of time points measured in such experiments is typically in the range $5 - 15$. Yet, the many small microarray datasets generated by different groups around the world represent a large resource for network inference. In order to facilitate the analysis of expression data compiled from multiple laboratories, Faith et al. compiled a unified microarray database for several microbe organisms [9]. This database contains only single-channel arrays using the same platform and the raw expression data is uniformly normalized to enable analysis across experiments without further user-dependent processing. We used the microarray data of *E.coli*, provided in this database, and filtered time course experiments that revealed at least one transition of the network. We found a total of 69 transitions obtained from 21 different experiments. Using the Lloyd-Max quantizer we discretized the data for each gene into q discrete expression levels. We considered the SOS pathway described in [10]. This pathway regulates cell survival and repair after DNA damage. Our "test network" comprises 9 genes including the principal mediators of the SOS response *lexA* and *recA* which are known to regulate many genes directly and tens or possibly hundreds indirectly [10]. Further, four genes (*ssb, recF, dinI and umuDC*) known to be involved in the SOS response are included as well as three sigma factor genes (*rpoD, rpoH* and *rpoS*) whose regulatory role in the SOS response is not fully understood [10]. The network is depicted in Figure 3.

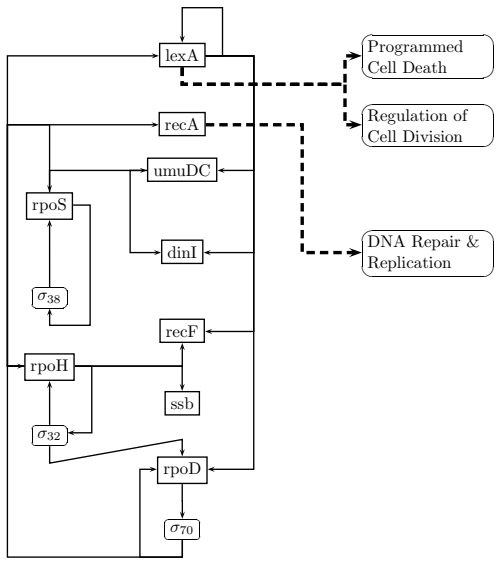


Figure 3. Diagram of interactions in the SOS network (as described in [10]). Angled boxes denote genes, small rounded boxes proteins.

We applied our algorithm as described above. For $q = 5$ we were able to reconstruct approximating functions of 3 of the genes (*lexA*, *rpoS*, *rpoH*). The input/output responses of these genes are depicted in Figure 4. Note

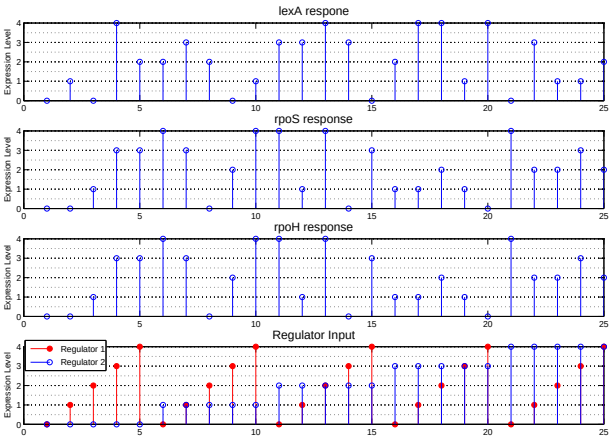


Figure 4. Inferred responses (rows 1-3) of the *lexA*, *rpoS*, *rpoH* to all 25 possible inputs of the regulators (last row).

that there are $5^2 = 25$ possible input patterns for each of these genes, shown in the last row, and a single quinternary response per input (shown in rows 1-3 for the different genes). Interestingly, *rpoS* and *rpoH* exhibit identical responses; both are regulated by *rpoD* and a direct self-loop, which may explain this observation. All found polynomials are linear $u = 1$, indicating that non-linear functions could not be found with the provided number of samples and quality of data. In an attempt to assess the statistical significance of our results, we randomly sampled 9 genes from the available 4292 in our dataset and applied the reverse engineering using their expression profiles, assuming the same topology as in Figure 3. Functions found in this way should be counted as false positives. This experiment was repeated 1000 times and the number of non-empty lists was counted. Only in 6 out of the 1000 times, 3 nodes were simultaneously inferred in the same network. The overall observed false positive rate per node was less than 3.8%. In order to find lists of possible update functions for more complex networks or to study organisms with more uncertainty in network topology, one needs more high quality microarray data of transition measurements. The functions inferred through our algorithm could then be used to simulate the response of the gene network to different perturbations and to propose treatments for certain diseases.

4.3. Extensions

The proposed list-decoding approach can be improved and extended in the following directions.

1. More efficient decoding algorithms for Reed-Muller codes can be used to further reduce the number of required observed transitions and exploit statistics of the observed time series as side information.

2. Genomic update functions are not arbitrary, e.g. rules have been found to be often canalizing. In addition, a given transcription factor is usually unidirectional, meaning that most of its regulatees are either all down-regulated or up-regulated. Incorporating this feature into the model refinement process may further improve the inference potential of our method.

3. The influence of quantization on network dynamics inference is still not well understood, and quantization techniques that take into account error models for DNA microarray measurements have to be considered in this setting.

Acknowledgments

Janis Dingel gratefully acknowledges financial support of the German Academic Exchange Service and the Deutsche Forschungsgemeinschaft (HA 1358/11-1).

5. REFERENCES

[1] T. Schlitt and A. Brazma, "Current approaches to gene regulatory network modelling.," *BMC Bioinformatics*, vol. 8 Suppl 6, pp. S9, 2007.

[2] I. Shmulevich, E. R. Dougherty, S. Kim, and W. Zhang, "Probabilistic boolean networks: a rule-based uncertainty model for gene regulatory networks.," *Bioinformatics*, vol. 18, no. 2, pp. 261–74, February 2002.

[3] S. Liang, S. Fuhrman, and R. Somogyi, "Reveal, a general reverse engineering algorithm for inference of genetic network architectures.," *Pac Symp Biocomput*, pp. 18–29, 1998.

[4] J. Dougherty, I. Tabus, and J. Astola, "Inference of gene regulatory networks based on a universal minimum description length," *EURASIP Journal on Bioinformatics and Systems Biology*, vol. 2008, 2008, Article ID 482090, 11 pages.

[5] R. Laubenbacher and B. Stigler, "A computational algebra approach to the reverse engineering of gene regulatory networks.," *J Theor Biol*, vol. 229, no. 4, pp. 523–37, August 2004.

[6] J. Dingel and O. Milenkovic, "A list-decoding approach for inferring the dynamics of gene regulatory networks," in *Proc. IEEE International Symposium on Information Theory, ISIT08*, July 2008, (to appear).

[7] J. Dingel and O. Milenkovic, "Coding theoretic methods for reverse engineering of gene regulatory networks," in *Proc. of the IEEE Information Theory Workshop 2008 (to appear)*, May 2008.

[8] R. Pellikaan and X.-W. Wu, "List decoding of q-ary Reed-Muller codes," *IEEE Transactions on Information Theory*, vol. 50, no. 4, pp. 679–682, April 2004.

[9] J. J. Faith, M. E. Driscoll, V. A. Fusaro, and et. al., "Many Microbe Microarrays database: uniformly normalized affymetrix compendia with structured experimental metadata.," *Nucleic Acids Res*, October 2007.

[10] T. S. Gardner, D. di Bernardo, D. Lorenz, and J. J. Collins, "Inferring genetic networks and identifying compound mode of action via expression profiling.," *Science*, vol. 301, no. 5629, pp. 102–5, July 2003.

TESTING FOR DIFFERENTIAL EXPRESSION IN SIMULATED AND REAL CDNA MICROARRAY DATA USING FREQUENTIST AND BAYESIAN METHODS

Timo Erkkilä[*]*, Matti Nykter, Harri Lähdesmäki, Miika Ahdesmäki, and Olli Yli-Harja*

Department of Signal Processing, Tampere University of Technology,
P.O. Box 553, FI-33101 Tampere, Finland
[*]timo.p.erkkila@tut.fi

ABSTRACT

In this paper, we test and compare different data models for finding differential expression in cDNA microarray measurements. We use Bayesian hierarchical error model (HEM) and its variants that are derived by changing the functional form of the original HEM variance. In addition to heterogeneous variance, we use the HEM with exponential and constant variance functions. The standard t-test for finding differential expression is our reference test. For both approaches, false discovery rates (FDR) are estimated. With data simulations, we test the accuracy of variance models and FDR estimators. The fit of exponential variance function to real data is observed as well. The parameters for the Bayesian models are estimated using Gibbs sampling.

1. INTRODUCTION

For different microarray technologies, data models that try to explain sources of variation of measured expression have been proposed (see [1] and [2]). One usual set-up for microarray measurements is to measure gene expression profiles of two or more biological samples under different conditions, or expression profiles of two or more different cell lines. The differences between the biological samples are then assumed to be characterized by the measured expression profiles that represent expressions proportional to the underlying mRNA concentrations [3].

Biological and technical variations are the primary sources of variation in the measured data. Other sources, *e.g.* , environmental conditions, degradation of mRNA [4], and labeling of samples [5], also have a significant role. Thus, without taking the high amount of uncertainty into account, one may not be able to accurately identify, say, differences between conditions, while increasing FDR. Throughout the study, we (a) try to take the nature of variation in the measurements into account, and (b) estimate FDR in finding differentially expressed genes.

In Section 2, we introduce the HEM variants, and give the formulas for calculating FDR estimates. In Sections 2.1 and 2.2, we introduce the used data simulation methods, prior distributions, and Gibbs sampling.

2. METHODS

In this study, we assume that there are no missing values in the data. We can therefore use the following labeling for our data sets: $i \in \{1, \ldots, I\}$ corresponds to gene index, $j \in \{1, 2\}$ corresponds to biological condition, and the replicates are denoted as $k \in \{1, \ldots, K\}$.

Before using the expression data, we transform it into $\log_2$-domain [3]. This is done for two reasons: the data may contain multiplicative biases for different microarray slides and, more importantly, the models we use assume data to contain $\log$-normally distributed components.

We fit the HEM to cDNA microarray expressions of 2-color cDNA microarrays; one color channel is for a biological sample under condition $j = 1$, and the other channel for a biological sample under condition $j = 2$. When biological replicates are missing from the experiment, *i.e.* , when only technical replicates are available, the HEM takes the form

$$y_{ijk} = x_{ij} + e_{ijk} \sim N(x_{ij}, \sigma_{ij}^2) \tag{1}$$

where

$$x_{ij} = \mu + g_i + c_j + r_{ij}. \tag{2}$$

In Eq. 1, y_{ijk} is the observed data and in Eq. 2, μ is the grand mean over all slides, g_i is the gene effect, c_j is the condition effect, and r_{ij} is the interaction effect of gene i and condition j. The term e_{ijk} models the error of the whole experiment process. Thus, the model is similar to the standard 2-way ANOVA, except that HEM uses prior knowledge for estimating unknown parameters and does not, in general, assume constant variance. Different ways to stabilize the variation of expression values have been proposed [6], but one may also give the variance a functional form; in this study, we have used the following functions:

$$\sigma^2(x_{ij}) = \begin{cases} \sigma_{ij}^2 & ,\text{heterogeneous} \\ a^2 + be^{-cx_{ij}} & ,a^2, b, c, x_{ij} > 0 \\ a^2 & ,b, c = 0 \end{cases} \tag{3}$$

where the word *heterogeneous* refers to the original HEM. In the exponential function, x_{ij} is the true expression of the gene i and condition j, and the variance is assumed to

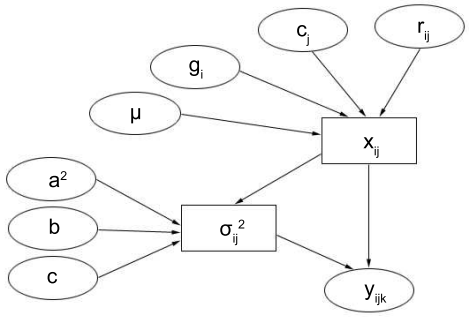


Figure 1. Model graph with functional variance. Ellipses represent stochastic nodes and rectangles represent function nodes. For all the ellipses, a prior distribution is assigned.

be intensity-dependent. The parameters a^2,b, and c need to be estimated from the data, and if b is negligible compared to a^2, or if c is close to zero, the variance shrinks to constant, which is the special case of the model. The assumption of functional variance complicates the dependency graph of parameters in the model, which can be seen in Fig. 1. Without the functional variance, the variance node itself would be stochastic, and no direct relationship to x_{ij} would exist. The edge between the variance and true expression, in fact, makes the model unhierarchical. We simplify the dependency by approximating the relationship using sample mean over the replicates k

$$\overline{x}_{ij} = \frac{1}{K}\sum_{k=1}^{K} y_{ijk} \tag{4}$$

instead of x_{ij}. The parameters of the HEM and its variants can be solved using, for instance, Gibbs sampling (see 2.2 for further information).

After Bayesian parameter estimation for the models, we use hypothesis testing methods to find differential expression in the data set. For HEM models, we use the H-score [7], a modified version of F-statistic, and for the t-test, we use p-values. The H-score for gene i is

$$H_i = \frac{1}{2}\sum_{j=1}^{2} \frac{(\hat{x}_{ij} - \hat{x}_{i\cdot})^2}{\hat{\sigma}^2(\hat{x}_{ij})} \tag{5}$$

where the hat over the letter denotes a Bayesian posterior mean parameter estimate, and the dotted subscript denotes averaging over that index. Since no null data is available, *i.e.* , a reference data with no differential expression from which to compute null H-scores is missing, we simulate such data by permuting the original data set, so that indices $i \in \{1, \ldots, I\}$ are preserved. H-scores of the null data set is computed with

$$H^0_{ip} = \frac{1}{2}\sum_{j=1}^{2} \frac{(\overline{x}_{ijp} - \overline{x}_{i\cdot p})^2}{\hat{\sigma}^2(\overline{x}_{ijp})} \tag{6}$$

where p is the permutation index, $p \in \{1, \ldots, P\}$, and

$$\hat{\sigma}^2(\overline{x}_{ijp}) = \begin{cases} \overline{\sigma}^2_{ijp} & ,\text{heterogeneous} \\ \hat{a}^2 + \hat{b}e^{-\hat{c}\overline{x}_{ijp}} & ,\text{exponential} \\ \hat{a}^2 & ,\text{constant} \end{cases} . \tag{7}$$

We use $P = 100$ permutations, and for each permutation, we use the standard sample estimators to calculate $\overline{x}_{ijp}$ (and $\overline{\sigma}^2_{ijp}$, if we are assuming heterogeneous variance), as Bayesian sampling after each permutation would drastically increase the computation time. It is noteworthy, that we have modified the H_0-score calculation in Eq. 6 to take into account the functional forms for variance. for both the H-score and H_0-score, it is crucial to use similar variance estimators, to reduce FDR estimator bias. See Fig. 3(a) for illustration of estimation bias: the actual scores are calculated using the assumed functional model for the variance, whereas the null scores are calculated using a sample variance estimator for the permuted data set. The FDR estimators for the HEM variants and the t-test (as proposed in the R implementation document of HEM and in [8], respectively) are

$$\widehat{FDR}_{HEM}(H_j) = \frac{\hat{\pi}_0 R^0(H_j)}{R(H_j)} \tag{8}$$

and

$$\widehat{FDR}_T(p_j) = \frac{\hat{\pi}_0 S^0(p_j)}{S(p_j)} \tag{9}$$

where

$$\begin{array}{l} R^0(H_j) = \frac{1}{P}\sum_{p=1}^{P} \#_i\{H^0_{ip} : H^0_{ip} > H_j\} \\ R(H_j) = \#_i\{H_i : H_i > H_j\} \\ S^0(p_j) = Ip_j \\ S(p_j) = \#_i\{p_i : p_i < p_j\} \end{array} . \tag{10}$$

The $\#_i$ denotes the number of values, that fulfill the terms inside the braces for $i \in \{1, \ldots, I\}$. The point estimates p_{λ_n} of π_0 are also calculated as in [7] and [8] using the percentiles $\lambda_n = 0.01n, n \in \{1, \ldots, 100\}$, but the estimator for π_0, the proportion of non-significant genes, is calculated using weighted average. We use the cumulative distribution function of $N(0.1, 0.3)$ to generate weights for each percentile λ_n:

$$c_{\lambda_n} = \Phi\left(\frac{\lambda_n - 0.1}{0.3}\right), \quad n \in \{1, \ldots, 100\} \tag{11}$$

The weight matrix is a diagonal matrix $C = \text{diag}\,(c_{\lambda_1}, \ldots, c_{\lambda_{100}})$, and $\mathbf{p} = [p_{\lambda_1}, \ldots, p_{\lambda_{100}}]^T$. The estimator $\hat{\pi}_0$ is therefore

$$\hat{\pi}_0 = \left(\mathbf{1}^T C \mathbf{1}\right)^{-1} \mathbf{1}^T C \mathbf{p}. \tag{12}$$

The reason for using C, that gives more weight as n increases, is to compensate the bias and variance of each point estimate; when n increases, the bias of point estimate p_{λ_n} decreases, whereas the variance increases [8].

2.1. Simulations

In the simulation study, we generate cDNA microarray data with outliers using methods proposed in [9]. The data consists of $I = 5000$ genes, $J = 2$ conditions, and $K = 10$ replicates. The distributions of the simulator are

$$\begin{array}{rl} \forall i: & z_i \sim Exp(\lambda') \\ \forall i: & o_i \sim Ber(1-\pi_0) \\ \forall o_i = 1: & s_i \sim Rademacher \\ \forall o_i = 1: & b_i \sim Beta(\alpha', \beta') \\ \forall i,j,k: & y_{ijk} \sim N(x_{ij}, \sigma^2(x_{ij})) \end{array} , \quad (13)$$

the functions of the simulator are

$$\begin{array}{rl} \forall o_i = 1: & \sqrt{t_i} = 10^{s_i b_i} \\ \forall o_i = 1: & z_{i1} = z_i \sqrt{t_i} \\ \forall o_i = 1: & z_{i2} = z_i / \sqrt{t_i} \\ \forall o_i = 0: & z_{i1} = z_{i2} = z_i \\ \forall i,j: & x_{ij} = \log_2(z_{ij}) \\ \forall i,j: & \sigma^2(x_{ij}) = a^2 + be^{-cx_{ij}} \end{array} , \quad (14)$$

and the parameters for the functions and distributions are set to

$$\begin{array}{c} \lambda' = 1000, \pi_0 = 0.96, \\ \alpha' = 1.7, \beta' = 4.8 \\ a^2 = 0.2, b = 1.0, c = 0.4. \end{array} . \quad (15)$$

So, the simulation of measurements in short: Generate I measurements from an exponential distribution. With probability $1 - \pi_0$, a measurement i is assigned as differentially expressed. With probability 0.5 it is an overexpression, and t is the shifting value between the conditions $j = 1$ and $j = 2$. The expressions are the $\log_2$-transformed, and variance is generated from the exponential function, using the $\log_2$-transformed measurements. Finally, for each replicate k, normally distributed noise is added.

2.2. Prior specification and Gibbs sampling

We have built the Gibbs samplers with WinBUGS (Bayesian inference Using Gibbs Sampling for Windows) [10]. The software is suitable for generating Gibbs samplers for models, where the parameter dependencies form a directed acyclic graph (DAG). WinBUGS can be downloaded from `http://www.mrc-bsu.cam.ac.uk/bugs/`.

The used prior and hyperprior distributions for all parameters are tabulated in Table 1. The priors are on the left-hand side and the used parameters for the distributions are on the right-hand side. The chosen parameter and distribution values are similar as in [7]. We use Gibbs sampling posterior mean to calculate the estimates for each parameter in the model, we generate a 600-point sample after a 300-point burn-in period. We noticed, that in this study such amount of iterations is sufficient for the Markov chains to converge.

$\mu \sim U(0, \mu_{max})$	$\mu_{max} = 50$
$g_i \sim N(0, \sigma_g^2)$	$\sigma_g = 1$
$c_j \sim N(0, \sigma_c^2)$	$\sigma_c = 1$
$r_{ij} \sim N(0, \sigma_r^2)$	$\sigma_r = 1$
$e_{ijk} \sim N(0, \sigma^2(x_{ij}))$	Eq. 3
$\sigma_{ij}^{-2}, a^{-2} \sim \Gamma(\alpha, \beta)$	$\alpha = 1, \beta = 0.125$
$b, c \sim U(0, t_{max})$	$t_{max} = 5$

Table 1. Prior and hyperprior distributions for the HEM.

3. RESULTS AND CONCLUSION

The simulation study consisted of data generation with known parameters, Bayesian and frequentist parameter estimation, visualization of FDR estimation accuracy for all models (Fig. 3(a) and 3(b)), and visualization of accuracy (ROC curves) for finding differential expression. The simulations show, that if such exponential variance structure exists, the functional form of variance in HEM can be modified to better fit the data (Fig. 2), thus resulting in more accurate differential expression detection (Fig. 3(c)). The approximation of dependency between the variance function and true expression could reduce the accuracy of the variance fit drastically, if the amount of replicates was small. Also, after each perumation for calculating the H_0-score, the using of Bayesian estimates instead of sample estimates would increse the performance of FDR estimation for the HEM variants.

4. ACKNOWLEDGEMENTS

This work was supported by the Academy of Finland, (application number 213462, Finnish Programme for Centres of Excellence in Research 2006-2011).

5. REFERENCES

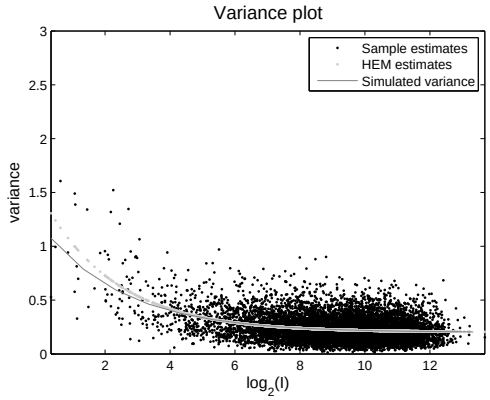
[1] A. Lewin, S. Richardson, C. Marshall, A. Glazier, and T. Aitman, "Bayesian modeling of differential gene expression.," *Biometrics*, vol. 62, no. 1, pp. 1–9, Mar 2006.

[2] K. Lo and R. Gottardo, "Flexible empirical bayes models for differential gene expression.," *Bioinformatics*, vol. 23, no. 3, pp. 328–335, Feb 2007.

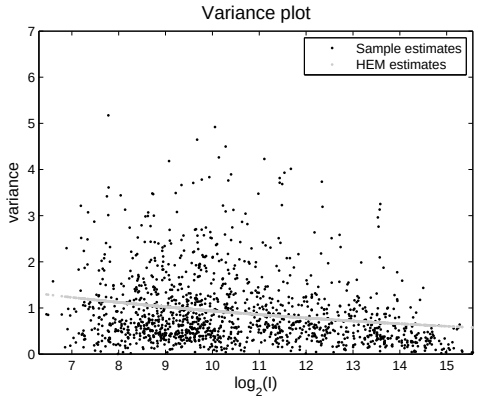
[3] J. Quackenbush, "Microarray data normalization and transformation.," *Nat Genet*, vol. 32 Suppl, pp. 496–501, Dec 2002.

[4] H. Auer, S. Lyianarachchi, D. Newsom, M. I. Klisovic, G. Marcucci, U. Marcucci, and K. Kornacker, "Chipping away at the chip bias: Rna degradation in microarray analysis.," *Nat Genet*, vol. 35, no. 4, pp. 292–293, Dec 2003.

[5] K. K. Dobbin, E. S. Kawasaki, D. W. Petersen, and R. M. Simon, "Characterizing dye bias in microarray experiments.," *Bioinformatics*, vol. 21, no. 10, pp. 2430–2437, May 2005.



(a) Exponential variance fit using simulated data. The solid grey line is the simulated variance, black dots are the sample variance estimates, and light gray dots are the HEM estimates.



(b) Exponential variance fit using 4 replicates of E-MEXP-1385 *mus musculus* data. The black dots are sample variance estimates, and the light gray dots are the HEM estimates.

Figure 2. Variance plots as functions of intensity.

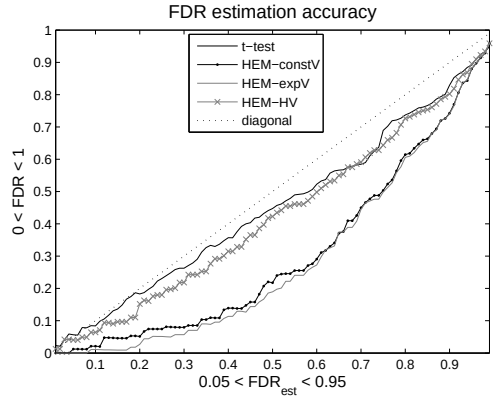
[6] W. Huber, A. von Heydebreck, H. Sültmann, A. Poustka, and M. Vingron, "Variance stabilization applied to microarray data calibration and to the quantification of differential expression.," *Bioinformatics*, vol. 18 Suppl 1, pp. S96–104, 2002.

[7] H. J. Cho and J. K. Lee, "Bayesian hierarchical error model for analysis of gene expression data.," *Bioinformatics*, vol. 20, no. 13, pp. 2016–2025, Sep 2004.

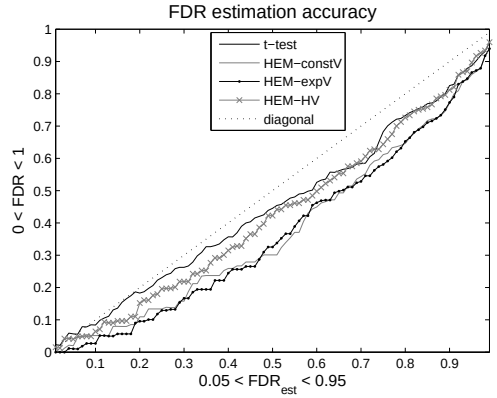
[8] J. D. Storey and R. Tibshirani, "Statistical significance for genomewide studies.," *Proc Natl Acad Sci U S A*, vol. 100, no. 16, pp. 9440–9445, Aug 2003.

[9] Y. Balagurunathan, E. R. Dougherty, Y. Chen, M. L. Bittner, and J. M. Trent, "Simulation of cdna microarrays via a parameterized random signal model.," *J Biomed Opt*, vol. 7, no. 3, pp. 507–523, Jul 2002.

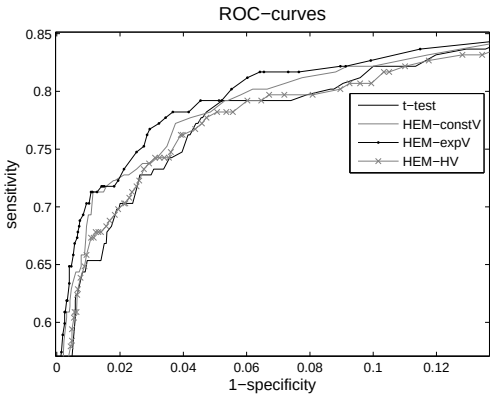
[10] D. J. Lunn, A. Thomas, N. Best, and D. Spiegelhalter, "Winbugs - a bayesian modelling framework: Concepts, structure, and extensibility," *Statistics and Computing*, vol. 10, pp. 325–337, 2000.



(a) FDR estimation accuracy using the H_0-score without variance correction; the FDR estimations of HEM models with functional variance perform poorly.



(b) Variance functionality taken into account; the bias is reduced.



(c) Increasing the accuracy of variance estimation increases the accuracy of finding differentially expressed genes. The HEM with exponential variance function performs somewhat better than the other models.

Figure 3. FDR estimation accuracy plots and ROC curves.

TOWARDS SYSTEMS BIOLOGY OF DEVELOPING BARLEY GRAINS: A FRAMEWORK FOR MODELING METABOLISM

E. Grafahrend-Belau [1], *B. H. Junker* [1], *D. Koschützki* [1,2], *C. Klukas* [1], *S. Weise* [1], *U. Scholz* [1] *and F. Schreiber* [1,3]

[1] Leibniz Institute of Plant Genetics and Crop Plant Research (IPK), Gatersleben, Germany
[2] Furtwangen University of Applied Sciences, Furtwangen, Germany
[3] Martin-Luther-University Halle-Wittenberg, Halle (Saale), Germany
grafahr@ipk-gatersleben.de, junker@ipk-gatersleben.de, dirk.koschuetzki@hs-furtwangen.de, klukas@ipk-gatersleben.de, weise@ipk-gatersleben.de, scholz@ipk-gatersleben.de, schreibe@ipk-gatersleben.de

ABSTRACT

Modeling of plant metabolism offers new approaches to improve the understanding of complex biological processes. In this paper we present a framework for the constraint-based stoichiometric analysis of crop plant metabolic models, which combines tools for (1) model reconstruction, (2) model analysis (i.e. flux balance analysis), and (3) model visualization. The approach is illustrated by describing its application to the modeling of cereal seed metabolism.

1. INTRODUCTION

Crop plants are the main source for feed and food and important contributors to chemical feedstocks and renewable fuels [1, 2, 3]. A detailed understanding of crop plant metabolism is necessary to improve their growth and yield [4, 5]. Mathematical modeling of metabolism offers new approaches to analyze the structure, dynamics and behavior of complex metabolic pathways. Metabolic models can be used to verify and extend the understanding of complex processes, to generate or test hypotheses and to explore *in silico* scenarios, thus allowing to identify suitable targets for metabolic engineering. The issue of mathematical modeling in plant metabolism is constantly gaining attention and several mathematical modeling approaches applied to plant metabolism exist [6, 7, 8].

Flux balance analysis (FBA) is a constraint-based modeling approach which can be used to predict metabolic capabilities and flux distributions under different environmental conditions. FBA has the advantage that is does not require the knowledge of kinetic parameters, instead only the stoichiometry of the metabolic network has to be known, and an objective function is needed to identify the optimal flux distribution among all possible steady state flux distributions. Although metabolic flux determination is acknowledged to be an important part of plant metabolic engineering [9], to the best of our knowledge FBA has not yet been applied to plant metabolic systems.

This paper describes in the first part a framework for the constraint-based stoichiometric analysis of crop plant metabolic models comprising the following steps of the modeling approach: (1) model reconstruction, management and export, (2) model simulation and analysis (FBA), and (3) model and flux visualization. Focusing on the tools and methods developed, each of the modeling steps is described and in the second part of the paper the application of the proposed framework is shown by a case study of storage metabolism in developing barley seeds.

2. METHODS

The workflow used for the constraint-based stoichiometric analysis of crop plant metabolic models is summarized in Figure 1. Each step of the workflow is described below:

2.1. Model reconstruction

The reconstruction of plant metabolic models requires detailed metabolic information. To facilitate the modelling of crop plant metabolic models, we developed MetaCrop (http://metacrop.ipk-gatersleben.de) [10], a manually curated database for crop plant metabolism. MetaCrop provides manually curated, detailed information about metabolic pathways in six major crop plants, including pathway diagramms, locations, transport processes, reaction kinetics and literature. Scientists working in the area of plant research can use the data of MetaCrop, thus accelerating the process of data curation. Moreover, new pathways can be created by combining existing information or by adding new data.

MetaCrop is based on the Meta-All software [11]. Both conversions and substances play a central role in MetaCrop. A conversion should be understood as a reaction or a translocation, which can be either an active or a passive process. Substances can be assigned to conversions and play certain roles within these processes, such as substrate, product, catalyst or inhibitor. Conversions can be combined to pathways and pathways to super-pathways, thus enabling the successive reconstruction of metabolic models.

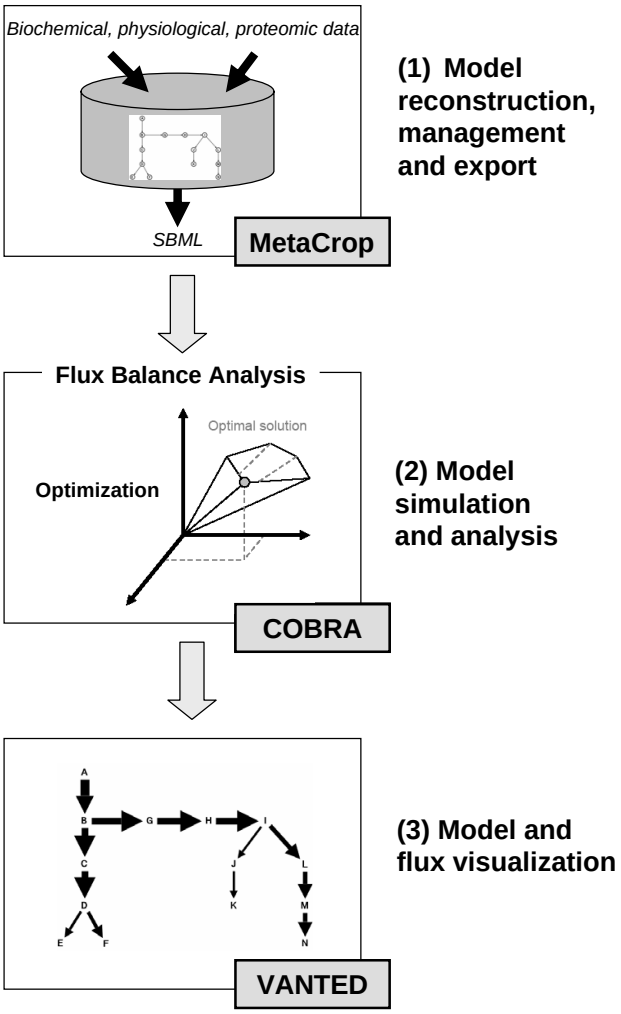


Figure 1. General workflow used for constraint-based stoichiometric analysis

All elements stored in MetaCrop can be enriched by fine-grained information. This includes kinetic data such as affinity or inhibitor constants, reaction or translocation type, formula, Enzyme Commission (EC) numbers and literature references. Furthermore, MetaCrop enables the user to store different parallel versions of pathways to represent different models and also because frequently ambiguities in biochemical data exist. In order to interact with simulation and analysis tools, models reconstructed in MetaCrop can be exported using the standardized Systems Biology Markup Language (SBML) format.

2.2. Model analysis

Flux balance analysis is a constraint-based modeling approach developed to characterize systemic properties of a metabolic network based on mass balance constraints [12]. Stoichiometric metabolic models are usually underdetermined systems of linear algebraic equations. FBA uses the principle of linear programming to solve the equations by defining an objective function and searching the solution space for an optimum flux distribution that meets the objective.

For growth simulations of crop plant metabolic models, the growth rate (i.e. the rate of biomass synthesis) was selected as the objective function to be maximized. The growth objective is mathematically defined as a flux drain comprised of all biosynthetic precursors and cofactors (e.g. amino acids, ATP) required for biomass production. Similar approaches have been proposed and proven useful in predicting *in vivo* cellular behavior of different biological systems [13, 14].

Flux balance analysis is performed using the COBRA toolbox (http://systemsbiology.ucsd.edu/downloads/COBRAToolbox/) [15], a constraint-based reconstruction and analysis toolbox running in the MATLAB environment. The toolbox has the advantage (1) to support SBML-import, thus allowing to import crop plant metabolic models reconstructed in MetaCrop, and (2) to be extendable by user-written MATLAB routines, thus allowing to incorporate functionalities not offered by the system. In order to interact with the visualization tool, analysis results obtained in COBRA can be exported as text- or csv-format.

2.3. Model and flux visualization

Visualization can improve the understanding of complex processes such as the structure of metabolic models and the results of their analysis. VANTED (visualization and analysis of networks containing experimental data) (http://vanted.ipk-gatersleben.de) is a platform-independent software system which enables researchers to evaluate extensive data from genomics, proteomics and metabolomics. In order to support the analysis and visualization of metabolic flux data, the data-mapping methods introduced in [16] and refined in [17], have been extended to support the assignment of experimental and computed datasets to network edges.

In addition to previously available data visualization approaches utilizing line-, bar- and pie-charts, it is now possible to map data onto graphical attributes such as edge width or arrow shape. For the visualization of fluxes we map the metabolic flux to the width of the reaction edge. This visualization approach supports a fast understanding of the fluxes in both overview and detail, see Figure 2.

3. CASE STUDY: A MODEL OF CEREAL SEED METABOLISM

3.1. Model reconstruction

The metabolic network of central metabolism in the developing endosperm of barley (*Hordeum vulgare*) was reconstructed with the aim of giving insight into cereal seed metabolism during starch accumulation. The information necessary for network reconstruction (biochemical, physiological, and proteomic data) was collected through an extensive survey of scientific literature (e.g. [18, 19, 20, 21]) and online databases (e.g. [22, 23, 24, 25]), where data has been additionally checked against literature. The data was integrated into MetaCrop and the model was reconstructed in a stepwise manner.

The resulting compartmented stoichiometric model includes central metabolism (glycolysis, pentose phosphate

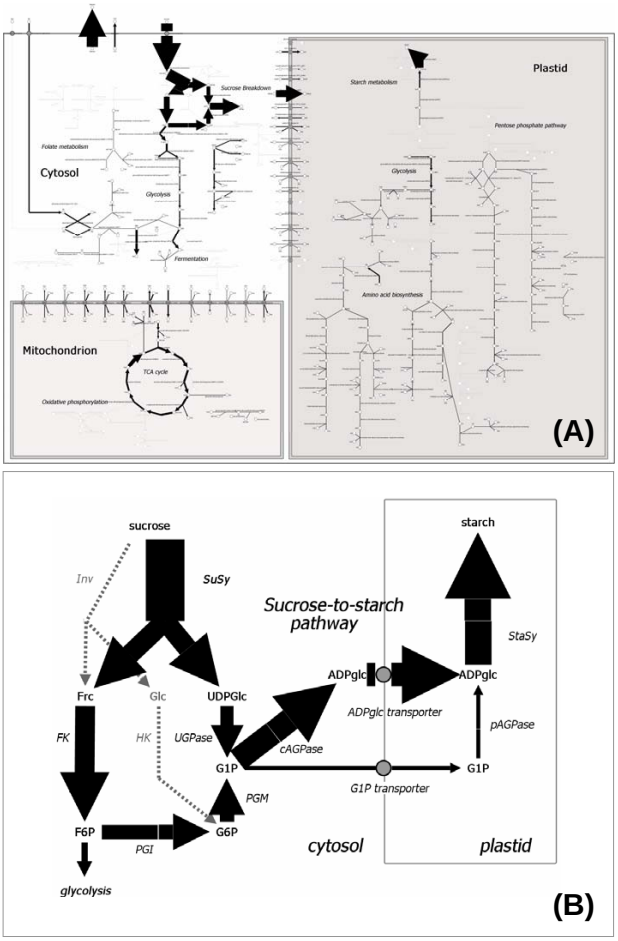


Figure 2. Carbon flux map of the reconstructed model of cereal seed metabolism under optimal growth conditions. (A) Complete model; (B) Sucrose-to-starch pathway.

pathway, citrate cycle), amino acid metabolism, starch synthesis, and some minor pathways. It comprises 234 metabolites and 257 reactions, of which 193 represent biochemical conversions and 64 represent transport processes compartmentalized between the extracellular medium and the intracellular compartments cytosol, mitochondria and plastid.

3.2. Model analysis and visualization

To elucidate the metabolic capabilities of barley grains concerning biomass production and to get insight into the underlying metabolic flux distribution, the FBA approach described in Section 2.2 was applied to the reconstructed model using the COBRA toolbox. Parameters necessary to perform FBA (maximum uptake and excretion rates, biomass composition, maintenance requirements) were determined based on published experimental results [26, 27, 28] and simulations under optimal growth conditions were run. Optimal growth was simulated by constraining the maximum sucrose uptake rate only. Based on the resulting flux vector, the carbon flux distribution was computed and the resulting flux map was visualized using VANTED, see Figure 2.

In general, the simulation results were found to be in agreement with the main biochemical properties of barley seed storage metabolism: the simulated growth rate, $\mu = 0{,}003$ h^{-1}, was in the range of experimental observations [29, 30], and the metabolic pathway pattern predicted by the model was in accordance with literature-based findings. As shown in Figure 2(B) the model correctly reproduces the sucrose-to-starch pathway reported from barley seed metabolism [31, 32] by predicting that (1) sucrose degradation is restricted to the sucrose synthase (SuSy) pathway and (2) synthesis of ADPglucose (ADPglc), which is the main precursor for starch synthesis, is predominantly catalyzed by the cytosolic isoform of ADPglucose pyrophosphorylase (cAGPase).

These results indicate that the reconstructed model has the potential to simulate cereal seed metabolism. Thus, by providing an initial template for studying seed metabolic behavior, the model can be used to generate or test hypothesis and to explore cereal seed metabolism *in silico*.

ACKNOWLEDGMENTS

This work was partly supported by the German Ministry of Education and Research (BMBF) under grants 031270-6A and 0315044A.

4. REFERENCES

[1] M. A. Grusak and D. DellaPenna, "Improving the nutrient composition of plants to enhance human nutrition and health," *Annu Rev Plant Physiol Plant Mol Biol*, vol. 50, pp. 133–161, 1999.

[2] J. O. Metzger and U. Bornscheuer, "Lipids as renewable resources: current state of chemical and biotechnological conversion and diversification," *Appl Microbiol Biotechnol*, vol. 71, no. 1, pp. 13–22, 2006.

[3] D. Tilman, J. Hill, and C. Lehman, "Carbon-negative biofuels from low-input high-diversity grassland biomass," *Science*, vol. 314, no. 5805, pp. 1598–1600, 2006.

[4] H. L. Jenner, "Transgenesis and yield: what are our targets?," *Trends Biotechnol*, vol. 21, no. 5, pp. 190–192, 2003.

[5] F. Carrari, E. Urbanczyk-Wochniak, L. Willmitzer, and A. R. Fernie, "Engineering central metabolism in crop species: learning the system," *Metab Eng*, vol. 5, no. 3, pp. 191–200, 2003.

[6] C. Giersch, "Mathematical modelling of metabolism," *Curr Opin Plant Biol*, vol. 3, no. 3, pp. 249–253, 2000.

[7] J. A. Morgan and D. Rhodes, "Mathematical modeling of plant metabolic pathways," *Metab Eng*, vol. 4, no. 1, pp. 80–89, 2002.

[8] M. G. Poolman, H. E. Assmus, and D. A. Fell, "Applications of metabolic modelling to plant

metabolism," *J Exp Bot*, vol. 55, no. 400, pp. 1177–1186, 2004.

[9] A. R. Fernie, P. Geigenberger, and M. Stitt, "Flux an important, but neglected, component of functional genomics," *Curr Opin Plant Biol*, vol. 8, no. 2, pp. 174–182, 2005.

[10] E. Grafahrend-Belau, S. Weise, D. Koschützki, U. Scholz, B. Junker, and F. Schreiber, "MetaCrop: a detailed database of crop plant metabolism," *Nucl Acids Res*, vol. 36, pp. D954–D958, 2008.

[11] S. Weise, I. Grosse, C. Klukas, D. Koschützki, U. Scholz, F. Schreiber, and B. H. Junker, "MetaAll: a system for managing metabolic pathway information," *BMC Bioinformatics*, vol. 7, pp. e465, 2006.

[12] J. S. Edwards, R. Ramakrishna, C. H. Schilling, and B. Ø. Palsson, "Metabolic Flux Balance Analysis," in *Metabolic Engineering*, S. Y. Lee and E. T. Papoutsakis, Eds., New York, 1999, pp. 13–57, Marcel Deker.

[13] J. S. Edwards, R. U. Ibarra, and B. Ø. Palsson, "In silico predictions of *Escherichia coli* metabolic capabilities are consistent with experimental data," *Nat Biotechnol*, vol. 19, no. 2, pp. 125–130, 2001.

[14] I. Famili, J. Forster, J. Nielsen, and B. Ø. Palsson, "*Saccharomyces cerevisiae* phenotypes can be predicted by using constraint-based analysis of a genome-scale reconstructed metabolic network," *Proc Natl Acad Sci U S A*, vol. 100, no. 23, pp. 13134–13139, 2003.

[15] S. A. Becker, A. M. Feist, M. L. Mo, G. Hannum, B. Ø. Palsson, and M. J. Herrgard, "Quantitative prediction of cellular metabolism with constraint-based models: the COBRA Toolbox," *Nat Protoc*, vol. 2, no. 3, pp. 727–738, 2007.

[16] L. Borisjuk, M. Hajirezaei, C. Klukas, H. Rolletschek, and F. Schreiber, "Integrating data from biological experiments into metabolic networks with the DBE information system," *In Silico Biology*, vol. 5, no. 2, pp. 93–102, 2005.

[17] B. H. Junker, C. Klukas, and F. Schreiber, "VANTED: A system for advanced data analysis and visualization in the context of biological networks," *BMC Bioinformatics*, vol. 7, pp. e109, 2006.

[18] J. D. Bewley and M. Black, *SEEDS: Physiology of Development and Germinantion*, Plenum Press, New York, 2. edition, 1994.

[19] B. B. Buchanan, W. Gruissem, and R. J. Jones, *Biochemistry and molecular biology of plants*, American Soc of Plant Physiologists, Rockville, Md, 4. edition, 2002.

[20] N. Sreenivasulu, L. Altschmied, V. Radchuk, S. Gubatz, U. Wobus, and W. Weschke, "Transcript profiles and deduced changes of metabolic pathways in maternal and filial tissues of developing barley grains," *Plant J*, vol. 37, no. 4, pp. 539–553, 2004.

[21] U. Wobus, N. Sreenivasulu, L. Borisjuk, H. Rolletschek, R. Panitz, S. Gubatz, and W. Weschke, "Molecular physiology and genomics of developing barley grains," *Recent Res Devel Plant Mol Biol*, vol. 2, pp. 1–29, 2005.

[22] "Brenda - BRaunschweig ENzyme DAtabase," http://www.brenda-enzymes.info/.

[23] "KEGG - Kyoto Encyclopedia of Genes and Genomes," http://www.genome.jp/kegg/.

[24] "MetaCyc - Metabolic Pathway Database," http://metacyc.org/.

[25] "ARAMEMNON - Plant Membrane Protein Database," http://aramemnon.botanik.uni-koeln.de/.

[26] F. C. Felker, D. M. Peterson, and O. E. Nelson, "[^{14}C]Sucrose uptake and labeling of starch in developing grains of normal and *seg*l barley," *Plant Physiol*, vol. 74, no. 1, pp. 43–46, 1984.

[27] OECD, "Consensus document on compositional considerations for new variety of barley (*Hordeum vulgare*): key food and feed nutrients and anti-nutrients," in *OECD Environmentl Health and Safety Publications*, Paris, 2004, vol. 12 of *Series on the Safety of novel foods and feeds*, pp. 1–42, OECD.

[28] F. W. T. Penning de Vries, "The cost of maintenance processes in plant cells," *Annals of Botany*, vol. 39, pp. 77–92, 1975.

[29] F. C. Felker, D. M. Peterson, and O. E. Nelson, "Growth characteristics, grain filling, and assimilate transport in a shrunken endosperm mutant of barley," *Plant Physiol*, vol. 72, no. 3, pp. 697–684, 1983.

[30] S. A. Quarrie, R. Tuberosa, and P. G. Lister, "Abscisic acid in developing grains of wheat and barley genotypes differing in grain weight," *Plant Growth Regulation*, vol. 7, no. 1, pp. 3–17, 1988.

[31] T. Thorbjørnsen, P. Villand, K. Denyer, O.-A. Olsen, and A. M. Smith, "Distinct isoforms of ADPglucose pyrophosphorylase occur inside and outside the amyloplasts in barley endosperm," *Plant J*, vol. 10, pp. 243–250, 1996.

[32] W. Weschke, R. Panitz, N. Sauer, Q. Wang, B. Neubohn, H. Weber, and U. Wobus, "Sucrose transport into barley seeds: molecular characterization of two transporters and implications for seed development and starch accumulation," *Plant J*, vol. 21, no. 5, pp. 455–467, 2000.

BAYESIAN MODELLING FOR GENETIC NETWORKS WITH TOPOLOGICAL CONSTRAINTS

Angela Grassi[1,2] *and Ernst Wit*[3]

[1]Institute of Biomedical Engineering, Italian National Research Council,
Corso Stati Uniti 4, I-35127 Padova, Italy
[2]Department of Information Engineering, University of Padova,
Via Gradenigo 6b, I-35131 Padova, Italy
[3]Department of Mathematics and Statistics, Lancaster University,
Lancaster, LA1 1AE, UK
angela.grassi@isib.cnr.it, e.wit@lancaster.ac.uk

ABSTRACT

In this paper we propose a Bayesian approach for modelling gene regulatory networks, starting from time-course gene-transcription data. Due to the potentially huge number of parameters, we assume a scale-free topological structure of the network, in agreement with the main features exhibited by biological networks. We construct a Bayesian hierarchical model in which the gene interaction matrix is an unknown parameter and a hyperparameter on it forces the desired topology. We consider a parsimonious mathematical model for describing the transcription dependencies. The identification of the parameters from real data is based on Markov Chain Monte Carlo techniques. A new way to do MCMC inference in high-dimensional problems with complex likelihoods is introduced for inferring the gene interaction matrix.

1. INTRODUCTION

In the last decade, the rapid improvement of experimental high-throughput technologies and the availability of data on a genome-wide scale have given rise to a great interest into a deeper understanding of a cell's underlying regulatory systems. Gaining insight into the complex structure of interactions among cellular elements is one of the prior aims of systems biology. Many different approaches have been proposed for the identification of gene interactions based on microarray data. Some of them based on profile comparison between couples of genes, e.g. [1], other model based, e.g. [2, 3]. See [4, 5] for an exhaustive review of the existing approaches for the modelling of gene regulatory networks. In this paper we consider a Bayesian approach for the reconstruction of gene regulatory networks starting from time-course gene-expression profiles. Our aim is to infer the regulatory interactions between genes, accounting also for the directionality of the regulation.

The remainder of the paper is organized as follows. In the next section we consider a parsimonious mathematical model for the transcription process. We then propose a Bayesian hierarchical model for gene regulatory networks imposing a topological constraint on the overall structure of the network. At the end of the section we discuss the interpretation of the parameters. Next we describe the Markov chain Monte Carlo technique used for inferring the parameters, discussing separately the non-standard update used for the gene interaction matrix. Finally we apply our algorithm to a 25-gene subnetwork in yeast, *Saccharomyces cerevisiae*.

2. MODELLING GENE TRANSCRIPTION

One of the principal means of control of the behavior of the cell is the control of the gene expression process. Transcription is the process by which the DNA sequence of a gene is expressed into mRNA molecules that then are translated into proteins. The regulation of transcription is due to special proteins called Transcription Factors (TFs) that can act as activators or inhibitors of the process. We say that a gene, i say, regulates the transcription process of gene j if the protein it encodes is a transcription factor for j, and is present in its active form (for instance phosphorilated). Modelling the dynamics of transcription and accounting for its regulatory mechanism, requires the knowledge of a number of biological quantities: the mRNA abundance levels, the active levels of TF proteins, and a set of gene-specific constants such as the basal expression level, the rate of decay of its mRNA, and the affinity that a specific transcription factor has for the given substrate [6]. Unfortunately some of these biological quantities, such as protein activity, are not yet available on a genome-wide scale. A common approach in modelling transcription assumes that the mRNA abundance level of a regulator approximates reasonably well the active level of the TF protein it produces. Although we are currently working on a Bayesian model which includes the dynamics of the transcriptional process, in this paper we assumes the following simplified dynamics.

2.1. Linear gene transcription model

The model we consider is a linear gene transcription model with Gaussian error on the log-transcription scale. We assume that the mRNA abundance level of each gene at time t, $y_j(t)$, results from a multiplicative effect of the mRNA abundance levels of a collection of other genes. By considering the log-transformed data, the relationship between the log-transcription level of a gene, say j, at time t and the others is assumed to be

$$\log y_j(t) = \sum_{i \neq j} x_{ij}\alpha_{ij} \log y_i(t) + \alpha_{0j} + \epsilon_j(t), \qquad (1)$$

where x_{ij} is the (i, j) element of the connectivity matrix X (indicating if gene i regulates gene j), α_{ij} are parameters representing the strength of interaction associated with x_{ij}, α_{0j} represents a sort of background mean expression level, and $\epsilon_j(t)$ is an i.i.d. Gaussian error with unknown variance σ^2.

3. A BAYESIAN MODEL

Starting from the available time-course gene-transcription data, y, we aim at the reconstruction of the directed graph which represents the regulatory influences at the gene level. We assume the gene transcription model (1) to hold and its unknown parameters being part of our Bayesian model. The connectivity matrix X is the structural parameter, representing the algebraic counterpart of the gene interaction graph.

3.1. Scale-free topological constraint

Assume the elements of the connectivity matrix X be defined as

$$x_{ij} = \begin{cases} 0 & \text{if gene } i \quad \text{does not regulate} \quad \text{gene } j; \\ 1 & \text{if gene } i \qquad \text{regulates} \qquad \text{gene } j. \end{cases}$$

The matrix X is a parameter that can be estimated from the available data. In this paper we follow the approach presented in [7], where a scale-free topological constraint was introduced in agreement with the main features exhibited by biological networks [8]: a relatively short path length between any two nodes (*small world property)*, the presence of many genes with few connections and few highly connected genes (*hubs)*, the lethal impact for the overall architecture of the network of the deletion of a hub (*centrality and lethality principle)*[9].

As the departing connectivity of each gene, *outdegree*, has been found to follow approximately a power law [10], in our model we impose a scale-free structure on the data via a power law prior on the outdegree:

$$P(x_{i\cdot} = k) \propto k^{-\gamma}, \qquad (2)$$

where $x_{i\cdot} = \sum_j x_{ij}$ is the outdegree of gene i, and γ is the scaling parameter. The way in which we actually incorporate this constraint in our model will be explained in subsection 4.2.

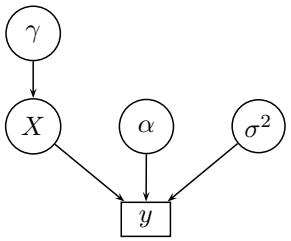


Figure 1. Directed acyclic graph showing the dependencies among the parameters of the model.

3.2. Interpretation of parameteres

The power law exponent γ can only take positive values, it represents a tuning parameter for our model; the higher gamma, the stronger the penalty on large numbers of gene-interactions. On γ we impose a uniform prior over the range $(0.5, 2.5)$, which are the most common values quoted in the recent literature. The noise variance σ^2 is assumed to follow an Inverse Gamma prior distribution.

4. MCMC INFERENCE

The hierarchical model has a directed acyclic graph structure (Figure 1), which in principle makes an MCMC implementation a standard option for performing Bayesian inference. We implemented a hybrid Gibbs and Metropolis-Hastings sampler in R with a few modifications that are described below. At every sweep parameters γ and σ^2 are updated in a standard manner. Due to the dimensionality of the connectivity matrix X and linear model parameters α, efficiency considerations force us to use something else for updating those parameters. Given that α is not our main interest, we estimate them within every sweep via an empirical Bayes approach. This means that X and α are updated jointly, whereby a proposal for X is supplemented with a value $\hat{\alpha}(X, y)$, to wit the maximum likelihood estimate for α given the connectivities and the data. The update for X itself involves a novel idea, using a fast ratio of p-values of a relevant test-statistic rather than the more involved ratio of conditional probabilities. This approach has strong similarities with [11].

4.1. Algorithm

Given an arbitrary set of starting values, the basic algorithm proceeds as follows:

(A1) X update: return updated values for X, α_{0j} and α_{ij}, $\forall i, j$.

(A2) σ^2 update: return updated value for σ^2.

(A3) γ update: return updated value for γ.

At each iteration of the algorithm we do a sweep of these three steps. Informal convergence criteria suggest that the sampler converges after 10,000 iterations. We obtain 300,000 iterations and consider the output after a 10,000 sweep burn-in, with a thinning value of 25.

4.2. Gene interaction matrix update

As we indicated above, we use a novel MCMC procedure for the update X' of X. The reason for this is that calculating the ratio $\frac{P(X'|\gamma)}{P(X|\gamma)}$ can be time-consuming. Instead, this ratio is replaced by a ratios of p-values from a frequentist testing strategy. This goodness-of-fit test-statistic

$$T_\gamma(X) = \sum_{k=1}^{N-1} \frac{(E_k - O_k)^2}{E_k},$$

where E_k are the expected counts under a power law distribution with parameter γ and O_k are the observed counts in the connectivity matrix X, checks to what extend the data is consistent with the scale-free topology with current exponent γ. The p-values,

$$p(T_\gamma(X)) = P(\chi^2_{(n-3)} > T_\gamma(X)) \tag{3}$$

from both the current value X and the proposal X' are then combined in the alternative acceptance probability

$$\text{AP} = \min\left\{1, \frac{p(T_\gamma(X'))}{p(T_\gamma(X))} \frac{p(y|X',\gamma,\sigma)}{p(y|X,\gamma,\sigma)}\right\}. \tag{4}$$

This acceptance probability guarantees that without data one would end up sampling from a network indistinguishable from one with a scale-free distribution. With data, the solution will converge to the most scale-free distribution that is consistent with the data. This approach, although novel, has direct links with Approximate Bayesian Computation [11], where for large probability calculations summary statistics are used instead.

In summary, for the X update we use the following procedure:

1. Propose a new value X' by flipping m random elements of X from 0 to 1 or viceversa, and its associated $\hat{\alpha}(y, X')$.
2. Compute
   - the likelihood with both the current value of $(X, \alpha(y, X))$ and the proposal $(X', \alpha(y, X'))$.
   - the p-value ratio as in equation (3).
3. Combine the information from Step 2 in forming the acceptance probability (4).
4. Accept the proposal $(X', \alpha(y, X'))$ with probability in step 3.
5. Proceed in the algorithm with σ^2 and γ updates.

5. APPLICATION

The model has been applied to a 25-gene subnetwork of the yeast, *Saccharomyces cerevisiae*, gene interaction network, the SGS1 neighbor subnetwork, described by [12]. We use expression data from a 77-time point microarray dataset of Spellman [13]. The aim is to infer the structure of the SGS1 subnetwork, reconstructing both the connectivity matrix X and the matrix α with the intensities of the interaction.

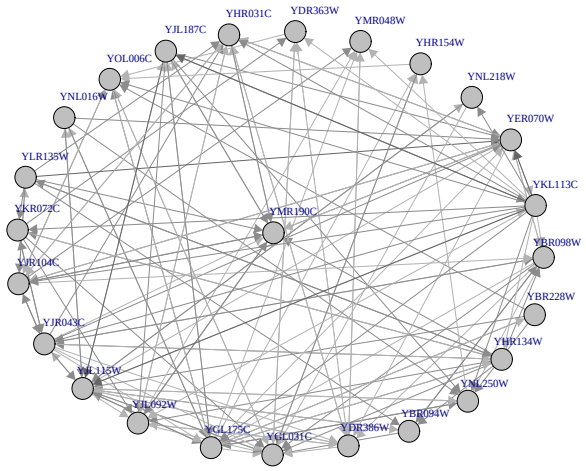


Figure 2. Reconstructed influence network, with YMR190C being the systematic name of SGS1.

We apply our algorithm and reconstruct the corresponding influence network displayed in Figure 2, where the different intensities of the directed edges represent the magnitude of the corresponding α parameters. To establish the presence of an edge between two genes, we chose a threshold of 0.8 on the posterior probability of X. Looking at the histogram of the outdegrees represented in Figure 3 we can see that the power law behavior is preserved by the algorithm. This is in contrast to the results that would have been obtained without any topological constraint. Figure 4 shows the outdegrees for the same model without topological constraint, where the number of links was chosen via a stepwise AIC procedure. This figure suggests that such model typically overestimates the number of links, leading to a non-sparse network.

6. CONCLUSIONS

In this paper we presented a Bayesian hierarchical model for the reconstruction of regulatory networks from gene expression data. We assume the transcriptional interactions to be described via a simple linear model on the logarithmic scale. Due to the typically high dimensionality of the data, a scale-free topological constraint on the outdegree distribution has been imposed. We develop an MCMC algorithm for inferring the structure of the network, implemented in the statistical software R. A key element of our procedure is that we impose the topological constraint on the network within the MCMC update of X. We tested the algorithm reconstructing a small regulatory network in yeast and compared our method with stepwise regression, showing that the decaying outdegree distribution is not preserved by that method. Our future work will be devoted to the extension of the model including a transcription dynamics closer to the biological behavior, which takes into account also the TF activities as unknown

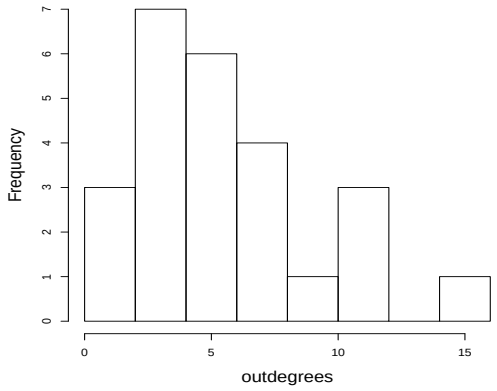


Figure 3. Histogram of outdegrees.

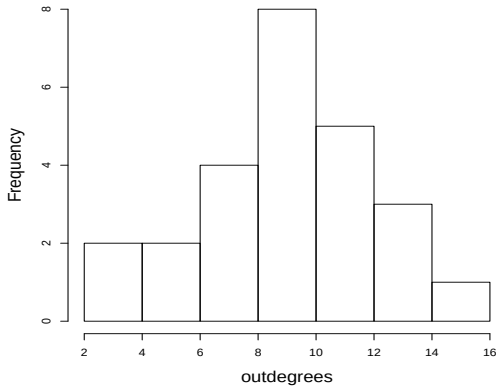


Figure 4. Histogram of outdegrees obtained via stepwise regression method.

parameters.

7. ACKNOWLEDGMENTS

Authors are grateful to the anonymous peer reviewers for their helpful comments. Angela Grassi is supported by a grant of Regione Veneto (Azione Biotech II - DGR 2112/02-08-05) to CNR-ISIB.

8. REFERENCES

[1] J. Schäfer and K. Strimmer, "An empirical bayes approach to inferring large-scale gene association networks," *Bioinformatics*, vol. 21, pp. 754–764, Mar. 2005.

[2] A. R. I. Nachman and N. Friedman, "Inferring quantitative models of regulatory networks from expression data," *Bioinformatics*, vol. 20, pp. 248 – 256, Aug. 2004.

[3] M. Barenco, D. Tomescu, D. Brewer, R. Callard, J. Stark, and M. Hubank, "Ranked prediction of p53 targets using hidden variable dynamic modeling," *Genome Biology*, vol. 7, no. 3, pp. R25, 2006.

[4] P. Smolem, D. Baxter, and J. Byrne, "Modeling transcriptional control in gene networks," *Bulletin of mathematical biology*, vol. 62, no. 2, pp. 247–292, 2000.

[5] T. S. Gardner and J. J. Faith, "Reverse-engineering transcription control networks," *Physics of Life Reviews*, vol. 2, no. 1, pp. 65–88, March 2005.

[6] R. Khanin, V. Vinciotti, V. Mersinias, C. P. Smith, and E. Wit, "Statistical reconstruction of transcription factor activity using michaelis-menten kinetics," *Bioinformatics*, vol. 63, pp. 816 – 823, Sep. 2007.

[7] E. Wit and N. Thomson, "Bayesian genetic networks with topological constraints," in *Proc. International Workshop on Statistical Modelling*, Sydney, Australia, Jul. 2005.

[8] P. Aloy and R. B. Russell, "Taking the mystery out of biological networks.," *EMBO Reports*, vol. 5, no. 4, pp. 349–350, Apr. 2004.

[9] H. Jeong, S. P. Mason, A.-L. Barabasi, and Z. N. Oltvai, "Lethality and centrality in protein networks," *Nature*, vol. 411, pp. 41 – 42, May. 2001.

[10] N. Guelzim, S. Bottani, P. Bourgine, and F. Képès, "Topological and causal structure of the yeast transcriptional regulatory network," *Nature Genetics*, vol. 31, pp. 60 – 63, May. 2002.

[11] V. Plagnol and S. Tavaré, "Approximate bayesian computation and mcmc," in *In Monte Carlo and Quasi-Monte Carlo Methods 2002*, H. Niederreiter, Ed. 2004, pp. 99–114, Springer-Verlag.

[12] A. H. Y. Tong, G. Lesage, G. D. Bader, H. Ding, H. Xu, X. Xin, J. Young, G. F. Berriz, R. L. Brost, M. Chang, Y. Chen, X. Cheng, G. Chua, H. Friesen, D. S. Goldberg, J. Haynes, C. Humphries, G. He, S. Hussein, L. Ke, N. Krogan, Z. Li, J. N. Levinson, H. Lu, P. Menard, C. Munyana, A. B. Parsons, O. Ryan, R. Tonikian, T. Roberts, A.-M. Sdicu, J. Shapiro, B. Sheikh, B. Suter, S. L. Wong, L. V. Zhang, H. Zhu, C. G. Burd, S. Munro, C. Sander, J. Rine, J. Greenblatt, M. Peter, A. Bretscher, G. Bell, F. P. Roth, G. Brown, B. Andrews, H. Bussey, and C. Boone, "Global mapping of the yeast genetic interaction network," *Science*, vol. 303, pp. 808 – 813, Feb. 2004.

[13] P. T. Spellman, G. Sherlock, M. Q. Zhang, V. R. Iyer, K. Anders, M. B. Eisen, P. O. Brown, D. Botstein, and B. Futcher, "Comprehensive identification of cell cycle-regulated genes of the yeast saccharomyces cerevisiae by microarray hybridization," *Molecular Biology of the Cell*, vol. 9, pp. 3273 – 3297, 1998.

PARTIAL ANNEALING AND LOCAL STRUCTURES IN BOOLEAN NETWORKS

Manu Harju, Juha Kesseli, and Olli Yli-Harja

Department of Signal Processing, Tampere University of Technology,
P.O. Box 553, FI-33101 Tampere, Finland
harju23@cs.tut.fi

ABSTRACT

Gene regulatory networks can be found to contain network motifs, small recurring local structures. In this work we utilize the Boolean network model to study the effects of some local structures on network dynamics and propose a partially annealed approach. In partial annealing the connections in the local structures are fixed while the other connections, including those between the fixed patterns and other nodes, are generated randomly. We focus on local structures that are dynamical, in the sense that their topology has no particular distinguishing characteristics, but they introduce correlations between connections and update rules. Comparison between predictions and simulations suggests that the partially annealed approach can be used to investigate networks with local dynamical structures. Based on the effects observed by comparing the partially annealed model and the traditional annealing we suggest that, if possible, the motifs should be defined based on both topological and dynamical characteristics.

1. INTRODUCTION

Boolean networks are a computationally simple way to model large dynamic networks [1]. In particular the model has been used in the context of genetic regulatory networks. A Boolean network can be described as a directed graph, where every node has a Boolean function as a update rule, and the inputs of an update rule are taken from the topology of the network. The network nodes are updated synchronously.

Despite their simplicity, many interesting phenomena occur. The response of the network to small perturbations can be used to characterize the general dynamical behavior of the system. In ordered networks small perturbations tend to die out, whereas in chaotic networks small perturbations spread to the system. The network is said to have critical dynamics, if the perturbations retain their original size on average. Recent studies indicate that real gene regulatory networks may have critical dynamics [2, 3].

Analysis of Boolean network dynamics can be done by using the annealed approximation [4, 5]. With this stochastic mean-field approximation it is possible to predict the dynamical behavior of a network in terms of average perturbation propagation for a given distribution of update rules. The annealed model gives an iterated map of the system states described with a simple probabilistic parametrization [6]. For example, perturbation size ρ and network state bias b can be predicted utilizing such a model. In this paper we generalize the annealed approximation to study the effects of fixed local structures.

Partially annealed approach has been used before in [7] to compute the pairwise mutual information between the nodes of random Boolean networks. In that case chains of nodes were found to have an important contribution to the special properties of critical networks. Here we focus on the effects of the specific dynamical local structures, that are introduced into the networks with an incidence above that expected by chance. The topological features of the structures are simple in themselves, but the networks can be used to study the effects of correlations between update rules and connections.

Observations from real gene regulatory networks suggest that the networks contain a significant number of small repeating structures called motifs [8]. In particular, the abundances and effects of feed-forward loops and dense overlapping regulons have been studied. In this paper we focus on simple tree-like motifs, leaving the analysis of these cases to future work.

2. PARTIALLY ANNEALED MODEL

The annealed model enables a probabilistic approach of studying the dynamics of Boolean networks. In the model the connections and the functions of the network are randomized after every time step, and the only topological information of the network is typically taken as the in-degree distribution, the out-degree distribution being Poissonian [1]. An annealed approximation for a given class of quenched Boolean networks is obtained by studying the behavior of an annealed model with the same distribution of functions. For many interesting cases, this turns out to be simple to analyze using probabilistic calculations.

Partial annealing is a generalization of the annealed model. In partial annealing only a part of the connections in the network is reshuffled. A specified fraction of the nodes is fixed to local structures, which remain intact. This allows a similar probabilistic calculations as in the traditional annealed case, making the analysis of the model computationally simple. In addition, the partially annealed model can also be simulated in the case of non-tree-like motifs, when our probabilistic approach is not directly applicable.

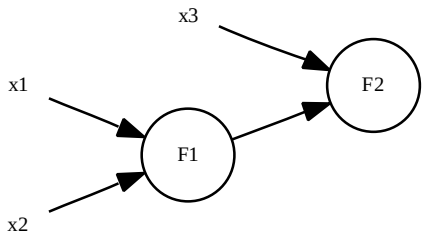


Figure 1. Two node motif.

We use the four-state model presented in [6] to keep track of the state of the network. In the four-state model we have two similar annealed networks, the first one without and the second one with a perturbation, i.e. a fraction of the nodes have their state inverted. The functions and the connections of the networks are the same for a given time step. For an arbitrary node there are four possible states, taking both networks into account. The probability that a node from the first network has value i and the corresponding node from the second network has value j at time t is denoted by $p_{ij}(t)$. For the analysis of local structures we need to keep track of the history of the network. The probabilistic description of the state at time k can be parametrized as

$$\mathbf{P}_k^t = [p_{00}(k), p_{01}(k), p_{10}(k), p_{11}(k)]\,.$$

For simplicity of notation, we denote the series of the state vectors at times $k < t$ by $\mathbf{P}^t$.

We can account for tree-like local structures by defining vector τ_i, which is defined for every internal node i of the motif. The length K_i of the vector $\tau_i = [\tau_{i,1}, \ldots, \tau_{i,K_i}]$ is the total number of inputs affecting the node in the partially annealed approximation. Note that this can be different from the number of inputs of the node, since all the leafs in the tree of influence of the node are considered inputs in this extended sense. For example, in Fig. 1 node F2 has three inputs x1, x2, and x3.

$\tau_{i,j}$ gives the distance of leaf node j from node i in the motif. In terms of dynamics, this means that the state of node i at time t will be affected by the state of a randomly selected node j at time $t - \tau_{i,j}$, taking into account the Boolean functions attached to the nodes. Due to the annealing, the state corresponds to one selected randomly from distribution $p_{mn}(t - \tau_{i,j})$. In our example shown in Fig. 1 the corresponding τ for the node F2 would be $[2, 2, 1]$.

The function distribution $\mathfrak{F}$ now consists of the functions constructed using composition of functions assigned to the nodes of the motifs and functions with annealed input connections. Continuing our example in Fig. 1, if both update rules f_1 and f_2 in the motif are two-input logical ands, the composed rule for node F2 will be given by $F_2(x_1, x_2, x_3) = f_2(x_3, f_1(x_1, x_2)) = x_1 \wedge x_2 \wedge x_3$, that is, a three-input logical and.

Like the annealed model, the partial annealing can also be formulated as a probabilistic model to predict the dynamical behavior of the network. The four-state model of probabilities $p_{ij}(t)$ in this generalized form can be written as a group of equations

$$\begin{cases} p_{00}(t) = \underset{f\in\mathfrak{F}}{E}\big[\sum_{x,y}(1-f(\mathbf{x}))(1-f(\mathbf{y}))A(\mathbf{x},\mathbf{y},\tau_f,\mathbf{P}^t)\big] \\ p_{01}(t) = \underset{f\in\mathfrak{F}}{E}\big[\sum_{x,y}(1-f(\mathbf{x}))f(\mathbf{y})A(\mathbf{x},\mathbf{y},\tau_f,\mathbf{P}^t)\big] \\ p_{10}(t) = \underset{f\in\mathfrak{F}}{E}\big[\sum_{x,y}f(\mathbf{x})(1-f(\mathbf{y}))A(\mathbf{x},\mathbf{y},\tau_f,\mathbf{P}^t)\big] \\ p_{11}(t) = \underset{f\in\mathfrak{F}}{E}\big[\sum_{x,y}f(\mathbf{x})f(\mathbf{y})A(\mathbf{x},\mathbf{y},\tau_f,\mathbf{P}^t)\big]. \end{cases} \tag{1}$$

The function A is defined as

$$A(\mathbf{x},\mathbf{y},\tau,\mathbf{P}^t) = \prod_{j=1}^{L}\prod_{m,n\in\mathbb{B}} p_{mn}(t-(\tau)_j)^{\delta(m-x_j)\delta(n-y_j)} \tag{2}$$

where L is the length of the vector τ, x_j is the j'th element of $\mathbf{x}$, $(\tau)_j$ is the j'th element of τ and $\delta(x)$ is the delta function. Note that A is the joint probability distribution function of $\mathbf{x}$ and $\mathbf{y}$, the inputs from networks 1 and 2, given the delay vector τ and the history of probabilities $\mathbf{P}^t$.

Since perturbation size ρ is often the main feature under study, it should be noted that the equations can be reparametrized with ρ and state biases b_1 and b_2 of both networks as follows [6]:

$$\begin{cases} \rho &= p_{01} + p_{10} \\ b_1 &= p_{10} + p_{11} \\ b_2 &= p_{01} + p_{11}. \end{cases} \tag{3}$$

For example, in the simulations that follow, the results are presented in terms of perturbation size.

3. RESULTS

In the simulations we have studied the simplest possible case of a local dynamical structure, the motif shown in Fig. 1. We used networks with half of the nodes in motifs and generated the rest of the network with random connections and fixed $K = 2$. Each network only contains dynamical local structures of one type. The functions of the rest of the network were generated randomly with bias $p = \frac{1}{2}$. The inputs for node type F1 were chosen randomly from the network, while the nodes of type F2 had node F1 as a fixed input. The second input of nodes of type F2 was also chosen randomly from the network.

In Section 3.1 we study a single example analytically with partial annealing to solve the fixed point of state bias. In Section 3.2 we present results of simulations for all selections of two-input functions with no redundant inputs in the motif shown in Fig. 1.

3.1. Bias map

Particular topics of interest are the fixed points of the system, since they can conveniently characterize the long-term behavior of the system. Those are often hard to obtain analytically, but there are also exceptions. We show a simple case where analytic solution can be found.

We denote b_t for the probability that a network node has value 1 at time t. The bias map [9] is an iterative mapping $b_t = f(\mathbf{B}_t^T)$, where $\mathbf{B}_t^T = [b_{t-1}, b_{t-2}, \ldots, b_{t-T}]$ is a vector consisting of the biases of T earlier time steps. For a single two-input and-node the bias of the next time step is $b_t = b_{t-1}^2$. We set the proportion of the motifs so that half of the network nodes are in the fixed structures. We consider the case in Fig. 1 where both the internal nodes have an and-function. For the motif with two sequential ands we can write the bias map as

$$b_t = \frac{1}{4}b_{t-2}^2 b_{t-1} + \frac{1}{4}b_{t-2}^2 + \frac{1}{4}. \quad (4)$$

To solve the bias fixed point b^*, we substitute $b_t = b^*$ for every t. Thus we end up solving the roots of the following polynomial:

$$\frac{1}{4}b^{*3} + \frac{1}{4}b^{*2} - b^* + \frac{1}{4} = 0. \quad (5)$$

The roots of the Eq. 5 are $b^* \approx -2.6511$, $b^* \approx 1.3772$ and $b^* \approx 0.2739$. The only valid solution for the bias fixed point is $b^* \approx 0.2739$. The solution found for b^* is a stable fixed point. The state bias can also e.g. show periodic behavior. For common mixtures of functions a stable fixed point is usually found [9].

The difference equations of higher degree in Eq. 1 can not in general be solved analytically. However, the fixed point can also be found by iterating the equations until convergence. The computational complexity of the iterations is determined by the number of functions in $\mathfrak{F}$ and the number of their inputs.

3.2. Comparison with simulations

In the numerical simulations of the networks we use only those ten Boolean rules which are truly two-input functions, so the total number of different motifs is 100. For every pair of functions we have generated 5000 networks with 600 nodes. Each network is set to a random state and a perturbation of 180 nodes is applied, which corresponds to $\rho = 0.3$. After flipping the nodes both the original and the perturbed networks are run for 20 time steps and the fractions p_{ij} are calculated for every time step. Each result is computed as an average over 5000 networks.

Predictions for the networks are calculated with both annealed and partially annealed approximations. For the partially annealed approximation we predict the first time step with the ordinary annealed approximation since there are no correlations due to dynamical motifs at this point. This is sufficient for initialization of the iterations in case the greatest element in every τ_i is at most 2. If there are larger dynamical motifs in the network, the initialization of the model should be considered in more detail than required in this case. In addition, we have also simulated some cases with networks having 1200 nodes with the same perturbation size $\rho = 0.3$ to get insight into how the network size affects the accuracy of the approximations.

The partial annealing is compared to ordinary annealed approximation and numerical simulations. The prediction errors for perturbation sizes averaged over all 100 dynamical motifs are shown in Fig. 2. It can be seen from the figure that partial annealing increases the accuracy of the predictions, when there are small fixed structures in the network.

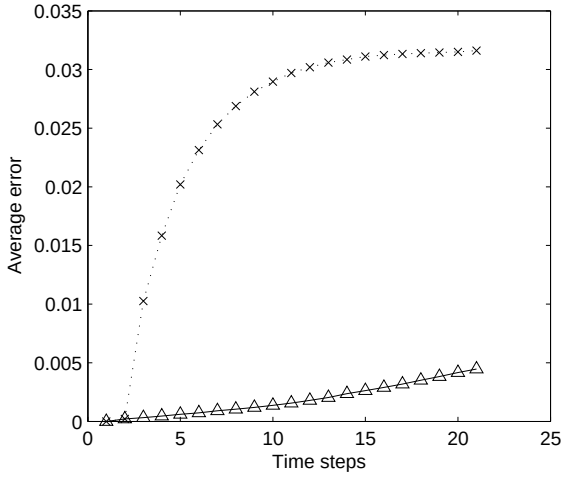


Figure 2. Average error for perturbation size ρ over time for the annealed (dotted line, crosses) and the partially annealed model (solid line, triangles). Initial perturbation size is set to 0.3. Errors are averaged over all 100 motifs.

For the motif with functions $x \wedge y$ for F1 and $x \wedge \neg y$ for F2, the progression of predicted and simulated perturbation sizes in time can be seen in Fig. 3. In addition to confirming the greater accuracy of the partially annealed approximation, it can be seen that for larger network sizes simulation results approach the partially annealed prediction, as expected. The standard deviations of the simulation results in Fig. 3 are bounded by $0.53 \cdot 10^{-3}$.

Figure 4 shows the histogram of the average effect of local dynamical structures over 100 cases considered. The effect sizes are computed by comparing the predictions given by the annealed and the partially annealed models after 19 time steps. As before, the initial perturbation size is set to 0.3. As can be seen from the figure, the effect of local dynamical structures can vary depending on the assignment of the update rules in the structure. However, the histogram is slightly biased to positive values, meaning that the local structures tend to preserve perturbations longer than networks with no dynamical motifs on average.

4. DISCUSSION

In this paper we only used the simplest possible motifs to demonstrate the principle of partial annealing. The partially annealed model can be applied to study the effects of biologically relevant and more complex motifs as well, even though the analysis of the model gets more difficult due to loops. Loops can create paths of different lengths between two given nodes, resulting in correlations that can not be analyzed using the parametrization $p_{ij}(t)$. Finding efficient computational methods for these cases remains a

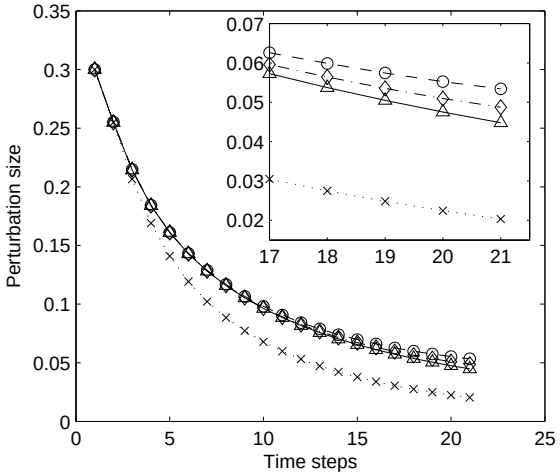


Figure 3. Perturbation avalanches for 600-node networks (dashed line, circles), 1200-node networks (dash-dot line, diamonds), partial annealing (solid line, triangles) and annealed approximation (dotted line, crosses). The functions for the motif nodes are [0001] for F1 and [0010] for F2.

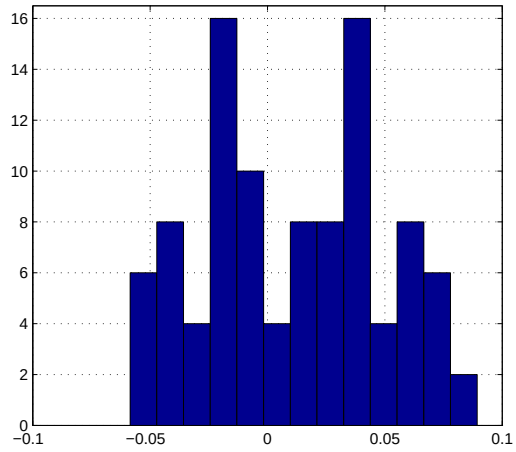


Figure 4. Histogram of differences between perturbation sizes predicted by the partially annealed and the annealed model. All 100 simulated motifs are included.

key topic of further studies.

Even though in this work we restricted our attention to networks with only one type of local structure with a fixed rate of occurrence, our method can be applied to study the effects of the proportion of nodes assigned to local structures. Networks with different mixtures of dynamical motifs can be investigated as well. These topics should be considered in more detail in future work.

The local structures of the simulations in this paper have no topological characteristic that would make them stand out if only the network graph was investigated. Instead, their main role was in introducing correlations between the assignment of network connections and the update rules. This suggests that all local structures of interest might not be found by looking only at the topological structure of the network. As more accurate information of genetic regulatory networks becomes available, it will be of interest to see if the global dynamics of the networks can be characterized in more detail utilizing abundances of local dynamical structures of the kind considered in this work. This, in turn, would pave the way towards deeper understanding of information flow in biological systems and, for example, support efforts of resolving the status of the hypothesis that cells might have critical dynamics.

5. REFERENCES

[1] M. Aldana, S. Coppersmith, and L. P. Kadanoff, *Boolean Dynamics with Random Couplings*, Springer Applied Mathematical Sciences Series. Springer, New York, 2003.

[2] M. Nykter, N. D. Price, M. Aldana, S. A. Ramsey, S. A. Kauffman, L. Hood, O. Yli-Harja, and I. Shmulevich, "Gene expression dynamics in the macrophage exhibit criticality," *Proceedings of the National Academy of Sciences of the USA*, vol. 105, pp. 1898–1900, 2008.

[3] I. Shmulevich, S. Kauffman, and M. Aldana, "Eukaryotic cells are dynamically ordered or critical but not chaotic," *Proceedings of the National Academy of Sciences of the USA*, vol. 102, no. 38, pp. 13439–13444, 2005.

[4] B. Derrida and Y. Pomeau, "Random networks of automata: a simple annealed approximation," *Europhysics Letters*, vol. 1, pp. 45–49, 1986.

[5] B. Derrida and D. Stauffer, "Phase transitions in two dimensional Kauffman cellular automata," *Europhysics Letters*, vol. 2, pp. 739–745, 1986.

[6] J. Kesseli, P. Rämö, and O. Yli-Harja, "Iterated maps for annealed Boolean networks," *Physical Review E*, vol. 74, pp. 046104, 2006.

[7] A. S. Ribeiro, S. A. Kauffman, J. Lloyd-Price, B. Samuelsson, and J. Socolar, "Mutual information in random boolean models of regulatory networks," *Physical Review E*, vol. 77, 2008.

[8] U. Alon, "Network motifs: theory and experimental approaches," *Nature Reviews Genetics*, vol. 8, pp. 450–461, 2007.

[9] P. Rämö, J. Kesseli, and O. Yli-Harja, "Stability of functions in gene regulatory networks," *Chaos*, vol. 15, pp. 034101, 2005.

INFERENCE IN A GENE NETWORK WITH TRANSCRIPTIONAL TIME DELAY

Catherine F. Higham

Department of Computing Science,
University of Glasgow,
Glasgow G12 8QQ, Scotland, UK
cfhigham@dcs.gla.ac.uk

ABSTRACT

We examine a proposed delay differential equation model for a gene regulatory network involving hes1 mRNA and its protein. This model reproduces reported oscillatory behaviour. By analysing the model we can a) explain possible dynamics of the biological system and b) inform the corresponding inference problem in parameter estimation. We extend published results to the case where mRNA and protein degradation rates may be different. Using linearisation, bifurcation theory and Lindstedt's method, we characterise the occurrence and nature of oscillatory solutions in terms of the system parameters. The usefulness of the results will be illustrated for the Bayesian inference problem on synthetic data based on biologically realistic data. Full details of the analysis and results on a real data set will be made available in a forthcoming technical report.

1. INTRODUCTION

The setting for our work is the biologically important instance where the expression of a gene is down regulated by its protein product. This arises, for example, with the p53 tumor suprressor protein whose intracellular activity is regulated through a feedback loop involving its transcriptional target [1]. Hes1 and NF-κB are also components of a short feedback inhibition loop [2] and [3]. We focus here on the case of the delayed Hes1 feedback loop featuring hes1 mRNA and Hes1 protein where both mathematical models and quantitative experimental data are available [4] and [2]. Here Monk [4] argues that the observed oscillatory behaviour is best accounted for by the introduction of a delay parameter that models the non-instantaneous nature of the transcription process, rather than by introducing an unknown third agent [2].

Verdugo and Rand [5] gave a mathematical analysis of Monk's model [4] for the Hes1 feedback loop. In particular, they derived closed form approximations for the amplitude and frequency of oscillation, where oscillatory behaviour is assumed to arise through Hopf bifurcation in the delay parameter. The analysis in [5] applies to the case where the decay rates of hes1 mRNA and Hes1 protein, key components of the feedback, are equal. In this work, we study the more realistic case where the decay rates are allowed to be different, also focussing on oscillatory behaviour. Our results are of interest in their own right as a means to understand how the system dynamics are driven by the model parameters and they can also be used to inform the corresponding Bayesian inference problem. In this extended abstract, we summarise our main results and apply them in a simple, controlled setting. Full details of the analysis and more extensive computations, including results for the data in Hirata et al. [2], will be made available in [6].

We remark that gene regulatory networks can be modelled at several different levels [7], [8] and [9]. In the case where molecule counts are low, it may be argued that discrete and/or stochastic models are more realistic than continuous-valued deterministic differential equations. However, in the context of inference, we believe that for the type of sparse time series data available in [2], the traditional chemical kinetics viewpoint is entirely appropriate.

2. MONK'S MODEL

Hes1 represses the transcription of its own gene through direct binding to regulatory sequences in the hes1 promoter [2]. Figure 1 provides a schematic representation of the Delayed Hes1 Feedback Loop. The Hes1 gene transcribes mRNA which passes from the nucleus to the cytoplasm. Hes1 protein is synthesised by the translation of hes1 mRNA. An interesting feature is that the protein represses transcript initiation from the hes1 gene through binding of Hes1 dimers to the promoter. Letting $m(t)$ and $p(t)$ denote the concentration of hes1 mRNA and Hes1 protein at time t, respectively, the model proposed by Monk [4] for the Hes1 feedback loop takes the form of a delay differential equation (DDE):

$$\dot{m} = \frac{1}{1+\left(p\left(t-\tau\right)/p_0\right)^n} - \mu_m m, \qquad (1)$$

$$\dot{p} = m - \mu_p p, \qquad (2)$$

where μ_m and μ_p are the rates of degradation of mRNA and protein, respectively, p_0 is the normalised repression threshold and n is the hill coefficient. The constant τ represents a time delay. This model was able to explain, via numerical simulations, the oscillation of hes1 mRNA and Hes1 protein in cultured cells observed by Hirata et al.[2].

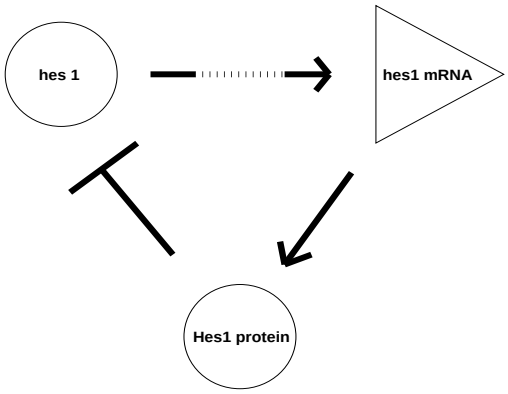


Figure 1. Representation of the Delayed Hes1 Feedback Loop. Transcription: Hes1 gene codes for mRNA which passes from the nucleus to the cytoplasm, Translation: Synthesis of Hes1 protein and Repression: Binding of Hes1 dimers to the promoter. The dashed line denotes a reaction involving a delay.

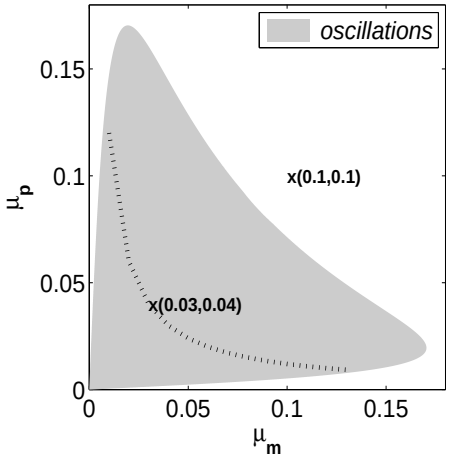


Figure 2. Oscillatory behaviour occurs when μ_m and μ_p lie within the bounded heart-shaped region. The dotted line across this heart represents our estimate of the value of $\mu_m \mu_p$, using analysis, from observed data.

3. BAYESIAN APPROACH

We propose a Bayesian approach to the parameter fitting problem which takes into account the inherent uncertainity in the data and uses our *a priori* bifurcation analysis to inform the choice of priors. Bayesian Inference and Markov Chain Monte Carlo (MCMC) methods have been recently advocated for the estimation of model parameters from Ordinary Differential Equations (ODEs) [10]. A Bayesian approach links the quantity that we are interested in, the probability that our parameters take certain values given the data, to two quantities that we can assign, the probability that we would have observed the measured data if the parameters took those values and our prior biological knowledge or ignorance about these parameters [11]. Whereas traditional parameter estimation methods are deterministic and point valued (for example COPASI [12]), these methods use Bayes' Theorem to assign probabilities to parameter values and can handle noise inherently.

Using the Bayesian approach to parameter estimation for nonlinear dynamical systems throws up several challenges, and these typically increase when time delays are included. Key issues are

- Dimensionality: models may involve several undetermined parameters,
- Identification: different parameter combinations may produce similar dynamics, for example increasing a production rate may be almost equivalent to decreasing a decay rate,
- Local maxima: the likelihood function may have many locally optimal values.

MCMC methods [13] can go some way to addressing these issues. In this work we show that *a priori* mathematical analysis can also be an effective tool. Our focus is on oscillatory time series, and we use bifurcation analysis and Lindstedt's method [14] in order to inform the choice of priors, thereby simplifying and focussing the parameter estimation process.

4. ANALYSIS

The work in [6] extends the results of Verdugo and Rand [5] to the more realistic case where the decay parameters (μ_m and μ_p) are not equal. The mathematical techniques involved include linearisation, bifurcation theory and Lindstedt's method. We summarize those results here and illustrate their use.

4.1. Stability of Equilibrium

Following the approach of Verdugo and Rand [5], equilibrium points for the system (equations (1) and (2)) are found by setting $\dot{m} = 0$ and $\dot{p} = 0$. After elimination and substitution, we obtain two expressions for p^* and m^*. To find out whether m^* and p^* are stable, we linearise about these points and define ζ and η to be deviations from the equilibrium. This results in a linear system which can be written as a second order Delay differential Equation (DDE). We look for periodic solutions of the form $e^{\lambda t}$ and obtain an expression for λ.

We then show that the equilibrium is stable for values of the time delay parameter τ below a critical value, denoted τ_{cr}, and unstable for values of τ above τ_{cr}. We obtain an explicit formula for ω, the oscillatory frequency at τ_{cr}, and show that oscillatory solutions arising from a Hopf bifurcation occur only when the difference between decay constants is sufficiently small. For physically reasonable solutions, we also require $\omega > 0$ and $\tau_{cr} > 0$. Using these conditions, it is possible to define a region where

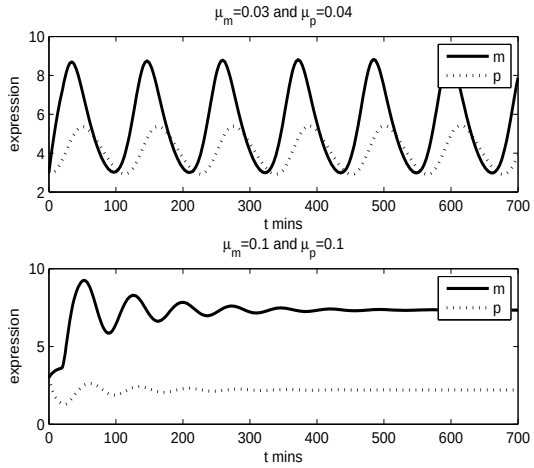


Figure 3. Typical system behaviour for parameters, from Figure 2, lying within the bounded area (above) and outside the bounded area (below). μ_m and μ_p are as stated. $p_0 = 90$, $n = 5$ and $\tau = 19$. The initial conditions were 3 for mRNA and 100 for protein and were assumed to hold for all $t < 0$.

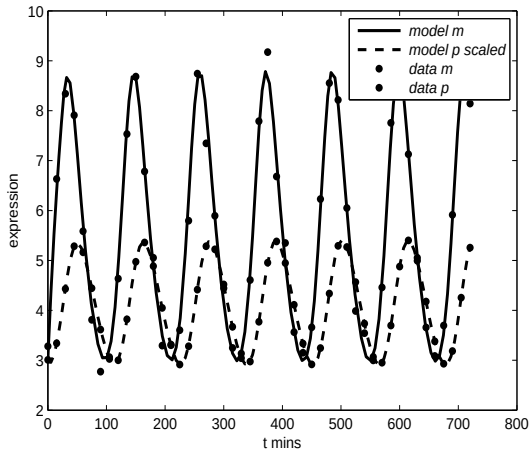


Figure 4. Numerically simulated data. Parameter values used: $p_0 = 90$, $n = 5$, $\mu_m = 0.03$, $\mu_p = 0.04$ and $\tau = 19$. The initial conditions were 3 for mRNA and 100 for protein and are assumed to hold for $t < 0$.

oscillations occur in terms of our system parameters. Figure 2 shows this region for $p_0 = 90$, $n = 5$ and $\tau > \tau_{cr}$. We see that this region is bounded, allowing us to consider the possibility of setting mathematically informed priors. Figure 3 compares behaviour for cases where parameters μ_m and μ_p lie within the bounded area and outside the bounded area. In the first case we see sustained oscillations and in the second the system moves towards a fixed steady state.

4.2. Lindstedt's Method

We summarise here the application of Lindstedt's method to our model. The Lindstedt method is a technique for uniformly approximating periodic solutions to ODEs when regular perturbation approaches fail [14]. Given that the time series is periodic, our goal is to find formulae that are relevant for all t. The key idea is to regard the frequency, ω, as unknown in advance, and to solve for it by demanding that an appropriate series expansion contains no secular terms. The result is closed form approximate expressions for the amplitude and frequency of oscillation. This information can be used to link τ_{cr} to the actual system delay.

5. EXPERIMENT

We now illustrate how our analysis can inform the Bayesian Inference approach to parameter estimation. We focus here on the inference of μ_m and τ assuming that the other parameter values are known. In [6] we show that μ_m is highly correlated with μ_p and so poses problems for inference. The time delay parameter τ is of interest because it relates to an important biological process. We use the analysis relating to the region of stability to set priors for μ_m and τ. As our objective is one of *proof of principle* we propose to test our method using synthetic data which has been generated knowing the underlying parameters.

5.1. Method

Data is numerically simulated from the mathematical model describing the system equations (1) and (2). Independent Gaussian noise (mean equals 0 and standard deviation equals 0.2) is then added to make this data more realistic. The data that we use in our inference experiment comprises 49 mRNA values taken every 15 minutes from $t = 0$ to $t = 720$ mins. See Figure 4 for details of the parameter values used in the model to generate the data points. We emphasise that although we show the corresponding data points for the protein, our inference experiment will be based on inferring all the parameters of the system from the mRNA data points alone.

To proceed with the Bayesian Inference method, we need a *likelihood function* and *prior probability density functions (pdfs)*. Following [10] we propose to base the likelihood function on the normal distribution, which is often used as a theoretical model for describing the noise or imperfections associated with experimental data. Its use is traditionally justified by appealing to the *central limit theorem* [11]. Our prior pdfs reflect our knowledge about our parameters and for this experiment we base our proposed priors within the bounded region defined in Figure 2 and related analysis. This gives a uniform value for $0.0018 < \mu_m < 0.1446$ and $13.4 < \tau < 55.6$. In biological terms these priors are extremely wide and we could use further approximations from [6] to reduce these prior ranges before MCMC methods are used. However, for illustration, we will conduct a grid search over the full range of parameters. Inference about the values of the unknown system parameters μ_m and τ is provided by the *posterior pdf* which according to Bayes Theorem can be obtained by multiplying our priors by the likelihood function.

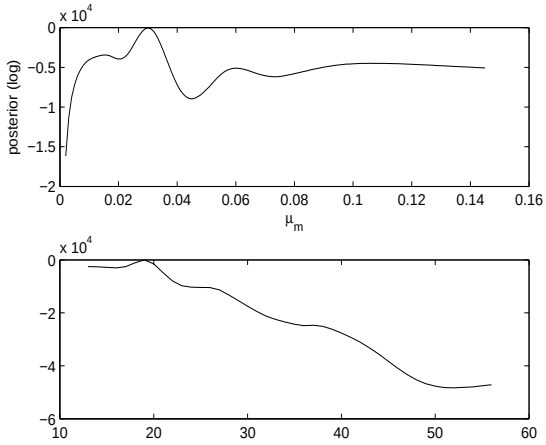


Figure 5. Log of posterior pdf for μ_m (upper) and τ (lower).

5.2. Results

Figure 5 shows, as logarithms, the relative probabilities of μ_m and τ given the data. Our best estimates for μ_m and τ are 0.03 and 19 respectively. These correspond, up to two significant figures, to our model parameters and so our method has successfully recovered this input. However the method has not just provided a point estimate but our evaluation of the parameters over a whole range of feasible solutions. We see that τ has an approximately uni-modal distribution but μ_m has several peaks. It is clear that μ_m poses problems for MCMC methods in determining the posterior pdf as the MCMC chains could get stuck within local maxima.

5.3. Summary from the Experiment

A Bayesian approach to parameter estimation can combine biological information and mathematical analysis to inform the choice of priors and predict not only the best fit but also alternative plausible solutions.

6. CONCLUSION

In this extended abstract we have outlined results that apply to a gene regulatory network model with time delay. Given that the biological system generates oscillatory solutions, *a priori* mathematical analysis can be called upon to focus the possible parameter range and hence improve the choice of priors in Bayesian inference. Further details, including fully Bayesian parameter estimates for the data in [2] will be made available in [6].

7. ACKNOWLEDGMENTS

CFH acknowledges the support of a Daphne Jackson Fellowship funded by the Leverhulme Trust. Useful feedback on this work was provided by Mark Girolami and Raya Khanin.

8. REFERENCES

[1] R. Bar-Or, R. Maya, L. Segel, U. Alon, A. Levine, and M. Oren, "Generation of oscillations by the p53-mdm2 feedback loop: a theoretical and experimental study," *Proceedings of the National Academy of Sciences USA*, vol. 97, pp. 11250–11255, 2000.

[2] H. Hirata, S. Yoshiura, T. Ohtsuka, Y. Bessho, T. Harada, K. Yoshikawa, and R. Kageyama, "Oscillatory expression of the bHLH factor Hes1 regulated by a negative feedback loop," *Science*, vol. 298, October 2002.

[3] A. Hoffman, A. Levchenko, M. Scott, and D. Baltimore, "The IκB-NF-κB signalling module: temporal control and selective gene activation," *Science*, vol. 298, pp. 1241–1245, 1997.

[4] N. A. Monk, "Oscillatory expression of Hes1, p53, and NF-kB driven by transcriptional time delays," *Current Biology*, vol. 13, pp. 1409–1413, 2003.

[5] A. Verdugo and R. Rand, "Hopf bifurcation in a DDE model of gene expression," *Communications in Nonlinear Science and Numerical Simulation*, vol. 13, pp. 235–242, 2008.

[6] C. F. Higham, "In preparation," *Department of Computing Science, University of Glasgow, Technical report*.

[7] D. J. Wilkinson, *Stochastic Modelling for Systems Biology*, Chapman and Hall/CRC, 2006.

[8] J. M. Bower and H. Bolouri, *Computational Modeling of Genetic and Biochemical Networks*, Massachusetts Institute of Technology, 2001.

[9] H. de Jong, "Modeling and simulation of genetic regulatory systems: a literature review," *J Comput. Biol.*, vol. 9, no. 1, pp. 67–103, 2002.

[10] S. Rogers, R. Khanin, and M. Girolami, "Bayesian model-based inference of transcription factor activity," *BMC Bioinformatics*, vol. 8, no. 2, 2006.

[11] D. Sivia, *Data Analysis: A Bayesian Tutorial (2nd ed.)*, Oxford University Press, 2006.

[12] S. Hoops, S. Sahle, R. Gauges, C. Lee, J. Pahle, N. Simus, M. Singhal, L. Xu, P. Mendes, and U. Kummer, "COPASI a COmplex PAthway SImulator," *Bioinformatics*, vol. 22, pp. 3067–74, 2006.

[13] C. Andrieu, N. de Freitas, A. Doucet, and M. I. Jordan, "An introduction to MCMC for machine learning," *Machine Learning*, vol. 50, pp. 5–43, Jan. - Feb 2003.

[14] S. H. Strogatz, *Nonlinear Dynamics and Chaos*, Addison Wesley, 1994.

STOCHASTIC MODELING OF INOSITOL-1,4,5-TRISPHOSPHATE RECEPTORS IN PURKINJE CELL SPINE

Katri Hituri[1], *Pablo Achard*[2], *Stefan Wils*[2,3], *Marja-Leena Linne*[1], *and Erik De Schutter*[2,3]

[1]Department of Signal Processing, Tampere University of Technology,
P.O. Box 553, FI-33101 Tampere, Finland
[2]Theoretical Neurobiology Laboratory, University of Antwerp, Belgium
[3]Computational Neuroscience Unit, Okinawa Institute of Science and Technology, Japan
katri.hituri@tut.fi

ABSTRACT

Transient rises in cytosolic calcium concentration play a crucial role in initiating long-term depression (LTD) of synaptic activity. Calcium release from endoplasmic reticulum is particularly important in LTD. In Purkinje cells, the release is mediated by inositol-1,4,5-trisphosphate (IP_3) receptors (IP_3Rs) that are highly expressed in dendritic spines. The small volume of spine and the small number of molecules involved increase stochasticity in biochemical processes. We studied the effects of stochasticity by comparing stochastic and deterministic simulations for two different IP_3R models. We found a significant difference between the responses when using small initial concentration of calcium or IP_3. Deterministic simulations of IP_3R activation do not produce realistic results under all conditions.

1. INTRODUCTION

Transient rises in the cytosolic Ca^{2+} concentration have an important functional role in neurons. In cerebellar Purkinje cell (PC) dendritic spines, they are essential for generation of LTD of synaptic strength [1, 2]. These temporary rises are due to Ca^{2+} entry from the extracellular space and Ca^{2+} release from intracellular stores such as endoplasmic reticulum (ER). In PC spines, IP_3Rs are responsible for the Ca^{2+} release from the ER and are relatively highly expressed.

Mathematical modeling is one of the important tools when trying to understand the complex behavior of proteins within networks and pathways. Several models have been proposed to describe the behavior of IP_3R (for a comprehensive review, see, for example, [3]). All the IP_3R models and simulations were deterministic until recent years. Deterministic models show the average behavior of the system, i.e. do not include any kind of randomness. However, when biochemical reactions occur in very small volumes, such as in dendritic spines, the number of molecules is low even with fairly large concentrations. The small number of molecules increases the possibility for stochastic effects in reactions. Both the randomness of molecular encounters and the fluctuations in the transitions between the conformational states of proteins become relevant. Given the small volume of the PC spine, it is of interest to test the stochastic nature of the system and to take the stochasticity into account to obtain biologically realistic simulations. Even though the deterministic approach is adequate in some cases, it fails to reflect the detailed nature of the biological system.

The aim of this work was to study the concentration levels at which the effects of stochasticity on the function of IP_3R can not be ignored. Among many mathematical models of IP_3R two recent ones were chosen as test cases. The models were implemented into two different software, GENESIS/Kinetikit [4, 5] for deterministic simulations and STEPS [6] for stochastic simulations, to perform two types of simulations, open probability simulations and dynamic simulations.

2. MATERIALS AND METHODS

2.1. IP_3R models

2.1.1. Model of Doi et al.

The IP_3R model of Doi et al.[7] was originally published as a part of a larger model for Ca^{2+} dynamics in the cerebellar PC spine and parameter values of this model were determined based on experimental data from Purkinje cells [7]. The model was originally implemented as deterministic. A schematic representation of the model is shown in Figure 1a.

All the reactions and their rate constants can be found in Supplemental material of the original article [7]. Briefly, in this model IP_3R needs to bind both IP_3 and Ca^{2+} to open and thus provide Ca^{2+} flux from ER lumen to cytosol. IP_3R has only one open state, RIC, in this model.

2.1.2. Model of Fraiman and Dawson

The IP_3R model of Fraiman and Dawson [8] (see Figure 1b) is the only model that has a Ca^{2+} binding site inside the ER in addition to the cytosolic binding sites found in other models. The parameter values used in this work can be found in Errata for the original article [8]. This model was originally simulated stochastically, as a Markov process.

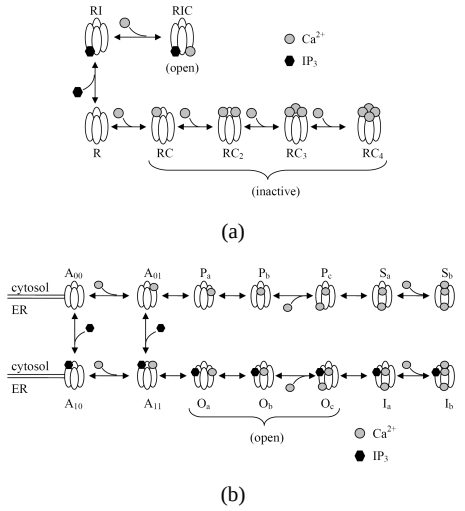


Figure 1. Schematic representation of the states and transitions of the IP_3R models. (a) Doi et al. (b) Fraiman and Dawson.

Originally, the six states, O_a, O_b, O_c, P_a, P_b, and P_c, are considered as open. However, IP_3R needs IP_3 to reach a stable open conformation [9, 10]. For this reason, three of the original open states were neglected in the present work and only states O_a, O_b, and O_c were considered as open. Also in the original article [8], the rate constant of the transition from A_{10} to A_{00} is defined as 'detailed balance'. We fixed the parameter by testing three values with deterministic open probability simulations (data not shown). Simulations were done as described in Section 2.3.1. The parameter values of 0 s^{-1} and 200 s^{-1} produced identical results while the value of 2000 s^{-1} slightly upraised the left side of the open probability curve. Based on these test simulations the value of 200 s^{-1} was chosen.

2.2. Simulation software

2.2.1. Genesis/Kinetikit

The GENESIS (GEneral NEural SImulation System) [4] simulation environment can be extended with Kinetikit [5] that is an extension for simulating reaction kinetics in well-mixed conditions. GENESIS/Kinetikit can be used to model and simulate the behavior of molecular networks and pathways. In this work, GENESIS version 2.2.1 for Cygwin and Kinetikit version 10 were used to obtain deterministic simulation results. Deterministic versions of the IP_3R models used are based on the law of mass action. The differential equation system was numerically solved (simulated) with the Exponential Euler method [4].

2.2.2. STEPS

STEPS (STochastic Engine for Pathway Simulation) [6] performs full stochastic simulation of reactions and diffusion of molecules in three dimensions. It extends the stochastic simulation algorithm (SSA) described by Gillespie [11]. In this work, STEPS developmental version 0.1.3 was used. Simulations were run both on a computer cluster and in a Cygwin environment on a standalone machine.

In the SSA, all reactions must be unidirectional. For this reason forward and backward parts of reversible reactions are defined as two separate reactions in the STEPS input file. In this early version of the software, the compartments of the modeled system are geometrically modeled as cubic shapes that are then discretized into small voxels. It is possible to define walls or surfaces between voxels that belong to different compartments. This enables modeling of surface bound molecules, such as ion channels, in their natural location.

2.3. Simulations

2.3.1. Open probability simulations

It has been experimentally shown that the open probability of IP_3R is dependent on the cytosolic Ca^{2+} concentration ($[Ca^{2+}]$) [12]. This dependence is bell-shaped with logarithmic x-axis. Originally, both models were built to reproduce this dependency.

In the deterministic open probability simulations, the behavior of a single IP_3R is simulated in an environment with constant $[Ca^{2+}]$ (several points, see Figure 2) and $[IP_3]$ (10 μM) until steady-state is achieved. The cytosol and also the ER had a volume of 0.1 μm^3 which is an experimentally defined average volume for PC spine cytosol [13]).

Deterministic simulations, using GENESIS/Kinetikit, were run for 5 or 15 s with a time step of 1 μs. The open probability of IP_3R was obtained at the end of simulation. In stochastic simulations with STEPS the models were simulated for 20 s using a sampling frequency of 0.1 s. In stochastic simulations the steady-state was achieved before 10 s time. For each initial Ca^{2+} concentration, 100 simulations were run with different seed values for the random number generator. The open probability was calculated as an average of the open IP_3Rs for the time interval 10-20 s over the 100 iterations.

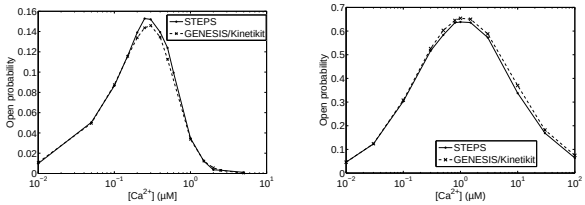
2.3.2. Dynamic simulations

A cell is a constantly evolving dynamic system. It is therefore important to study the dynamic behavior of intracellular functions in addition to steady-state properties. In this work, we studied the cytosolic Ca^{2+} concentration as a function of time. In the dynamic simulations, the Ca^{2+} flux through the open IP_3R was modeled in addition to IP_3R state transitions. In GENESIS/Kinetikit, the flux is modeled using the *kchan* entity which describes a ligand-gated channel. The equation for flux behind *kchan* is not published, but it is known to depend on the concentration gradient over the membrane and the rate of the flux is controlled with a parameter defined by the user. In STEPS, the flux is also dependent on the concentration gradient. Based on test simulations (data not shown) the equations for the flux are almost identical in GENESIS/Kinetikit and in STEPS.

Rate parameters of the flux were estimated for both simulators separately. It is estimated that 5400 Ca^{2+} ions go through open IP_3R during one opening and that the

Table 1. Initial conditions for dynamic simulations.

Species	Value
Number of IP_3Rs (naive state)	16
$[IP_3]$	0.1 μM, 0.2 μM, 0.5 μM, 1.0 μM, 5.0 μM
$[Ca^{2+}]_{cyt}$	0.01 μM, 0.05 μM, 0.1 μM, 0.2 μM, 0.5 μM, 1 μM
$[Ca^{2+}]_{ER}$	150 μM



(a) (b)

Figure 2. Results of open probability simulations. (a) Doi et al. (b) Fraiman and Dawson.

mean open time of IP_3R is 3.7 ms in physiological conditions [14]. The estimated parameter values for flux functions were 595 (unit not known) for GENESIS/Kinetikit and $5.8{\cdot}10^8$ $M^{-1}s^{-1}$ for STEPS.

The compartments in these dynamic simulations had the same volume as in open probability simulations and the volumes were considered as well-mixed (i.e diffusion was not taken into account). The initial conditions used in dynamic simulations are given in Table 1. The average number of IP_3Rs in a PC spine has been estimated to be 16 (see Supplemental material of [7]). There are five different initial concentrations for IP_3 and six for cytosolic Ca^{2+}. All combinations of the initial concentrations were used in simulations. A deterministic simulation response and 100 stochastic simulation responses were obtained for each situation. Data analysis was done with MATLAB®.

3. RESULTS

3.1. Open probability simulations

The results from open probability simulations are presented in Figure 2. The open probability curves obtained from deterministic (GENESIS/Kinetikit) and stochastic simulations (STEPS) are consistent. This expected result shows that both models were correctly implemented in both simulation environments.

3.2. Dynamic simulations

To study the dynamic behavior of the two IP_3R models, cytosolic $[Ca^{2+}]$ was followed as a function of time. Examples of simulation results with both IP_3R models are shown in Figure 3. The 100 individual stochastic iterations are shown as thin gray curves, their mean as thick solid curve, and the deterministic curve as dashed line for comparison. The variation in stochastic simulations increases, i.e. the gray curves are more spread out, when initial $[IP_3]$ and $[Ca^{2+}]$ are decreased.

The data was examined in two ways. First, the maximum Ca^{2+} concentration reached during simulations was measured as a function of the initial $[IP_3]$ and initial cytosolic $[Ca^{2+}]$ (data not shown) for the deterministic and for the mean of the stochastic cases. Second, the time at which half of the maximum cytosolic Ca^{2+} concentration was reached was measured as a function of the initial $[IP_3]$ and initial cytosolic $[Ca^{2+}]$. This is a convenient way to compare the curve slopes at the steepest region.

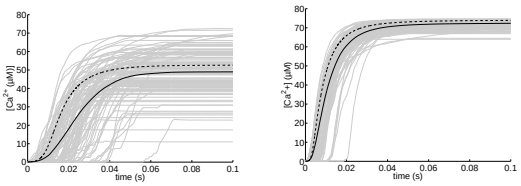
The maximum cytosolic Ca^{2+} concentration attained in the deterministic simulations with both models is dependent only on initial $[IP_3]$, not on initial $[Ca^{2+}]$. The latter might be due to the quick response to the rising $[Ca^{2+}]$. $[Ca^{2+}]$ rises when the channel opens and so the initial concentration does not have much influence on the maximum concentration. In the stochastic simulations, the results are similar to the deterministic ones above initial cytosolic $[Ca^{2+}]$ of 0.1 μM. Below this concentration value, the maximum $[Ca^{2+}]$ might be also dependent on the initial $[Ca^{2+}]$. This concentration threshold is identical for both models.

The time at which half of the maximum cytosolic Ca^{2+} concentration was reached is dependent on the initial $[IP_3]$ in deterministic and stochastic simulations. However, in the deterministic simulations, only a minor dependence on the initial $[Ca^{2+}]$ can be seen, whereas, in stochastic simulations, dependence on the initial $[Ca^{2+}]$ is more emphasized. In stochastic simulations, the dependence on both $[IP_3]$ and $[Ca^{2+}]$ is evidently seen. These results are consistent in both models.

To study the difference between deterministic and stochastic simulation results in times at which half of the maximum cytosolic Ca^{2+} concentration was reached the deterministic plots were subtracted from the stochastic plots for both models. The difference between stochastic and deterministic simulation results is shown in Figure 4. Furthermore, a threshold, below which the effect of stochasticity seems to be significant, can be determined from these plots. In the case of IP_3R model of the Doi et al. the thresholds for the initial $[IP_3]$ is around 1.0 μM and for the initial cytosolic $[Ca^{2+}]$ between 0.1 μM and 0.2 μM. In the case of the IP_3R model of Fraiman and Dawson the thresholds are slightly lower, namely 0.5 μM for $[IP_3]$ and 0.1 μM for $[Ca^{2+}]$. Our work implies that there is a difference when having 100 or less molecules. An important thing to notice is that the thresholds for $[Ca^{2+}]$ are close to the resting level of Ca^{2+} concentration, 70 ± 29 nM, if we apply results from hippocampal pyramidal neuron [15] to PC spines.

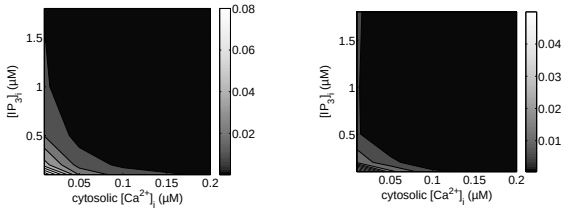
4. CONCLUSIONS

In this work, the importance of stochasticity in simulation of IP_3 receptor function was determined. The stochastic simulation algorithm gives more realistic results than the deterministic one because it takes random fluctuations



(a) (b)

Figure 3. Examples of dynamic simulations. Results from deterministic simulations (dashed) and the mean value (solid) of 100 stochastic simulations (thin gray) are shown. Initial concentrations: $[IP_3]$ = 0.2 μM, $[Ca^{2+}]$ = 0.1 μM. (a) Doi et al. (b) Fraiman and Dawson.



(a) (b)

Figure 4. Difference (gray scale) between deterministic and stochastic simulation results as a function of initial $[IP_3]$ and $[Ca^{2+}]$. (a) Doi et al. (b) Fraiman and Dawson.

into account. Based on dynamic simulation results of both models, we evaluated that there exists a threshold for initial IP_3 and cytosolic Ca^{2+} concentrations below which the effect can not be neglected. The threshold for Ca^{2+} concentration is close to the resting level of Ca^{2+} concentration in spines and thus it corresponds to the resting state of a spine before Ca^{2+} signals are induced. The present study strongly advocates for stochastic modeling and simulation of protein function.

5. ACKNOWLEDGMENTS

This work was supported by the Academy of Finland, project No. 213462 (Finnish Centre of Excellence program, 2006 - 2011). Tampere University of Technology Graduate school, Tampere Graduate school in Information Science and Engineering, and Finnish Cultural Foundation are acknowledged.

6. REFERENCES

[1] A. Konnerth, J. Dreessen, and G. J. Augustine, "Brief dendritic calcium signals initiate long-lasting synaptic depression in cerebellar Purkinje cells," *PNAS*, vol. 89, pp. 7051–7055, Aug 1992.

[2] M. Ito, "Cerebellar long-term depression: characterization, signal transduction, and functional roles," *Physiol. Rev.*, vol. 81, pp. 1143–1195, Jul 2001.

[3] J. Sneyd and M. Falcke, "Models of the inositol trisphosphate receptor," *Prog. Biophys. Mol. Biol.*, vol. 89, pp. 207–245, Nov 2005.

[4] J. M. Bower and D. Beeman, *The book of GENESIS: Exploring realistic neural models with the GEneral NEural SImulation System*, Springer-Verlag, New York, 1998.

[5] U. Bhalla, *Methods in Enzymology*, chapter Use of Kinetikit and GENESIS for modeling signaling pathways, pp. 3–23, Academic Press, New York, 2002.

[6] S. Wils and E. De Schutter, "STEPS: Stochastic simulation of reaction-diffusion in complex 3-D environment.," in *Abstract book of 15th Annual Meeting of the Organization for Computational Neurosciences CNS*2006*, Edinburgh, UK, July 2006.

[7] T. Doi, S. Kuroda, T. Michikawa, and M. Kawato, "Inositol 1,4,5-trisphosphate-dependent Ca^{2+} threshold dynamics detect spike timing in cerebellar Purkinje cells," *J. Neurosci.*, vol. 25, pp. 950–961, Jan 2005.

[8] D. Fraiman and S. P. Dawson, "A model of IP_3 receptor with a luminal calcium binding site: stochastic simulations and analysis," *Cell Calcium*, vol. 35, pp. 403–413, May 2004.

[9] J. S. Marchant and C. W. Taylor, "Cooperative activation of IP_3 receptors by sequential binding of IP_3 and Ca^{2+} safeguards against spontaneous activity," *Curr. Biol.*, vol. 7, pp. 510–518, Jul 1997.

[10] C. W. Taylor, P. C. da Fonseca, and E. P. Morris, "IP_3 receptors: the search for structure," *Trends Biochem. Sci.*, vol. 29, pp. 210–219, Apr 2004.

[11] D. T. Gillespie, "Exact stochastic simulation of coupled chemical reactions," *J. Phys. Chem.*, vol. 81, pp. 2340 – 2361, 1977.

[12] I. Bezprozvanny, J. Watras, and B. E. Ehrlich, "Bell-shaped calcium-response curves of Ins(1,4,5)P_3- and calcium-gated channels from endoplasmic reticulum of cerebellum," *Nature*, vol. 351, pp. 751–754, Jun 1991.

[13] K. M. Harris and J. K. Stevens, "Dendritic spines of rat cerebellar Purkinje cells: serial electron microscopy with reference to their biophysical characteristics," *J. Neurosci.*, vol. 8, pp. 4455–4469, Dec 1988.

[14] I. Bezprozvanny and B. E. Ehrlich, "Inositol (1,4,5)-trisphosphate ($InsP_3$)-gated Ca channels from cerebellum: conduction properties for divalent cations and regulation by intraluminal calcium," *J. Gen. Phys.*, vol. 104, pp. 821–856, Nov 1994.

[15] B. L. Sabatini, T. G. Oertner, and K. Svoboda, "The life cycle of Ca^{2+} ions in dendritic spines," *Neuron*, vol. 33, pp. 439–452, Jan 2002.

TOWARDS MODELING LIVER LOBULE REGENERATION IN 3D

Stefan Hoehme[1], Jan G. Hengstler[2], Marc Brulport[2], Alexander Bauer[2] and Dirk Drasdo[1,3]

[1] Interdisciplinary Centre for Bioinformatics (IZBI), University of Leipzig, Härtelstr. 16–18, D-04107 Leipzig, Germany; [2] Leibniz Research Centre for Working Environment and Human Factors (IfADo); University of Dortmund, Ardeystraße 67, D-44139 Dortmund, Germany; [3] Institut National de Recherche en Informatique et en Automatique (INRIA), Unit Rocquencourt B.P.105, 78153 Le Chesnay Cedex, France; hoehme@izbi.uni-leipzig.de, [last name]@ifado.de, dirk.drasdo@inria.fr

ABSTRACT

Liver regeneration is a complex process, having evolved to protect animals from the consequences of liver loss caused by food toxins. We establish a computational 3D single-cell-based model of the liver lobule regenerating after intoxication by CCl_4. In order to constitute a statistically representative liver lobule, we assemble information from light and confocal microscopy analyzed by an image processing chain. Furthermore we reconstruct the lobule sinusoidal blood vessel system and use direct 3D volume visualization to verify our results. This lays the foundation for understanding the complex regeneration dynamics through iterated experimentation, modeling and predictions. Preliminary simulations identified a sufficiently large cellular motility, necrotaxis as being beneficial and rapid cellular reorientation and polar cell-cell adhesion after division as essential for successful liver regeneration.

1. INTRODUCTION

Liver regeneration counteracts the consequences of loss of hepatic tissue caused by food toxins [1]. Such damage to hepatic tissue may experimentally be mimicked by administration of hepatic toxins. Carbon tetrachloride (CCl_4) is often used for that purpose ([2], [3]). It causes hepatocyte necrosis primarily in the peri-central areas of the liver lobules. The liver lobules are the building blocks of the liver. A human has about one million, a mouse about 1000 liver lobules.

Liver regeneration is a complex but precisely defined process in which the loss of hepatic tissue is balanced by proliferation of the existing mature hepatocytes, the parenchymal cells of the organ. In addition the other hepatic cell types, namely biliary epithelial cells, fenestrated endothelial cells, Kupffer and Ito cells, proliferate and regenerate lost hepatic tissue. Importantly, liver regeneration after toxic insult is not only a matter of hepatocyte proliferation but also of the capacity of the new cells to organize themselves within the characteristic three-dimensional liver lobule architecture that is tightly linked to liver function. A lobule has an approximately hexagonal shape. It contains microscopic branches of three types of vessels: the portal vein (transporting blood from the intestine to the liver), the hepatic artery (bringing in highly oxygenated blood),and bile ductules (carrying bile away to larger bile ducts) [1] (Figure 1A).

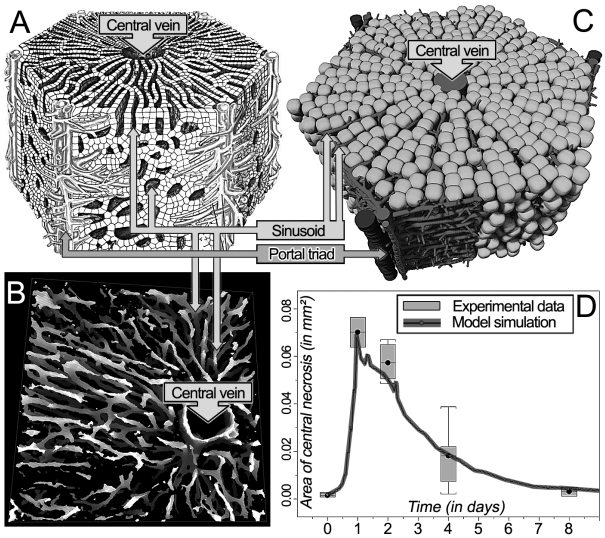


Figure 1: (A) Schematic illustration of a liver lobule (image from [11]), (B) Volume representation of the sinusoidal network after image/volume processing steps (1)-(6), (C) statistically representative liver lobule used as initial state for model simulations, (D) Area of central necrosis over time in experiment and model.

On average, each lobule is supplied by three portal veins. In order to guarantee liver function the lobule architecture has to insure that (i) the blood can freely flow from the portal veins through the sinusoids to the central vein localized in the center of each lobule, and that (ii) a one-to two-cells thick wall between the sinusoids form to permit maximum exchange of metabolites between the blood and the hepatocytes. Since the same processes occur in all liver lobules, it is sufficient to consider the regeneration of a single lobule.

In this paper we describe a process chain which we believe reflects the necessary steps in the systems biology of multi-cellular tissues: (i) the experiments that monitor spatial-temporal information of the liver tissue, (ii) then the image processing chain to extract information on key lobule parameters in 3D from confocal microsco-

py. Considering further experimental data from light microscopy, we are able to set up a statistically representative liver lobule. (iii) Finally we present a mathematical spatial-temporal model of liver lobule regenerating after CCl_4 intoxication.

The basic unit of our model is the individual cell. Such single-cell-based models (reviewed in [5]) are particularly suited to represent the spatial-temporal multi-cellular organization processes within the complex architecture of liver lobules since they permit to represent spatial structures that vary on the scale of individual cells (e.g. one-cell-thick layers [6]). Intracellular processes can easily be integrated into single-cell-based models [7].

The presented model belongs to the class of off-lattice models [8]. In contrast to cellular-automaton models where cells are represented as objects on a discrete lattice [9], cell positions in off-lattice models can gradually change and biomechanical influences can directly be represented. This makes off-lattice models being particularly suited to mimic the dynamics of cells in complex spatial. Within our model each individual cell is controlled by a limited number of effective parameters, such as for instance the probability of proliferation or the velocity of cellular reorientation after division. We use the presented model of a liver lobule regenerating after CCl_4 intoxication to study which parameters are most relevant for the regeneration process.

2. EXPERIMENTS

The left liver lobes of male C57BL/6N mice were used. 50 – 100µm thick vibratome slices were prepared for immunostaining as described in [10]. Briefly, sinusoidal lining cells were stained by an anti-CD31 antibody and a Cy3-conjugated secondary antibody. The nuclei were stained with 2.35µg/mL 4',6 diamidino 2 phenylindole (DAPI) for 5min resulting in blue fluorescence [10]. Central veins were identified by glutamine synthetase immunostaining [11]. Liver damage was induced by intraperitoneal injection of CCl_4 (1.6 mg/kg body weight). Mice were analyzed 12h, 1d, 2d, 3d, 4d, 8d and 16d after injection of CCl_4. Three mice were analyzed per time point. Data on cellular proliferation was investigated by BrdU incorporation as described in [11] and later was used to parameterize our model. BrdU was injected 6, 4 and 2h before preparation of the livers. Immunofluorescence was investigated by means of a confocal laser scanning microscope (LSM Meta 510; Zeiss, Oberkochen, Germany).

3. IMAGE/VOLUME PROCESSING CHAIN

The confocal microscope records spatially consecutive images of multi-stained (CD31 and DAPI) liver lobule slices into TIFF files with 32 bits per pixel (bpp) and RGBA color space. One dataset of a specific lobule consists of 50-100 layers of effectively 1 µm offset. In a first image/volume processing chain of six sub-steps we reconstruct the lobular vessel network. Initially, we focus on the red-channel (8 bpp) which yields the CD31-staining that accentuates endothelial cells enwrapping vessels in red. To enhance the contrast, we (1) rescaled the amplitude of the source images using a window-level transformation and an adaptive histogram equalization technique. We further applied (2) a non-linear 5x5 median filter [12] to reduce inherent noise. After (3) discarding consecutive, virtually empty slices (with only very few pixels above the signal threshold of $\psi = 128$), we import the remaining images into our software Cellsys for volume processing. We apply (4) three-dimensional non-linear anisotropic diffusion filtering [13] to perform edge preserving smoothing of the volume data and (5) delete remaining isolated structures. To limit computational complexity, we (6) binarize the volume data (1 bpp). The resulting volume representation (for an example see Figure 1B) of the lobular vessel network is used to detect the exact position and spatial orientation of the central vein and the sinusoidal vessel network. Both are integrated into a graph representation (the vessel graph) by first searching the volume for accumulated voxels (nodes) and then testing the connectivity of nodes in vicinity (edges). The vessel graph is further optimized e.g. to exclude parts where the staining in the experiment may have failed. We then use the constructed vessel graph to extract statistical information for example on the mean vessel radii, the mean minimal orthogonal sinusoid distance and specific sinusoidal branching properties.

A second similar image/volume processing chain is used to obtain the positions of the hepatocyte nuclei. Here, we focus on the blue-channel (8 bpp) which yields a DAPI-staining accentuating hepatocyte nuclei in blue color by forming fluorescent complexes with natural double-stranded DNA. The steps (1) - (6) are applied using an adjusted signal threshold of $\psi = 60$. The resulting volume representation of the hepatocyte nuclei is transformed to a set of points (nodes) in 3D by searching the volume for accumulated voxels. This point set is used to obtain statistical information for example on the mean hepatocyte position, volume, size and shape and the mean neighbor distances using three-dimensional Voronoi tesselation. This secondary volume analysis integrates knowledge from first processing chain for example to properly rescale the calculated mean hepatocyte volume by subtracting the volume of the vessel network.

We applied equivalent processing chains to confocal micrographs of six different lobules to obtain a representative parameterization as a starting point for our model simulations (Figure 1C).

4. MODEL

The basic model unit is an individual cell. Since freshly isolated hepatocytes in suspension (Figure 2A) have a spherical shape, we assume each model cell to be spherical in isolation. We subdivided the interphase into G1, S and G2-phase. Similarly as determined in [14], we assume that a cell after it receives a stimulus to enter the proliferation phase needs on the average 10 hours to enter the S-phase which in our simulations has a length of 8 hours. Cell growth is modeled by a gradual increase of cell volume until the cell has doubled its volume at the end of the G2-phase (Figure 2B). Hence, cell divisions

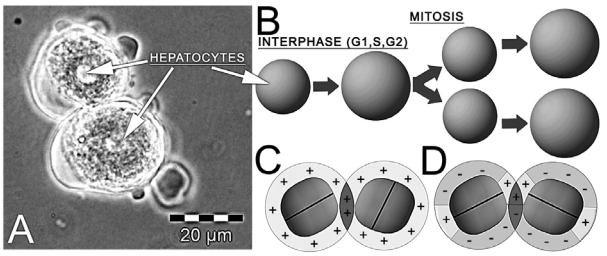


Figure 2: (A) Isolated hepatocytes cultured in suspension, (B) Illustration of cell division in model, (C) Isotropic cell-cell adhesion, (D) Polar cell adhesion; (C+D) "+": adhesive, "-": non-adhesive surface, dark grey: cell-cell contact area.

are accompanied by an increasing requirement of space. We assume an average intrinsic cell cycle time τ to be influenced at the level of individual cells by regulatory factors and mechanical stress. The spatial-temporal proliferation pattern is directly inferred from experiments with BrdU label (S-phase marker). We further assume that a cell may sense the degree of its deformation [15]. In our model, a cell can enter the cell cycle only if it is not deformed greatly, mimicking contact inhibition of growth.

We model the attractive and repulsive interactions utilizing an extended Hertz-model that yields an appropriate description of the cell-cell interaction [8,16,17]. This model approximates cells as homogeneous, isotropic, elastic and adhesive spheres. Once cells get into contact, cell-cell adhesion initially leads to an increase of the interaction area accompanied by an increasing deformation of the cell until adhesion and deformation forces are balanced. The interaction energy V_{ij} contains the cell stiffness, its compressibility, the density of cell surface receptors involved in the contact between hepatocytes and the bond energy of a single receptor. We distinguish between two cases: (a) Isotropic (non-polar) cells. The force between adjacent cells here depends only on the distance but not on the orientation of the cells (Figure 2C). (b) Polar cells. For polar cells we assume that cell adhesion molecules are placed only in certain membrane regions of the cell surface. The force between adjacent cells is proportional to the overlap area of the membrane regions that are covered with adhesion receptors: $\underline{F}_{ij}^{pol} = A_{ij}^{adh} A_{ij}^{-1} \underline{F}_{ij}$, where A_{ij}^{adh} is the overlap area of the adhesive membrane regions, A_{ij} the total contact area and $\underline{F}_{ij}$ the interaction force if the cells were isotropic. $\underline{F}_{ij}^{pol}$ is the interaction force of polar cells. The overlap of the adhesive surface areas in 3D depends on the angles Φ_1 (on x-y-plane) and Φ_2 (orthogonal to x-y-plane) between two polar hepatocytes (Figure 2D). Note that adhesion reaches a maximum if adhesive regions completely overlap (e.g. if Φ_1=0 and Φ_2=0).

In the absence of chemotactic signals, cultured hepatocytes perform a random walk-like movement that we characterize by the cell diffusion constant. While in mechanical contact with other cells, proliferating cells exert a pressure on their neighbors. The neighboring cells try to escape this pressure by moving against the friction caused by other neighbor cells and extracellular material (e.g. extracellular matrix). The movement could be partly passive, due to pushing, and active, if cells migrate into the direction into which they escape the mechanical stimulus [17]. The cell migration dynamics is modeled by an equation of motion (Equation 1) for each individual cell that summarizes active and passive forces. For each cell it follows the equation of motion [17]:

$$\varsigma \underline{v}_i(t) = \sum_j \varsigma_{ij} (\underline{v}_j(t) - \underline{v}_i(t)) + \sum_j \underline{F}_{ij} + \sqrt{2\varsigma^2 D} \underline{\eta}_i(t) \quad (1)$$

$\varsigma \underline{v}_i(t)$ denotes the velocity of movement of cell i, the first term on the rhs the friction between adjacent cells, $\sum_j \underline{F}_{ij}$ is the force between cell i and its neighbor cells j and the last term the random component of the cell movement. ς denotes the effective friction between a cell i and its surrounding extra-cellular matrix, ς_{ij} the friction between adjacent cells. $\underline{v}_i(t)$ denotes the velocity of cell i. $\underline{F}_{ij}$ summarizes adhesive and repulsive forces between cells i and j, $\underline{\eta}_i(t)$ is a "noise" term that summarizes the random component in the cell movement.

For polar cells we permit cell orientation changes. For simplicity we model such reorientation by energy minimization which can be shown is an alternative to a forced-based single-cell dynamics [8].

Our model is parameterized by measurable quantities. However, not all parameters for cells in the liver lobule are precisely known. This is why we estimated the model parameters partly from experiments, partly from published data: L = 23µm, τ = 24h, elastic modulus E = 400 Pa, Poisson ratio ν = 0.4, density of surface receptors $\rho = 10^{15}/m^2$, effective friction ς = ~0.1kg/s, cell diffusion rate D~10^{-11}cm^2/s. $F_T = 10^{-16}$ J. (See also [15] for published data on biophysical and cell-biol. parameters.)

5. SOFTWARE

For all model simulations and image/volume processing we use the software Cellsys developed by our group (http://ms.izbi.uni-leipzig.de/). Cellsys is an object-oriented software approach comprising modeling, simulation, measurements, visualization and specialized image processing capabilities for multi-cellular systems. It is written in portable ANSI C++ and thus works with all versions of Microsoft Windows and Linux operating systems. Cellsys utilizes the free and portable libraries zlib (www.zlib.net), libtiff (www.libtiff.org) and the OpenGL programming interface (www.opengl.org) for producing real-time 3D graphics.

The software is able to visualize and quantify measurements, obtain screenshots and videos of the simulations and produce compressed state saves in predefined intervals. State saves can later be loaded and used to examine simulated liver lobules in real-time 3D using the interactive user interface. Cellsys is also able to output

scene description files e.g. for the popular open-source raytracer Povray (www.povray.org) to create high resolution images (Figure 1C). The core algorithms in Cellsys are parallelized using OpenMP (www.openmp.org) to exploit the computational power of modern shared memory multi-core multi-processor machines. The solution of the stochastic equations of motion is supported by the multithreaded version of SuperLU for shared-memory parallel machines [18]. The calculation of neighboring cells essential to short-range cell-cell interactions is implemented based on spatial hashing. To further accelerate the calculation we use inline assembly for intensely called functions as float to int conversions. We find the solution of the equations of motion and the neighbor detection to be the major performance bottlenecks of our model simulations. However, on a typical Intel Core2 E6500 system with 2 GB RAM a characteristic model simulation in 3D of a lobule intoxication/regeneration experiment spanning 21 days completes in ~26 hours.

6. DISCUSSION

We were able to set up a statistically representative liver lobule in 3D using experimental data and imagery from light and confocal microscopy. The assembly of such images to 3D volume data and subsequent image/volume processing steps enabled us to determine numerous parameter values inaccessible to common measurements using only 2D slices. However, knowledge of those parameters as the specific branching structure of the sinusoidal network, the mean radius of a sinusoidal vessel (3.6 μm) or the mean orthogonal minimal sinusoid distance in 3D (14.1 μm) is essentially needed to construct a representative liver lobule in 3D that may serve as an initial state for subsequent modeling.

Based on this representative lobule we established a single-cell-based, lattice-free model in 3D for the regeneration process after CCl_4 intoxication. Conditions for modeling were chosen to reflect the in vivo situation as closely as possible such that computer simulation and experimental observation of many aspects (hepatocyte number and density, number of BrdU positive cells, key properties of the sinusoidal network, area of the central necrosis) are in good agreement for all observed time points (for an example refer to Figure 1D).

We are aware that several details of our current model deviate from genuine liver lobules. For instance the size of hepatocytes varies and non-parenchymal cells have not yet been included. Nevertheless, preliminary simulations of different experimental scenarios show a very good agreement with the experimental situation and demonstrate predictive power.

Although an exhausting and detailed study of the influences of all key parameters in the complex process of liver regeneration after intoxication is still pending and will be published in [19], preliminary simulations founding on the basics described in this paper identified a rapid cellular reorientation (towards the direction of the central vein) after cell division and polar cell-cell adhesion as essential for a successful regeneration of the characteristic lobule architecture. Moreover, a sufficiently large cellular motility and necrotaxis (cellular movement towards a cytokine gradient) turned out to be largely beneficial for a complete lobule regeneration.

7. ACKNOWLEDGEMENTS

S.H. and D.D. acknowledge support by the BMBF-grant Hepatosys project 0313081.

8. REFERENCES

[1] G.K. Michalopoulos and M. DeFrances, "Liver regeneration", *Adv Biochem Eng Biotechnol*, 93:101-34, 2005.

[2] F. Lafdil et al., "Induction of Gas6 protein in CCl_4-induced rat liver injury and anti-apoptotic effect on hepatic stellate cells", *Hepatology*, 44(1):228-39, 2006.

[3] A. Nussler et al., "Present status and perspectives of cell-based therapies for liver diseases", *J Hepatol*, 45(1):144-59, 2006.

[4] D.A. Badger et al, "The role of inflammatory cells and cytochrome P450 in the potentiation of CCl_4-induced liver injury by a single dose of retinol. Toxicol", *Appl Pharmacol*, 141(2):507-19, 1996.

[5] A.R.A. Anderson et al. (eds.), *Single-cell-based models in biology and medicine*. Birkhäuser, 2007.

[6] D. Drasdo and M. Loeffler, "Individual-based models on growth and folding in one-layered tissues: Intestinal Crypts and early development", *Nonlin. Analysis*, 47:245-256, 2001.

[7] P. Hogeweg, "Evolving Mechanisms of Morphogenesis: on the Interplay between Differential Adhesion and Cell Differentiation", *J. theor. Biol.*, 203:317-333, 2000.

[8] D.Drasdo et al. “On the role of physics in the growth and pattern formation of multi-cellular systems: What can we learn from individual-cell based models? *J. Stat. Phys*. 128(1-2):287-345 2007.

[9] A. Deutsch and S. Dormann. *Cellular Automaton Modeling of Biological Pattern Formation*. Birkhäuser, 2004.

[10] M. Brulport et al., „Fate of extrahepatic human stem and precursor cells after transplantation into mouse livers”, *Hepatology*, 46(3):861-70 2007.

[11] S. Hoehme et al., “Mathematical modeling of liver regeneration after intoxication with CCl_4“, *Chemico-Biological Interactions*, 168, 74-93, 2007.

[12] J.W. Tukey, *Exploratory Data Analysis*, Addison-Wesley, Reading, MA, 1971.

[13] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 12(7):629-639, 1990.

[14] O.K. Vintermyr and S.O. Doskeland, "Cell cycle parameters of adult rat hepatocytes in a defined medium. A note on the timing of nucleolar DNA replication.", *J Cell Physiol*., 132(1):12-21, 1987.

[15] D. Drasdo and S. Hoehme, "A single-cell based model of tumor growth in vitro: monolayers and spheroids.", *Phys. Biol*., 2:133-147, 2005.

[16] R.E. Mahaffy et al., "Scanning probe-based frequency dependent microrheology of polymer gels and biological cells", *Phys.Rev.Lett*., 85:880-883, 2000.

[17] D. Drasdo, "Coarse graining in simulated cell populations.", *Adv. Compl. Syst*., 2&3, 319-363, 2005.

[18] J. Demmel et al., "An Asynchronous Parallel Supernodal Algorithm for Sparse Gaussian Elimination", *SIAM J. Matrix Analysis and Applications*, 20, 915-952, 1999.

[19] S. Hoehme et al., "Liver regeneration after intoxication by CCl_4: Experiment and Theory", in. prep., 2008.

NOISE-DRIVEN STEM CELL AND PROGENITOR POPULATION DYNAMICS

Martin Hoffmann[1] *and Joerg Galle*[1]

[1]Interdisciplinary Centre for Bioinformatics, University of Leipzig,
Haertelstr. 16-18, 04107 Leipzig, Germany,
hoffmann@izbi.uni-leipzig.de, galle@izbi.uni-leipzig.de

ABSTRACT

The balance between maintenance of the stem cell state and terminal differentiation is influenced by the cellular environment. The switching between these states has long been modeled as a transition between stable attractor states defined by molecule networks. Herein stochastic fluctuations are either suppressed or can trigger transitions between deterministic attractors but they do not play a governing role. We present a novel mathematical concept in which stem cell and progenitor population dynamics are described as a probabilistic process that arises from cell proliferation and small fluctuations in the state of differentiation. These state fluctuations reflect random transitions between different activation patterns of the underlying regulatory network. Importantly, the associated noise amplitudes are assumed to be set by the environment in a state-specific manner and it is their variability that actually governs population dynamics. We suggest that state-specific noise modulation by external signals can be instrumental in controlling stem cell and progenitor population dynamics.

1. INTRODUCTION

A growing body of evidence indicates that noise is not generally detrimental to biological systems but can be employed to generate genotypic, phenotypic, and behavioral diversity. In particular, noise-driven solutions are expected to prevail in cellular adaptation to variable environments [1, 2]. It has been proposed that biological systems have built-in molecular devices for noise control [1, 2]. These mechanisms are of specific importance in developing organisms [2]. This view is supported by experimental findings demonstrating that noise is down-regulated in embryonic stem cells [3] and fluctuations of Nanog predispose cells towards differentiation [4]. The results of the present study suggest that noise regulation can be an effective strategy in stem cell differentiation.

Stem cells are characterized by their ability to self-maintain and generate differentiated cell types and functional tissues. Moreover, they show flexibility and reversibility in their use of these options [5]. Populations derived from these cells, subsequently denoted as ‘stem cell populations’, comprise stem cells, progenitors, and differentiated cells. The structure of these populations is strongly influenced by environmental factors such as specific cell-cell interactions [6], growth factor and oxygen supply [7], as well as the geometry and mechanical properties of the local environment [8]. Changing these factors results in either cell death or adaptation within days [9, 10]. Recently, progress has been made in the modeling and understanding of these processes on different levels of complexity [11, 12].

Our previous studies on stem cell population dynamics focused on the reversibility and stochasticity of cellular fate decisions [13]. In the model of Roeder et al. [12] individual cells gain and loose stem cell properties depending on whether they localize inside or outside a specific niche environment, respectively. Thus, the environment directs the cellular fate and the reversibility of cell fate decisions is enabled by probabilistic switches between different micro-environments. The model well described several experimental data sets on the *in vivo* organization of normal and malignant hematopoietic stem cell populations [12]. However, even within homogeneous *in vitro* environments stem cells are capable of expanding and maintaining the aforementioned stem cell populations. For modeling these systems the present study expanded the ideas of Roeder et al. [12] by assuming that cells gain and loose stem cell properties according to a probabilistic process whose state-specific amplitudes are set by the environment. Within this approach cell fate decisions are basically reversible. The assumed cell state fluctuations can be hypothesized to be generated by intra- and extracellular noise triggering random transitions between different regulatory network activation patterns. This concept is in agreement with experimental findings demonstrating that epigenetic gene silencing, known to be instrumental in cell differentiation and fate control, has a strong stochastic component [14].

The regularity of biological development in spite of the ubiquitous presence of noise has raised the concept of a ‘potential energy landscape’ or ’attractor landscape’ explaining cell differentiation and phenotypic diversification in terms of non-linear systems theory and equilibrium thermodynamics [15]. In this concept, cells visit their accessible states driven by the potential gradient and non-state-specific so-called additive noise. Potential minima constitute attractive states corresponding to population density maxima in equilibrium. The alternative concept put forward in the present study assumes

that noise is predominant in most parts of the cellular state space. Its essence is that the population density is determined by state-specific so-called multiplicative noise forming a 'noise landscape', with low noise states representing the attractive states. Cells subjected to an environment not matching their internal state are assumed to be destabilized by a high noise amplitude. They subsequently adapt to this environment by traveling towards low noise states.

2. MODEL

The present study focuses on the degree of differentiation as the basic cellular attribute of interest. It is defined as the loss of stem cell properties and goes along with but is not identical to lineage commitment. Cell differentiation is quantified by a variable α taking values between zero (full stem cell potential) and one (complete cell differentiation). Each value of α may stand for a set of regulatory network activation patterns. Physically, α depends on the abundance and sub-cellular localization of proteins and RNAs, as well as other types of signaling and metabolic molecules. The α-dynamics of a single cell can be modeled according to a one-dimensional Langevin equation:

$$\frac{d\alpha}{dt} = f(\alpha) + g(\alpha)\xi(t)\,, \qquad (1)$$

with $f(\alpha)$ representing the deterministic part of the dynamics and $g(\alpha)\xi(t)$ denoting the usual Gaussian white noise term ($<\xi(t)>=0$, $<\xi(t)\xi(t')>=\delta(t-t')$). In applying Equation 1 one may focus on deterministically dominated ($|f(\alpha)| > g(\alpha)$) or noise modulation-dominated ($|f(\alpha)| < g(\alpha)$) dynamics, both of which can give the same equilibrium distribution of α-values when sampled over time. In the following, we concentrate on noise modulation-dominated dynamics. Carrying the predominance of noise to an extreme we completely neglect any deterministic dynamics in our model ($f(\alpha) = 0$, corresponding to globally equivalent deterministic potential energy states) and simulate stem cell differentiation as a result of noise modulation alone.

In order to simulate population dynamics in terms of the number of cells in state α we transfer the general ideas of the Langevin approach equation (1) to a classical population dynamics model which is similar in structure to a master equation for a composite Markov process [16]. The model assumes each cell's α-value to randomly fluctuate according to a state-specific noise amplitude $\sigma(\alpha)$. Starting from an initial value α a cell assumes a new value α' drawn from a Gaussian distribution $p(\alpha'|\alpha)$ that is centered around α and has standard deviation $\sigma(\alpha)$. The frequency of this random transfer is determined by the randomization rate $R(\alpha)$ defining the number of random events per time. We assume $R(\alpha)$ to increase linearly with the cell proliferation rate $r(\alpha)$ accounting for cell division as a major source of randomization [17]. Finally, the dynamics of the average number of cells $N(\alpha)$ in state α is governed by the random transfer towards and away from α, and by cell proliferation:

$$\frac{\partial N(\alpha)}{\partial t} = \int_0^1 p(\alpha \mid \alpha')\, R(\alpha')\, N(\alpha')\, d\alpha' - R(\alpha)\, N(\alpha) + r(\alpha)\, N(\alpha) \qquad (2)$$

with

$$p(\alpha \mid \overline{\alpha}) \propto \exp\left(-\frac{(\alpha-\overline{\alpha})^2}{2\sigma^2(\overline{\alpha})}\right) \quad \text{and} \quad R(\alpha) = R_0 + R_1\, r(\alpha). \qquad (3)$$

As a consequence of experimental findings we replaced the proliferation term in equation (2) by the cell cycle model of León et al. [18] assuming cell cycle progression to be a multi-step process (five cell cycle steps were used in all simulations). Figure 1 illustrates the general principle of state-specific (multiplicative) noise-driven dynamics. Directional movement results from differences in the step sizes of successive forward and backward jumps.

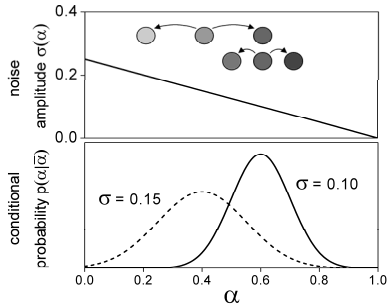


Figure 1. Multiplicative noise-driven dynamics. Upper panel: state-specific noise amplitude (standard deviation) $\sigma(\alpha)$ of the Gaussian conditional probability density function (cpdf) $p(\alpha'|\alpha)$ assumed to be a linear decreasing function of α. The pictogram shows a cell with $\alpha = 0.4$ being scattered towards $\alpha = 0.2$ and 0.6, respectively (upper row). The subsequent scatter starting at $\alpha = 0.6$ has a smaller range (lower row). This results in an average rightward drift of the cell. Lower panel: Gaussian cpdf $p(\alpha|\alpha')$ as a function of α for $\alpha' = 0.4$ (left) and $\alpha' = 0.6$ (right). The corresponding standard deviations are $\sigma(0.4) = 0.15$ and $\sigma(0.6) = 0.10$.

2.1. Basic assumptions

Figure 2 shows simplified noise amplitudes $\sigma(\alpha)$ and proliferation rates $r(\alpha)$. The functional form of the noise amplitudes $\sigma(\alpha)$ is assumed to be determined by the environment. The stem cell maintaining environments S1 and S2 stabilize stem cell-like states with low α-values whereas the differentiation promoting environments D1 and D2 stabilize committed states with large values of α by the assignment of low noise levels. The noise amplitudes are assumed to be linear functions of α for sim-

plicity. Stem cells and differentiated cells are mostly quiescent whereas progenitors are proliferative. This is reflected by the bell-shaped proliferation rates $r(\alpha)$ being zero at the interval boundaries and assuming their maximum value $r_{max} > 0$ halfway in between. Under these assumptions and with $R(\alpha) > 0$, an initial distribution of α-values evolves towards a stationary distribution representing a growing cell population.

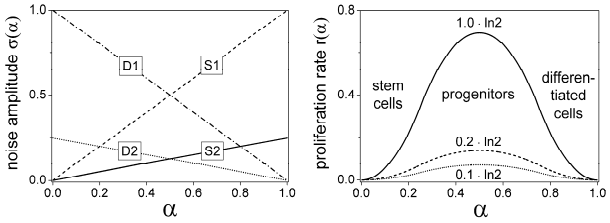


Figure 2. Noise amplitude $\sigma(\alpha)$ (left) and proliferation rate $r(\alpha)$ (right) as a function of cell differentiation α. The noise amplitude is shown for four idealized environments: i) two stem cell maintaining environments (S1 and S2) stabilizing stem cell states and ii) two differentiation promoting environments (D1 and D2) stabilizing differentiated states. The proliferation rate is zero (quiescence) at the interval boundaries for pure stem cells and differentiated cells, respectively, and assumes its highest values at intermediate α. The maximum proliferation rates $r_{max} = 0.1$, 0.2, and $1.0 \cdot \ln2$/d correspond to minimum cell cycle times of $\tau_{min} = 10$, 5, and 1 days, respectively.

3. RESULTS

In the following, the general behavior of our model is illustrated for different parameter settings. The displayed graphs characterize the respective stem cell populations in terms of the numerically calculated relative frequencies $P(\alpha_i) = N(\alpha_i)/\sum_j N(\alpha_j)$ with $N(\alpha_i)$ denoting the number of cells in the respective differentiation state interval centered at α_i.

3.1. Environmental adaptation

Figure 3 shows the adaptation dynamics for two cell populations being transferred from a stem cell maintaining environment S to a differentiation promoting environment D. The timescale of the equilibration processes is of the order of days consistent with experimental data [9, 10]. In both cases, the S and D environments fully stabilize pure stem cells ($\alpha = 0$) and differentiated cells ($\alpha = 1$), respectively. However, in the S2 and D2 environments these states can hardly be accessed dynamically because the associated cumulative sum of directed steps is too small on average. This dynamical hindrance together with the stronger stabilization of proliferative progenitor states results in equilibrium distributions that are peaked at intermediate α-values. Generally, extensive low noise domains of the state space can hardly be accessed from outside these domains.

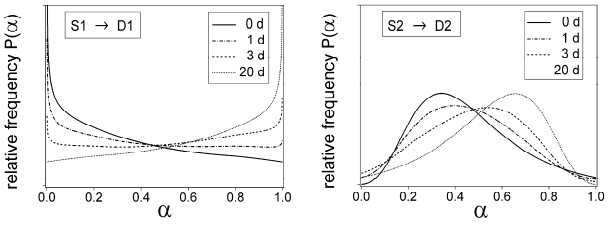


Figure 3. Adaptation dynamics of a stem cell population after instantaneous switching from a stem cell maintaining environment S to a differentiation promoting environment D. Left panel: S1 to D1. Right panel: S2 to D2. Snapshots are taken at the time of switching and 1, 3, and 20 days, respectively, after switching. $R_0 = 0.3$/d, $R_1 = 0.9$, $r_{max} = 1.0 \cdot \ln2$/d.

3.2. Randomization rate

The influence of the noise parameter R_1 on the frequency distribution of α-states is illustrated in the left panel of Figure 4 for the S2 environment. A high value of R_1 disperses the cells away from the most proliferating states around the mid-interval towards the noise-reduced states at low α-values. The effect of the background noise parameter R_0 is similar but without the state-specific modulation by the proliferation rate $r(\alpha)$. It drives the cells into the low-noise attractors when proliferation is down-regulated. This is demonstrated in the right panel of Figure 4 for different values of r_{max}. The equilibrium distribution of non-proliferating cells ($r_{max} = 0$) would be a delta peak at $\alpha = 0$. Conversely, in the absence of noise the population would converge to a delta peak at $\alpha = 0.5$ when starting from an equal distribution. In summary, randomization and proliferation act as antagonists in modulating the cell state distribution, with proliferation enabling the maintenance of subpopulations in environmentally unfavored states.

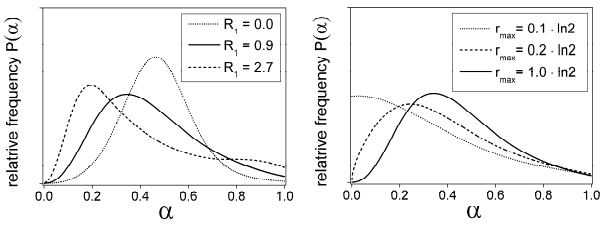


Figure 4. Impact of the noise parameter R_1 (left panel) and the maximum proliferation rate r_{max} (right panel) on the stationary distributions for the S2 environment. A high value of R_1 disperses the cells away from the central proliferation zone towards the more noise-reduced states. A small cellular growth as expressed by low values of r_{max} lets noise dominate over proliferation even in the presence of a dynamic hindrance in approaching the noise-reduced states (see text). The parameters are iden-

tical to those of Figure 3, except for R_1 in the left panel and r_{max} in the right panel.

4. DISCUSSION

Noise is ubiquitous in biological systems and must be controlled to ensure reliable cell functioning, at least in higher multicellular organisms that feature noise-sensitive processes like alternative splicing and epigenetic regulation of gene expression. Noise regulation is most economic if applied only to those cellular states that are relevant under the prevailing environmental conditions. Noise regulation is thus expected to dependent on the match between the internal state of a cell and its environment. In the present study we introduced a simple few-parameter model of stem cell and progenitor population dynamics that is explicitly based on noise regulation. It assumes that state-specific noise regulation in response to environmental signals serves as a selector of certain differentiation states representing specific functional cellular programs. This noise-driven selection scheme appears to be an economic general purpose mechanism for environmental adaptation and diversification.

5. CONCLUSIONS

In conclusion, we suggest that noise regulation can be effective in cellular development and environmental adaptation. It is expected to be relevant especially in higher multicellular organisms that comprise exposed noise-sensitive phenomena. Decoding the 'noise landscape' will be essential for the understanding of cell functioning.

6. ACKNOWLEDGMENTS

The authors would like to thank Hannah H Chang, Sui Huang, and Donald E Ingber for sharing their unique data set in order to demonstrate the predictive potential of our model (results not shown) and for helpful suggestions. Markus Loeffler, Ingo Roeder, and Ingmar Glauche contributed by critical discussion.

7. REFERENCES

[1] M. Acar, A. Becskei, and A. van Oudenaarden, "Enhancement of cellular memory by reducing stochastic transitions." *Nature*, vol. 435, pp. 228-32, 2005.

[2] A. M. Arias and P. Hayward, "Filtering transcriptional noise during development: concepts and mechanisms." *Nat Rev Genet*, vol. 7, pp. 34-44, 2006.

[3] H. Szutorisz, A. Georgiou, L. Tora, and N. Dillon, "The proteasome restricts permissive transcription at tissue-specific gene loci in embryonic stem cells." *Cell*, vol. 127, pp. 1375-88, 2006.

[4] I. Chambers, J. Silva, D. Colby, J. Nichols, B. Nijmeijer, M. Robertson, J. Vrana, K. Jones, L. Grotewold, and A. Smith, "Nanog safeguards pluripotency and mediates germline development." *Nature*, vol. 450, pp. 1230-4, 2007.

[5] M. Loeffler and I. Roeder, "Tissue stem cells: definition, plasticity, heterogeneity, self-organization and models--a conceptual approach." *Cells Tissues Organs*, vol. 171, pp. 8-26, 2002.

[6] A. Wilson and A. Trumpp, "Bone-marrow haematopoietic-stem-cell niches.," *Nat Rev Immunol*, vol. 6, pp. 93-106, 2006.

[7] W. L. Grayson, F. Zhao, R. Izadpanah, B. Bunnell, and T. Ma, "Effects of hypoxia on human mesenchymal stem cell expansion and plasticity in 3D constructs.," *J Cell Physiol*, vol. 207, pp. 331-9, 2006.

[8] D. E. Ingber, "Mechanical control of tissue morphogenesis during embryological development." *Int J Dev Biol*, vol. 50, pp. 255-66, 2006.

[9] H. H. Chang, P. Y. Oh, D. E. Ingber, and S. Huang, "Multistable and multistep dynamics in neutrophil differentiation." *BMC Cell Biol*, vol. 7, pp. 11, 2006.

[10] J. D. Gibbs, D. A. Liebermann, and B. Hoffman, "Terminal myeloid differentiation is uncoupled from cell cycle arrest." *Cell Cycle*, vol. 6, pp. 1205-9, 2007.

[11] M. Kaern, T. C. Elston, W. J. Blake, and J. J. Collins, "Stochasticity in gene expression: from theories to phenotypes." *Nat Rev Genet*, vol. 6, pp. 451-64, 2005.

[12] I. Roeder, M. Horn, I. Glauche, A. Hochhaus, M. C. Mueller, and M. Loeffler, "Dynamic modeling of imatinib-treated chronic myeloid leukemia: functional insights and clinical implications." *Nat Med*, vol. 12, pp. 1181-4, 2006.

[13] I. Roeder, K. Braesel, R. Lorenz, and M. Loeffler, "Stem cell fate analysis revisited: interpretation of individual clone dynamics in the light of a new paradigm of stem cell organization." *J Biomed Biotechnol*, vol. 2007, pp. 84656, 2007.

[14] E. Y. Xu, K. A. Zawadzki, and J. R. Broach, "Single-cell observations reveal intermediate transcriptional silencing states." *Mol Cell*, vol. 23, pp. 219-29, 2006.

[15] P. Ao, C. Kwon, and H. Qian, "On the existence of potential landscape in the evolution of complex systems." *Complexity*, vol. 12, pp. 19-27, 2007.

[16] N. G. van Kampen, *Stochastic processes in physics and chemistry.* Elsevier, Amsterdam., 2004.

[17] J. Beckmann, S. Scheitza, P. Wernet, J. C. Fischer, and B. Giebel, "Asymmetric cell division within the human hematopoietic stem and progenitor cell compartment: identification of asymmetrically segregating proteins." *Blood*, vol. 109, pp. 5494-501, 2007.

[18] K. Leon, J. Faro, and J. Carneiro, "A general mathematical framework to model generation structure in a population of asynchronously dividing cells." *J Theor Biol*, vol. 229, pp. 455-76, 2004.

MODELING IP_3 RECEPTOR FUNCTION USING STOCHASTIC APPROACHES

Jukka Intosalmi[1,2], Tiina Manninen[1,2], Katri Hituri[2], Keijo Ruohonen[1], and Marja-Leena Linne[2]

[1]Department of Mathematics, Tampere University of Technology,
[2]Department of Signal Processing, Tampere University of Technology,
P.O. Box 553, FI-33101 Tampere, Finland
jukka.intosalmi@tut.fi

ABSTRACT

The time evolution of chemical systems is traditionally modeled using deterministic ordinary differential equations. Chemical reactions, however, are random in nature, and the deterministic approach is valid only for a restricted class of systems. Stochastic models take random fluctuations into account and are thus more realistic. In this work, we simulate an inositol trisphosphate receptor model using ordinary differential equations, stochastic differential equations, and the Gillespie stochastic simulation algorithm. The main goal of this work is to study the applicability of these methods for a system containing small numbers of molecules and ions. We concentrate especially on the SDE approach and investigate how well it models systems with small numbers of chemical species.

1. INTRODUCTION

Biochemical reactions can be modeled stochastically using numerous different methods [1, 2]. An ideal model would have the following three important properties. First, the model should be as realistic as possible, second, the mathematical method should be easily implementable as a computer algorithm, and third, the algorithm should be computationally effective. Some realistic modeling approaches can be derived directly from chemical kinetics without making any approximations. Such approaches are called exact. A good example of an exact modeling approach is the stochastic simulation algorithm (SSA) developed by Gillespie [3, 4]. The SSA is applicable when the molecular populations in the system are small, but it becomes computationally inefficient when the numbers of molecules increase [4].

In order to construct stochastic models that can be effectively simulated, new mathematical approaches have to be explored. As an approximate method also stochastic differential equations (SDEs) have been considered a promising way to model biochemical reactions stochastically [5]. The SDE approach is attractive especially if we consider a system for which the SSA is computationally inefficient and the traditional deterministic ordinary differential equation (ODE) approach cannot be used as a good approximation.

In this study, we simulate the inositol trisphosphate receptor (IP_3R) model containing small numbers of chemical species. The SSA is evidently the most efficient modeling approach in this case. However, our goal is rather to study the typical characteristics of different approaches. This kind of knowledge is extremely valuable when the modeling approaches are applied for larger systems.

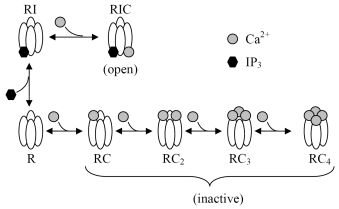


Figure 1. States and transitions of the IP_3R model.

2. SYSTEM AND METHODS

Several models have been proposed for the IP_3 receptor (for a review, see, e.g. [6]). In this study, we use the model of Doi et al. [7] which was originally published as a part of a larger model for calcium ion (Ca^{2+}) dynamics in the cerebellar Purkinje cell spine. The graphical illustration of the model is given in Figure 1. The transitions between the states are described by reversible chemical reactions of the form

$$\mathrm{A} + \mathrm{B} \underset{k_b}{\overset{k_f}{\rightleftharpoons}} \mathrm{C}, \tag{1}$$

where A, B, and C are chemical species, and k_f and k_b are *rate constants* for forward and backward reactions, respectively. The reactions of the model are given in Table 1. The rate constants of these reactions have been determined from experimental data [7]. The used volume of cytosol is $0.1\ \mu\mathrm{m}^3$. In the following, [X] denotes the concentration of species X.

The IP_3R model involves one open state (i.e. RIC). Once the IP_3R channel structure is open, Ca^{2+} flux from the endoplasmic reticulum (ER) to the cytosol starts. In this study, we model the Ca^{2+} flux using the differential equation

$$\begin{aligned}\frac{d[\mathrm{Ca}^{2+}]_{\mathrm{cyt}}}{dt} &= -\frac{d[\mathrm{Ca}^{2+}]_{\mathrm{ER}}}{dt} \\ &= k[\mathrm{RIC}]([\mathrm{Ca}^{2+}]_{\mathrm{ER}} - [\mathrm{Ca}^{2+}]_{\mathrm{cyt}}), \\ &\quad \text{when } [\mathrm{Ca}^{2+}]_{\mathrm{ER}} - [\mathrm{Ca}^{2+}]_{\mathrm{cyt}} > 0, \text{otherwise } 0,\end{aligned} \tag{2}$$

Table 1. Reversible reactions, reaction rates, and rate constants for the IP_3R model of Doi et al. [7]

	Reaction	Reaction rate	k_f	k_b
R_1	RI + $Ca^{2+} \underset{k_b^{R_1}}{\overset{k_f^{R_1}}{\rightleftharpoons}}$ RIC	$v_{R1} = k_f^{R_1}[RI][Ca^{2+}]_{cyt} - k_b^{R_1}[RIC]$	$8 \times 10^9 \frac{1}{Ms}$	$2000 \frac{1}{s}$
R_2	R + $IP_3 \underset{k_b^{R_2}}{\overset{k_f^{R_2}}{\rightleftharpoons}}$ RI	$v_{R_2} = k_f^{R_2}[R][IP_3] - k_b^{R_2}[RI]$	$10^9 \frac{1}{Ms}$	$25800 \frac{1}{s}$
R_3	R + $Ca^{2+} \underset{k_b^{R_3}}{\overset{k_f^{R_3}}{\rightleftharpoons}}$ RC	$v_{R_3} = k_f^{R_3}[R][Ca^{2+}]_{cyt} - k_b^{R_3}[RC]$	$8.889 \times 10^6 \frac{1}{Ms}$	$5 \frac{1}{s}$
R_4	RC + $Ca^{2+} \underset{k_b^{R_4}}{\overset{k_f^{R_4}}{\rightleftharpoons}}$ RC_2	$v_{R_4} = k_f^{R_4}[RC][Ca^{2+}]_{cyt} - k_b^{R_4}[RC_2]$	$2 \times 10^7 \frac{1}{Ms}$	$10 \frac{1}{s}$
R_5	RC_2 + $Ca^{2+} \underset{k_b^{R_5}}{\overset{k_f^{R_5}}{\rightleftharpoons}}$ RC_3	$v_{R_5} = k_f^{R_5}[RC_2][Ca^{2+}]_{cyt} - k_b^{R_5}[RC_3]$	$4 \times 10^7 \frac{1}{Ms}$	$15 \frac{1}{s}$
R_6	RC_3 + $Ca^{2+} \underset{k_b^{R_6}}{\overset{k_f^{R_6}}{\rightleftharpoons}}$ RC_4	$v_{R_6} = k_f^{R_6}[RC_3][Ca^{2+}]_{cyt} - k_b^{R_6}[RC_4]$	$6 \times 10^7 \frac{1}{Ms}$	$20 \frac{1}{s}$

where k is rate parameter, [RIC] is the concentration of open channels, and Ca^{2+} denotes calcium ions passing through the open channel. For k, we use the value $5.8 \times 10^8 \frac{1}{Ms}$, and the initial value for $[Ca^{2+}]_{ER}$ is 150 μM (cf. [8]).

2.1. Ordinary differential equation modeling

A set of chemical reactions can be modeled deterministically using the law of mass action and ODEs. According to the law of mass action, we can determine the *reaction rate* v of the reaction in Equation 1 by means of the equation

$$v = -\frac{d[A]}{dt} = -\frac{d[B]}{dt} = \frac{d[C]}{dt} = k_f[A][B] - k_b[C]. \quad (3)$$

If we consider a system of n species X_i, $i = 1, \ldots, n$, and m reactions R_j, $j = 1, \ldots, m$, the time evolution of the ith species is described by the equation

$$\frac{d[X_i]}{dt} = \sum_{j=1}^{m} s_{ij} v_j, \quad (4)$$

where s_{ij} is the stoichiometric coefficient and v_j is the reaction rate of the jth reaction. The stoichiometric coefficient $s_{ij} \in \mathbb{Z}$ describes how many molecules of a certain kind are involved in a certain reaction. It is positive if the amount of the molecule is increasing, negative if the amount is decreasing, and 0, if the amount is not changing in the reaction.

We now have a set of coupled ordinary differential equations that can be written in the form

$$\frac{d\mathbf{X}(t)}{dt} = \mathbf{S}\mathbf{v}(\mathbf{K}, \mathbf{X}(t)), \quad (5)$$

where $\mathbf{X}(t) : [0, \infty) \longrightarrow \mathbb{R}^n$ consists of the concentrations of the chemical species $X_i, i = 1, \ldots, n$, $\mathbf{v}(\mathbf{K}, \mathbf{X}) : \mathbb{R}^n \longrightarrow \mathbb{R}^m$ describes the reaction rates, $\mathbf{S} \in \mathbb{R}^{n \times m}$ is the stoichiometric matrix including the stoichiometric constants, and $\mathbf{K}$ is a vector including the rate constants.

2.2. Stochastic differential equation modeling

SDE modeling is based on the theory of stochastic integration. If we consider the n-dimensional deterministic ODE model introduced in Subsection 2.1, we can obtain an SDE model by incorporating an Itô integrable stochastic term in Equation 5. As a result, we have the equation

$$d\mathbf{X}(t) = \mathbf{S}\mathbf{v}(\mathbf{K}, \mathbf{X}(t))dt + \mathbf{S}\mathbf{P}\mathbf{V}(\mathbf{X}(t))d\mathbf{B}(t), \quad (6)$$

where $\mathbf{B}(t) \sim N(\mathbf{0}, t\mathbf{I})$ is the m-dimensional Brownian motion, $\mathbf{P} \in \mathbb{R}^{m \times m}$ is a diagonal matrix describing the parameters, $\mathbf{V} : \mathbb{R}^n \longrightarrow \mathbb{R}^{m \times m}$ is a diagonal matrix including reaction rates without rate constants, and $\mathbf{X}, \mathbf{S}$, and $\mathbf{v}$ are as in the ODE model described by Equation 5 [5]. If we want to incorporate randomness in each reaction rate constant separately, we just consider one reversible reaction as two separate non-reversible reactions and use the same technique as described above.

Equation 6, describing a stochastic process, can also be written in the form

$$\mathbf{X}(t) = \mathbf{X}_0 + \int_0^t \mathbf{S}\mathbf{v}ds + \int_0^t \mathbf{S}\mathbf{P}\mathbf{V}d\mathbf{B}(s), \quad (7)$$

where $\mathbf{X}_0$ is the initial state, the first integral is the Riemann integral, and the second integral is the Itô integral [9]. The expected value and the variance of this process are usually difficult to solve. Simulation studies are thus needed. Parameters included in $\mathbf{P}$ should be estimated using some estimation algorithm.

2.3. Stochastic simulation algorithm

The stochastic simulation algorithm (SSA) is a Monte Carlo procedure, which is used to generate numerically the time evolution of a chemically reacting system [3]. It treats chemical species discretely and simulates every reaction one at a time [3, 4]. In the following, the basic idea of the SSA is presented.

Let us consider the system of n species and m reactions introduced earlier in Subsections 2.1 and 2.2, and let $\mathbf{X}(t) : [0, \infty) \longrightarrow \mathbb{Z}^n$ be a vector containing the numbers of molecules of each species at time t. Each reaction R_j, $j = 1, \ldots, m$, in the system can be characterized by a *propensity function* $a_j(\mathbf{X})$ which depends on the current state of the system. A *state change vector* $\mathbf{v}_j \in \mathbb{Z}^n$ describes the stoichiometry of the reaction R_j. In the simulation algorithm, the propensity functions are used for determining the distributions of the next reaction to happen (j) and the time to the next reaction (τ). These distributions are then sampled and the state of the system is updated by state change vector. The SSA consists of the following steps:

1. Initialize the time $t = t_0$ and the state of the system $\mathbf{X}(t) = \mathbf{X}_0$.
2. Evaluate $a_j(\mathbf{X}(t)), j = 1, \ldots, m$, and $a_0(\mathbf{X}(t)) = \sum_{k=1}^{m} a_k(\mathbf{X}(t))$.
3. Generate two uniformly distributed random variables r_1 and r_2 and take $\tau = (1/a_0(\mathbf{X}(t))) \ln(1/r_1)$ and j such that $\sum_{k=1}^{j-1} a_k(\mathbf{X}(t)) < r_2 a_0(\mathbf{X}(t)) \leq \sum_{k=1}^{j} a_k(\mathbf{X}(t))$.
4. Replace $\mathbf{X}(t + \tau) = \mathbf{X}(t) + \mathbf{v}_j$ and $t = t + \tau$.
5. Return to step 2 or end the simulation.

3. RESULTS

We simulate the IP_3R model using ODEs, SDEs, and the SSA. All simulatios are run in MATLAB®. The Ca^{2+} flux described by Equation 2 is modeled simply as a part of the set of differential equations in the ODE and SDE implementations. In the SSA simulations, the flux is described as a forward reaction for which the propensity function is determined by the number of open channels and by the number of Ca^{2+} ions in the cytosol and ER.

3.1. ODE and SSA

When modeling biochemical systems, the selection of the model plays an important role. The model should describe the natural phenomenon as rigorously as possible, but ignore the details that are not essential for system level behavior. After a proper model has been selected, the next step is to choose the formalism to describe the model and find out how to implement the model as an algorithm.

Previous computational studies considering the IP_3R model show that the traditional ODE approach provides us with a satisfactory approximation only in the case in which the concentrations are relatively large (see e.g. [8]). When the numbers of chemical species are small, the relative amount of random fluctuations in the system is greater. In this case, we have to use modeling methods that are capable of taking these fluctuations into account. In the following, we concentrate on the cases in which stochastic methods are needed.

When the IP_3R model is simulated stochastically using the SSA, the results differ notably from the results of the ODE simulations (Figure 2(a)). The main reason for this is that the SSA simulation quite often leads to a closed receptor state. This means that there is no open channel for Ca^{2+} flux from the ER and thus the number of Ca^{2+} ions in the cytosol does not increase. The SSA simulations also support the intuitive assumption that the two reactions leading to the open state of the receptor are the most essential when the stochastic nature of the model is concerned.

It is clear that the SSA is the most efficient approach when it comes to computational time if the numbers of chemical species are small. However, it is also useful to study approximative methods in order to learn about their properties and behavior. It is clear that many continuous time approximations of the SSA cannot be applied. For example, the use of the chemical Langevin equation (CLE) requires certain conditions to be fulfilled [4]. First, several reactions must occur during one time step, and second, the time step should be small enough. When we take a closer look at our SSA simulations, we observe that both of these conditions cannot be satisfied at the same time.

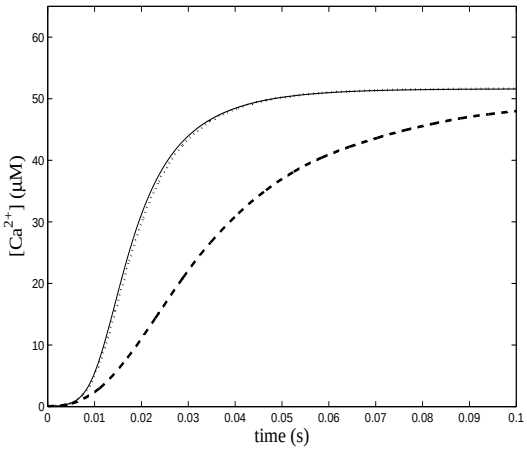
3.2. SDE

In biological systems, the concentrations of chemical species are often very small and the SDE modeling is thus challenging. The possibility of negative concentrations and the risk of an unstable model are always present. This means that although the model would be mathematically correct, it might not be biologically realistic. Therefore, the type of the SDE model, the model parameters, and the numerical method for solving the SDE have to be chosen carefully.

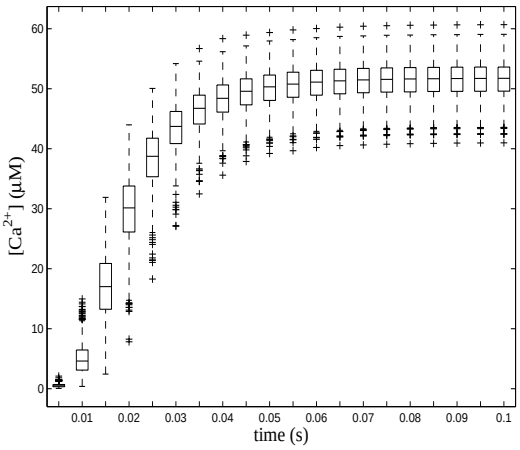
The SDE models tested in this study are built on the basis of the results obtained from the SSA simulations. As mentioned already in Subsection 3.1, the two reactions leading to the open state of the IP_3 receptor (R_1 and R_2 in Table 1) are the most significant when we study the Ca^{2+} levels in the system. When the SDE model is tuned so that randomness is incorporated only in these two reactions, the model is incabable of producing similar results as the SSA. The problem is that in order to avoid negative concentrations, we have to adjust the model parameters and the time step so that variance in the rate constants is very small. Thus, the system is always driven towards the open state and consequently the Ca^{2+} concentration in the cytosol increases. The same result is obtained if randomness is incorporated in all rate constants.

In addition to the two reactions leading to the open state, also the Ca^{2+} flux has an essential role in the model. When the whole model is constructed using the SDE, we are able to allow a greater variance of the fluctuations in the rate parameter of the flux. The drawback of this approach is that random fluctuations in the flux overpower the fluctuations in the other rate constants. This shows that the same results can be obtained using an SDE model in which randomness is incorporated only in the flux.

In order to illustrate the results, we show in Figure 2(a) the sample mean of $[Ca^{2+}]$ from thousand SSA and SDE model (randomness only in the flux) runs, and the deterministic ODE model response. In the simulations, the initial concentrations for Ca^{2+}, IP_3, and R were 0.05



(a) Sample mean of Ca^{2+} concentration.



(b) Boxplot illustration of the distribution of SDE paths.

Figure 2. (a) Sample mean of Ca^{2+} concentration in IP_3R model simulated with SDE ($\cdots$) and SSA ($--$), and deterministic response of the ODE (—). (b) Boxplot illustration of the distribution of SDE paths.

μM, 0.2 μM, and 0.2657 μM, respectively. Other initial concentrations were equal to zero. We see clearly that the SSA differs from the deterministic response, whereas the SDE model converges to it. Figure 2(b) illustrates the distribution of the solution of the SDE model. Similar analysis for the SSA reveals the great variance of the SSA paths (not shown). The deterministic response is solved numerically using the Euler method with time step 2×10^{-6} s and the SDE model is simulated using the Euler-Maruyama method with the same time step.

4. CONCLUSION

In this study, three approaches to the modeling of chemically reacting systems are introduced. The modeling approaches, namely the deterministic differential equation modeling, stochastic differential equation modeling, and the stochastic simulation algorithm, are then applied in the modeling of an IP_3 receptor model. The simulations show that when the numbers of molecules in the system are small, realistic results can be obtained only using stochastic modeling approaches. In addition, it is concluded that stochastic differential equation modeling might lead to an unstable model when the numbers of molecules are small.

5. ACKNOWLEDGMENTS

This work was supported by the Academy of Finland, project nos 213462 (Finnish Programme for Centres of Excellence in Research 2006-2011), 106030, and 124615, as well as Tampere Graduate School in Information Science and Engineering (TISE) and Tampere University of Technology Graduate School.

6. REFERENCES

[1] T. E. Turner, S. Schnell, and K. Burrage, "Stochastic approaches for modelling in vivo reactions," *Comput. Biol. Chem.*, vol. 28, pp. 165–178, 2004.

[2] T. Manninen, *Stochastic methods for modeling intracellular signaling*, Ph.D. thesis, Department of Science and Engineering, Tampere University of Technology, Tampere, Finland, 2007.

[3] D. T. Gillespie, "A general method for numerically simulating the stochastic time evolution of coupled chemical reactions," *J. Comput. Phys.*, vol. 22, no. 4, pp. 403–434, 1976.

[4] D. T. Gillespie, "Stochastic chemical kinetics," in *Handbook of Materials Modeling*, S. Yip, Ed., pp. 1735–1752. Springer, Dordrecht, 2005.

[5] T. Manninen, M.-L. Linne, and K. Ruohonen, "Developing Itô stochastic differential equation models for neuronal signal transduction pathways," *Comput. Biol. Chem.*, vol. 30, no. 4, pp. 280–291, 2006.

[6] J. Sneyd and M. Falcke, "Models of the inositol trisphosphate receptor," *Prog. Biophys. Mol. Biol.*, vol. 89, pp. 207–245, 2005.

[7] T. Doi, S. Kuroda, T. Michikawa, and M. Kawato, "Inositol 1,4,5-trisphosphate-dependent Ca^{2+} threshold dynamics detect spike timing in cerebellar Purkinje cells," *J. Neurosci.*, vol. 25, no. 4, pp. 950–961, 2005.

[8] K. Hituri, "Simulation of IP_3 receptor function in cerebellar Purkinje cell dendritic spine: Importance of stochasticity," M.S. thesis, Faculty of Medicine, Institute of Medical Technology, University of Tampere, Tampere, Finland, 2007.

[9] B. Øksendal, *Stochastic Differential Equations: an Introduction with Applications*, Springer-Verlag, Berlin, 6th edition, 2007.

A SPATIAL SIRS BOOLEAN NETWORK MODEL FOR THE SPREAD OF H5N1 AVIAN INFLUENZA VIRUS AMONG POULTRY FARMS

Alexander Kasyanov[1], *Leona Kirkland*[2], *and Mihaela Teodora Matache*[3]

[1]Laboratory for Avian Influenza and Poultry Disease Epidemiology, Federal Centre for Animal Health, 600901 Yur'evets, Vladimir, Russia,
[2]GR Exypnos, Inc., 6426 S 164th Ave., Omaha, NE 68135, USA,
[3]Department of Mathematics, University of Nebraska at Omaha, Omaha, NE 68182, USA
kasjanov@arriah.ru, lkirkland@exypnos.org, dmatache@mail.unomaha.edu

ABSTRACT

To predict the spread of Avian Influenza we propose a synchronous Susceptible-Infected-Recovered-Susceptible (SIRS) Boolean network of poultry farms, using probabilistic Boolean rules. Gravity models from transportation theory are used for the probability of infection of a node in one time step, taking into account farm sizes, distances between farms, and mean distance travelled by birds. Basic reproduction numbers are computed analytically and numerically. The dynamics of the network are analyzed and various statistics considered such as number of infected nodes or time until eradication of the epidemic. We conclude that mostly when large farms (eventually) become infected the epidemic is more encompassing, but for a farm that does not have a very large poultry population, the epidemic could be contained.

1. INTRODUCTION

The spread of Highly Pathogenic Avian Influenza (HPAI) H5N1 viruses across Asian and European countries has devastated domestic poultry industries. The development of strategies to moderate the spread of influenza among poultry flocks and humans is a top government priority. To investigate the spread of HPAI between poultry farms we propose a Susceptible-Infected-Recovered-Susceptible (SIRS) Boolean network model.

Various individual-based models have been successful in modelling real-world epidemics and understanding mechanisms of epidemic outbreaks [1]. The field of complex networks has now been recognized as an important line of study for epidemiology. For example, Barthélemy et.al. [2], or May and Lloyd [3], have published a number of papers on epidemics in scale-free networks. A large class of physical, biological, chemical networks have been modelled as Boolean networks in recent years (e.g. [4], [5], [6], [7]). General interest in Boolean networks and their applications started much earlier with publications such as the one by Kauffman [8], whose work on the self-organization and adaptation in complex systems has inspired many other research studies.

The spread of HPAI among poultry farms has not yet been investigated in the context of Boolean networks. We propose a new model in which each farm is considered a node of a Boolean network and can be in one of two states, "infected" or "not infected" by the disease. A node can become infected based on the number of other infected nodes in its neighborhood, their distance from the node under consideration (small distances allow for an easier spread of infection through wild bird or workers interaction of neighboring farms through the common market places), the size of the nodes (large farms have a bigger chance being infected through the synanthropic birds interaction and humans and equipment movement), and the distance travelled by birds in one time step. To define the probability of infection in one time step we use an approach similar to Xia et. al. [9] who have implemented a gravity model from transportation theory [10] to epidemiological coupling and dynamics using a transient force of infection exerted by infecteds in one location on susceptibles in a different location, proportional to the number of susceptibles and the number of infecteds, and inverse proportional to the distance between the locations. This is similar to Newton's gravitational law.

2. THE BOOLEAN NETWORK MODEL

In this section we describe the SIRS Boolean model. Consider a network with N nodes (farms). Each node c_n can take on two values 0 (not infected) or 1 (infected). The synchronous evolution of the nodes from time t to time $t+1$ is given by a Boolean rule which is considered the same for all nodes, but depends on varying parameters from one node to another. Initially all the nodes are considered susceptible (S). If a node is infected (I), it undergoes a period of cleaning and quarantine during which it could spread the disease to other nodes in its neighborhood; however the force of infection decreases with time, and the node recovers (R) completely eventually. After the quarantine the node becomes again susceptible (S), unless it goes out of business.

Let $c_n(t)$ be the value of the node c_n at time t. Define the Boolean rule

$$c_n(t+1) = X(t) \cdot \chi_{\{0\}}(c_n(t)) + Y(t) \cdot \chi_{\{1\}}(c_n(t)) \quad (1)$$

where $X(t)$ is a Bernoulli random variable X with param-

eter $p_n(t)$ representing the probability that the susceptible node c_n becomes infected at time t, and $Y(t) = 1$ if the node is infectious at time $t + 1$, and $Y(t) = 0$ if the node is noninfectious at time $t + 1$. Here $\chi_{\{a\}}(b) = 1$ if $a = b$ and zero otherwise. If $c_n(t)$ becomes 0 at time t during quarantine, then we set $Y(t) = 0$ automatically until the end of quarantine. On the other hand, if a node goes out of business after an infection, then $c_n = 0$ permanently. To define $p_n(t)$ let B_n denote the size of the node c_n, that is B_n is the number of poultry at location c_n. Let $\hat{c}_n$ denote the collection of all the farms in the neighborhood of node c_n (excluding the node itself). Then $B(n) = \sum_{c_k \in \hat{c}_n} B_k$ is the total number of poultry in the neighborhood of node c_n. Let d_{nk} denote the physical distance between nodes c_n and c_k, with $k \in \{1, 2, 3, \ldots, N\}$. We define the probability $p_n(t)$ that the node c_n becomes infected at time t as follows:

$$p_n(t) = \sum_{c_k \in \hat{c}_n} c_k(t) \left(B_k/B(n)\right)^{\tau} \frac{1}{1 + (d_{nk}/d_0)^{\rho}} f(t). \tag{2}$$

Here d_0 represents the mean distance the infected wild birds are able to cover in one time step. The function $f(t) \in [0, 1]$ is a random factor that accounts for a reduction of the probability of infection from the infectious node c_k while cleaning and disinfection take place. The factor $B_k/B(n) = B_n B_k / \sum_{c_k \in \hat{c}_n} B_n B_k$ is a version of the size terms and $1/\left(1 + (d_{nk}/d_0)^{\rho}\right)$ is a version of a distance kernel in the gravity model of [9]. Here τ determines how the transient "emigration" probability scales with the donor population size, while ρ quantifies how attraction decays with distance.

In the next sections we analyze the actual network of farms and discuss the parameters of the model. Then we study the evolution of the disease in the network and we compute some related statistics.

3. THE NETWORK OF FARMS

Information regarding the poultry farms are taken from the National Agriculture Statistics Service USDA (www.nass.usda.gov) and topographic maps ($1 : 100,000$ Digital Raster Graphics; Conservation and Survey Division; School of Natural Resources; University of Nebraska-Lincoln). To simulate a network of farms we identify the geographical center of each county and compute the distances between these centers. We approximate each county by a square centered at the county center. In each square we apply a uniform geographical spread of the farms. The size of each farm is obtained as a random number from a Poisson distribution with mean equal to the average number of poultry per farm in each county. In Figure 1, we provide a network of 1198 poultry farms generated as above. This network is used further in the paper. We observe that Butler county accounts for about 63% and Polk county for about 32% of the poultry population of Nebraska.

We provide a boxplot for the node sizes in Figure 2 (a). The frequencies of the distances between nodes are in Figure 2 (b).

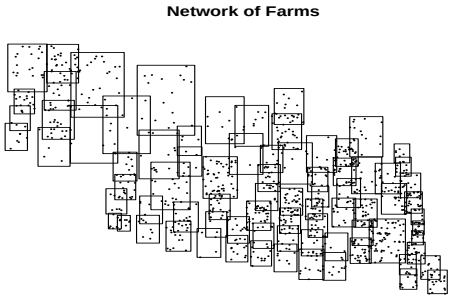


Figure 1. Geographical spread of the network.

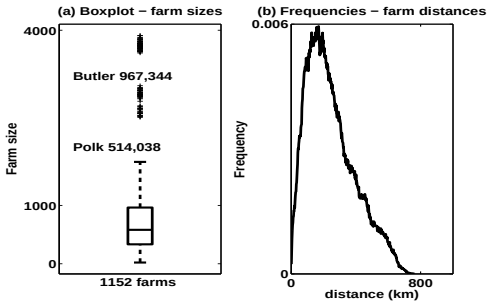


Figure 2. Boxplot of node sizes and distance frequencies. Most of the nodes have a rather small size, except farms in Butler and Polk counties, provided separately.

4. THE PARAMETERS OF THE MODEL

Recent studies have shown that the human influenza has an average time interval from infection of one individual to when their contacts are infected of about two days [11]. This number has been used by various authors in assessing the potential impact of a human pandemic of HPAI. At the same time time two days is the minimum time needed to get preliminary results back from a diagnostic center for HPAI. Consequently we assume that the basic time step is two days, so the infections within a time step are secondary cases from infected nodes in the previous time step within a neighborhood. The basic neighborhood is considered a circle of radius R km centered at each node. Given an infected node, the probability that it will infect nodes outside its neighborhood is equal to zero. We will use mostly $R = 100$ km, but the impact of the value of R will be considered in the analysis. The parameters τ and ρ will be varied to understand the impact of how the transient "emigration" probability scales with the donor population size, and how attraction decays with distance. We do not posses real data to estimate these parameters. The parameter d_0 is roughly estimated to 3.1 km from available information on home ranges for permanent resident birds and migrating birds of Nebraska. However, due to incomplete data, we believe that this number underestimates the true value of d_0 and therefore we use values of $d_0 \geq 10$ km.

The government quarantines an infected location for $Q = 21$ time steps as specified in the USDA national response plan. During this process the disease can still spread to other locations due to migration of synanthropic

birds, rodents, humans and equipment movement, but the probability of infection decreases with time. To account for this, the random factor $f(t)$ in formula 2 is set equal to 1 during the first time step after infection, and is subsequently given for all nodes by a Beta distribution $\beta(1, h(T))$ where T is the number of time steps since the beginning of the quarantine, and $h(T)$ is an increasing function of T ($h(T) = T$ in simulations). We set $c_n(t) = 0$ after 15 time steps of quarantine. After the quarantine the node re-enters the normal process if the location is repopulated. Small farms are assumed to have a 50% chance of going out of business versus repopulation.

Next we provide a formula for computing the basic reproduction numbers and generate simulations that allow us to understand the impact of a change in parameters on this quantity.

5. BASIC REPRODUCTION NUMBERS

Consider now the infection probability given by formula 2, used to compute the basic reproduction numbers, or the average amount of secondary infections generated by a primary infection. We assume that exactly one node, say c_K, is infected at time $t = 0$, that is $c_K(0) = 1$. We want to see what is the distribution of the number of infected nodes at time $t = 1$.

Let c_n be a node in the neighborhood of node c_K. Then $p_n = (B_K/B(n))^\tau / (1 + (d_{nK}/d_0)^\rho)$. This is the probability that $c_n(1) = 1$ given that $c_K(0) = 1$ and $c_k(0) = 0$, for all nodes $c_k, k \neq K$. Thus, at time $t = 1$ the number m of nodes that are turned ON can vary from 0 to the number M_K of nodes in the neighborhood of node c_K (not including the node c_K). So if $p_1, p_2, \ldots, p_{M_K}$ are the probabilities corresponding to the M_K nodes of the neighborhood of node c_K, then the probability $q_K(m)$ that exactly m nodes are infected at time $t = 1$ is given by $q_K(m) = \sum p_{i_1} p_{i_2} \ldots p_{i_m} \prod_j (1 - p_j)$, where the sum is over all possible combinations of m nodes out of M_K, $1 \leq i_1 < i_2 < \cdots < i_m \leq M_K$, and $j = 1, 2, \ldots, M_K, j \neq i_l, l = 1, \ldots, m$. Thus the random variable giving the number of nodes that are infected at time $t = 1$ is: $P(m \text{ infected nodes}) = q_K(m), m = 0, 1, \ldots, M_K$. Then $\sum_{m=0}^{M_K} q_K(m) = 1$ by the following result.

Remark 2. For any integer $k > 0$ and any real numbers $a_1, a_2, \ldots, a_k$, we have that $\sum_{l=0}^{k} \sum a_{i_1} a_{i_2} \ldots a_{i_l} \cdot \prod_j (1 - a_j) = 1$, where the sum is over $1 \leq i_1 < i_2 < \cdots < i_l \leq k$, and $j = 1, 2, \ldots, k, j \neq i_n, n = 1, 2, \ldots, l$. We make the convention that $l = 0$ means that there is only one term in the inside sum and all the factors of this term are of the type $(1 - a_j)$.

Proof: The proof is by induction on k. Let S_k denote the sum in Remark 2. Clearly $S_1 = a_1 + (1 - a_1) = 1$. Also $S_{k+1} = S_k \cdot a_{k+1} + S_k \cdot (1 - a_{k+1}) = S_k$. ♢

Then the average number of infected nodes given only one infected node at time $t = 0$, $c_K(0) = 1$, is $A_K = \sum_{m=0}^{M_K} m \cdot q_K(m)$.

Remark 3. For any integer $k > 0$ and any real numbers $a_1, a_2, \ldots, a_k$, we have that $\sum_{l=0}^{k} \; l \sum a_{i_1} a_{i_2} \ldots a_{i_l} \cdot \prod_j (1 - a_j) = a_1 + a_2 + \cdots + a_k$, where the sum is over $1 \leq i_1 < i_2 < \cdots < i_l \leq k$, and $j = 1, 2, \ldots, k, j \neq i_n, n = 1, 2, \ldots, l$.

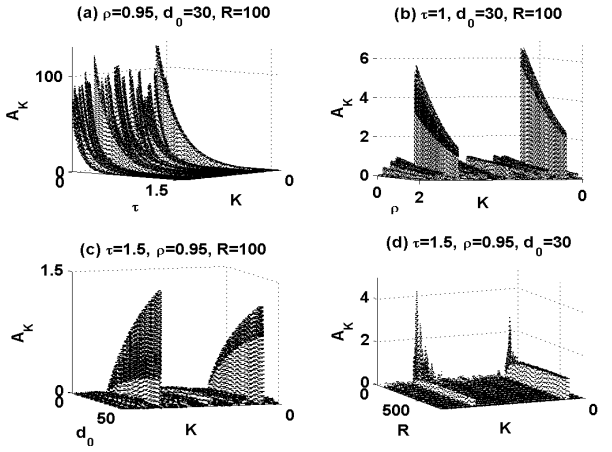


Figure 3. Plot of the average number of infected nodes in one time step, A_K, versus K, the index of the initial infected node. This is done in four different scenarios corresponding to the variation of one of the parameters ((a) τ, (b) ρ, (c) d_0, (d) R) while keeping the other ones fixed as mentioned in the titles of the subplots. Observe that A_K is decreasing as a function of τ, ρ or R, and increasing as a function of d_0. The two peaks correspond to Butler and Polk counties. The values of A_K are impacted most dramatically by changes in τ. When R increases there is an approximate threshold value beyond which the neighborhood size makes no difference since far away farms will not be infected.

Proof: The proof is by induction on k. Let S_k denote the sum in Remark 3. Observe that $S_1 = 0 \cdot (1 - a_1) + 1 \cdot a_1 = a_1$. Also $S_{k+1} = S_k \cdot (1 - a_{k+1}) + S_k \cdot a_{k+1} + a_{k+1} \cdot \sum_{l=0}^{k} \sum a_{i_1} a_{i_2} \ldots a_{i_l} \prod_j (1 - a_j)$ where the second sum is over $1 \leq i_1 < i_2 < \cdots < i_l \leq k$, and $j = 1, 2, \ldots, k, j \neq i_n, n = 1, 2, \ldots, l$. Thus, $S_{k+1} = S_k + a_{k+1} \cdot 1 = a_1 + a_2 + \cdots + a_{k+1}$. ♢

So the average number of nodes infected at time $t = 1$ or the basic reproduction numbers given $c_K(0) = 1$ are

$$A_K = p_1 + p_2 + \cdots + p_{M_K} \qquad K = 1, 2, \ldots N. \quad (3)$$

We graph A_K versus K and one other parameter (τ, ρ, d_0, and R respectively) in Figure 3. A modification of the fixed parameters mentioned in the titles of the plots does not change the shape of the graphs, only the values of A_K. For example, when τ is varied, an increase in the fixed ρ generates overall smaller values of A_K due to the fact that the distance kernel in formula 2 decreases.

Now we can focus on one parameter combination and analyze the average number of infected nodes by time steps and time until eradication of the epidemic.

6. NETWORK EVOLUTION AND SOME STATISTICS

We set the parameters as follows: $\tau = 1.5, \rho = 0.95, d_0 = 30$ km, and $R = 100$ km which yields an average of 195 farms per neighborhood. In the next graph we list the nodes horizontally (in the alphabetic order of the counties) and represent the infected ones by dots. We iterate formula 2 exactly 50 time steps. In Figure 4 we start with one infected node in Butler county and we plot dots for

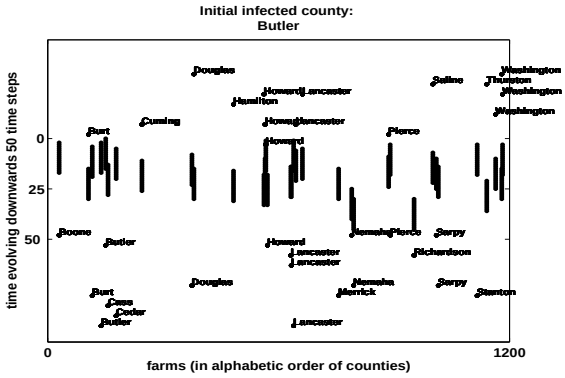


Figure 4. Sample spread of the infection starting with one infected node in Butler county. The infected nodes are listed: Butler, Howard, Lancaster, and Pierce counties, followed by the rest of the listed counties at various times.

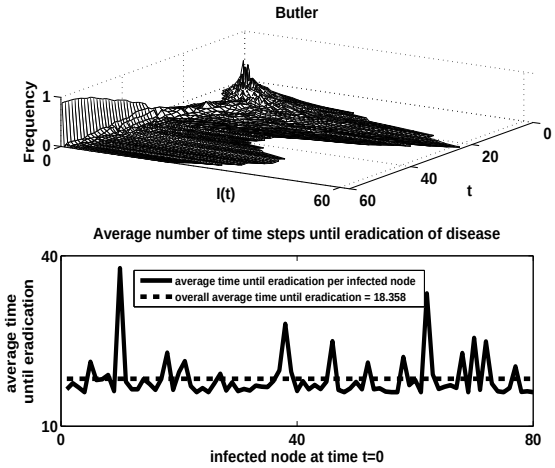


Figure 5. Frequency of $I(t), t = 1, 2, \ldots, 60$ when the initial infected nodes are in Butler county. For small and large values of t less nodes are infected as expected. Around $t = 20$ which is the time around which the first infections are cured larger values of $I(t)$ are more frequent. The second graph represents the average number of time steps for eradication of the network infection versus the initial infected node. The overall average is represented by a horizontal line.

all the nodes infected at each time step listing their names. Note that Butler is followed by Howard, Lancaster, Pierce, and then the rest of the listed counties at various times. A total of 34 nodes are infected and the infection is contained during the 50 time steps.

The quantity $I(t) = \sum_{n=1}^{N} c_n(t)$ is the number of infected nodes at time t. We generate frequency plots of $I(t)$ for $t = 1, 2, \ldots 60$, starting with one infected county. For example the first graph of Figure 5 corresponds to the infection of Butler. We observe that small and large values of t correspond to mostly small values of $I(t)$, while for medium values of t there are higher frequencies of larger values of $I(t)$. The epidemic may not be contained. For smaller counties, the plots are concentrated around small values of $I(t)$ for all t.

Now consider the time until the eradication of the disease starting with one infection, averaged over multiple sample evolutions. The results are in the second graph of Figure 5. The two peaks are for Butler and Polk counties. The overall network average is about 18 time steps.

We note that the infection of small counties has little impact on the network, unless they are close enough to one of the bigger nodes. When the spread of the disease is more encompassing, the bigger nodes are infected and spread the disease to other nodes faster and throughout a wider area. However, for a medium node the disease could be contained rather fast. On the other hand, it could be that even small nodes spread the disease to bigger nodes and produce an outbreak. However, for most cases the infection spreads to only a few or no other nodes.

7. REFERENCES

[1] H. W. Hethcote, "The mathematics of infectious diseases," *SIAM Review*, vol. 42(4), pp. 599–653, 2000.

[2] M. Barthélemy, R. Pastor-Satorras, and A. Vespignani, "Dynamical patterns of epidemic outbreaks in complex heterogeneous networks," *Journal of Theoretical Biology*, vol. 235, pp. 275–288, 2005.

[3] M. E. J. May and A. L. Lloyd, "Infection dynamics on scale-free networks," *Physical Review E*, vol. 64, pp. 066112, 2001.

[4] R. Albert and A.-L. Barabasi, "Dynamics of complex systems: scaling laws for the period of boolean networks," *Physical Review Letters*, vol. 84(24), pp. 5660–5663, 2000.

[5] M. T. Matache and J. Heidel, "Asynchronous random boolean network model based on elementary cellular automata rule 126," *Physical Review E*, vol. 71, pp. 026232, 2005.

[6] C. Goodrich and M. T. Matache, "The stabilizing effect of noise on the dynamics of a boolean network," *Physica A*, vol. 379, pp. 334–356, 2007.

[7] H. Lahdesmaki, S. Hautaniemi, I. Shmulevich, and O. Yli-Harja, "Relationships between probabilistic boolean networks and dynamic bayesian networks as models of gene regulatory networks," *Signal Processing*, vol. 86(4), pp. 814–834, 2006.

[8] S. A. Kauffman, *The origins of order*, Oxford University Press, Oxford, 1993.

[9] Y. Xia, O. N. Bjørnstad, and B. T. Grenfell, "Measels metapopulation dynamics: a gravity model for epidemiological coupling and dynamics," *The American Naturalist*, vol. 164, pp. 267–281, 2004.

[10] S. Erlander and N. F. Stewart, *The gravity model in transporation analysis: theory and extensions*, International Science, Netherlands, 1990.

[11] N. M. Ferguson, D. A. T. Cummings, S. Cauchemez, C. Fraser, S. Riley, A. Meeyai, S. Iamsirithaworn, and D. S. Burke, "Strategies for containing an emerging influenza pandemic in southeast asia," *Nature articles*, vol. 437, pp. 209–214, 2005.

RELATIONSHIPS BETWEEN GENETIC ABERRATIONS AND GENE EXPRESSION LEVELS IN GASTROINTESTINAL TRACT TUMORS

Virpi Kivinen[1], *Matti Nykter*[1,2], *Antti Ylipää*[1], *Limei Hu*[3], *David Cogdell*[3], *Kelly Hunt*[4], *Wei Zhang*[3], *and Olli Yli-Harja*[1]

[1]Department of Signal Processing, Tampere University of Technology, Tampere, Finland
P.O. Box 553, FI-33101 Tampere, Finland
[2] Institute for Systems Biology, Seattle, USA
[3]Department of Pathology and [4]Surgical Oncology, The University of Texas M.D. Anderson Cancer Center, Houston, Texas, USA
{virpi.kivinen, matti.nykter, antti.ylipaa, olli.yli-harja}@tut.fi
{lhu, dcogdell, khunt, wzhang}@mdanderson.org

ABSTRACT

Gastrointestinal stromal tumor (GIST) and leiomyosarcoma (LMS) are both tumors of gastrointestinal tract. Genetic events leading to these malignancies are not yet fully understood. In this paper, we analyze two types of genomic data, collected from GIST and LMS primary tumors. Data from gene expression and array comparative genomic hybridization (aCGH) measurements are co-analyzed by studying their correlations and identifying differentially expressed genes. Relationships between gene expression and genomic aberrations provide an insight into the underlying genetic events. We use gene ontology enrichment analysis to identify the biological processes that are affected by both a copy number aberration in the DNA and differential expression in the gene expression level.

1. INTRODUCTION

In many cases cancer development initiates from alterations in the function of key regulatory processes in cells. Accumulation of genetic aberrations in chromosomal segments can lead to abnormal mRNA transcript levels and results in the malfunctioning of cellular processes [1]. Processes crucial to carcinogenesis include those which help cells to increase in number or affect the cell differentiation and maturation [2].

High-throughput methods such as microarray technologies have significantly extended the possibilities of biological research due to their efficiency and quickness. Gene expression data from DNA microarrays have been widely analysed to quantify the changes in mRNA levels of genes, for example on tumor samples. To examine genomic aberrations, a technique called comparative genomic hybridization (CGH) [3] can be used. Array comparative genomic hybridization (aCGH) [4] allows the high-resolution mapping of DNA aberrations at tens of thousands of locations distributed throughout the genome. These technologies have been successfully used in studying the genetic profiles of different kinds of cancer types.

To get a better view to genetic events leading to formation of malignancies, it is advantageous to measure both gene expression and CGH data from the same tumor samples. In this way one can quantify the effects of genetic aberrations on the expression of genes. This kind of co-analysis has already been performed for many cancer types. Mostly, the analyses have been concentrated on identifying genes which are both over-expressed and have change in copy number [5]. In some studies, the correlations between copy number and gene expression have been studied [6]. Co-analysis has also been used to understand how the aberrations influence cancer pathophysiologies [7] and to identify relationships between DNA copy number, gene expression, and drug sensitivity [8].

Human gastrointestinal stromal tumors (GIST) and leiomyosarcomas (LMS) of the abdominal cavity and retroperitoneum are unusual malignancies that, until recently, were classified histologically together as LMS because of their similarities on light microscopy [9]. Advances in histopathology later provided ultrastructural evidence that GIST are distinct from muscle tumors [10]. In addition, the distinction of GIST and LMS based on microarray data has been reported [11].

In this work, we systematically co-analyze gene expression and aCGH data measured from GIST and LMS tumors. We study the similarity of the gene expression and array CGH measurement profiles over all measured genes. We identify highly expressed genes from gene expression data and genetic aberrations from array CGH data. These lists of highly expressed genes and genetic aberrations are then compared to uncover the degree of overlap between these two types of data. Gene Ontology enrichment analysis is used to investigate which biological processes are affected by genetic aberrations and high mRNA levels. Finally, we compare different cancer samples at the system level by computing correlation

between the chromosomally ordered expression and copy number data.

2. METHODS

2.1. Data

Primary tumors, 20 GIST and 20 LMS, were used to obtain DNA for aCGH experiments. Each tumor sample was hybridized against normal DNA. Agilent Human Genome CGH Microarrays (4x44k) were used in this study. Data was extracted from microarrays using Agilent feature extraction software version 9.5 with default settings. Finally, data were imported to Matlab and Lowess normalized to compensate for dye bias.

For gene expression experiments, mRNA from 68 primary tumor samples, 37 GIST and 31 LMS, was used. 37 of these 68 arrays were made using the genetic material from the same tumors that were used on aCGH arrays. Agilent human whole-genome microarray chips (44k) were used to measure the gene expression levels. Agilent feature extraction software version 8.0 was used to extract the data. Data were imported to Matlab and Lowess normalized to compensate for dye bias. Data was further quantile normalized to standardize the intensity distributions.

2.2. Determining highly expressed genes from gene expression data

Highly expressed genes were identified by finding the probes, whose intensity value is among the highest 5% of all intensity values. It should be noted as we do not have data measured under normal conditions, this was the best available approach to identify highly expressed genes. Normalization using e.g. computational median over all the samples would have reduced the amplitude of the genes that are systematically highly expressed in most of the samples. Finally, based on the expression of individual probes, the expression of each gene was determined by averaging over all the probes mapped to a given gene.

2.3. Identifying aberrations from aCGH data

In aCGH data, genetic aberrations usually span multiple probes. To find the aberrations, the data needs to be segmented and segments should then be identified as aberrated or normal. The normalized log ratio intensity values were segmented using circular binary segmentation (CBS) algorithm [12]. CGHcall algorithm [13] was used for making the aberration calls for each segment. A gene was considered aberrated if the segment, in which the gene resides, was classified as aberrated. Finally, genes were flagged as aberrated if probes at that region belong to an aberrant segment.

2.4. Identifying common abnormalities

We examined which genes behave abnormally throughout the sample set or in the majority of samples. We generated lists of those genes which were highly expressed in 60, 70, 80, 90 or 100 % of samples. Likewise, lists of genes which were aberrated in 60…100 % of samples, were generated. In addition, we generated a list of those genes which were both highly expressed and aberrated in 60…100 % of samples. The analyses were performed to GIST and LMS sample sets separately.

2.5. Examining the correlation

We examined global similarity of the genetic profiles between gene expression and aCGH data. Using only the 37 samples that were made using the genetic material from the same tumors we estimated correlation between each possible gene expression – aCGH sample pair. First, all the probes corresponding to a given gene were combined and only the genes that appear on both of the arrays were kept. Next the genes were ordered into the order they appear on the chromosomes. Then, the correlation was estimated using Pearson's correlation coefficient.

2.6. Enrichment analysis

Enrichment analysis was performed to the list of genes, which were both highly expressed and aberrated, because those genes are assumed to be related to the biological processes underlying the cancer types under study. We can utilize gene annotations to gene ontology (GO) terms, by studying which terms get more annotations than would be expected by random [14]. Given a list of interesting genes L, we can go through all GO terms and for each term i, count how many genes from list L are annotated to it, denoted by k_i. To obtain a null distribution, the same process can be repeated using a list of all possible genes, that is, all the genes on the microarray. As a result, for each GO term the number of annotations n_i is obtained. Let $N = \sum n_i$ and $K = \sum k_i$, then we can use the cumulative distribution function F of hypergeometric distribution to obtain a p-value for observing at least k_i annotations to term i as

$$p_i = 1 - F(k_i - 1; N; K; n_i) = 1 - \sum_{j=0}^{k_i - 1} \frac{\binom{K}{j}\binom{N-K}{n_i - j}}{\binom{N}{n_i}}$$

Terms with smallest p-value are considered to be the most interesting ones.

3. RESULTS

In the first part of our results we show, how the number of genes that are highly expressed, aberrated or both, change as the function of the number of samples. As our method for identifying highly expressed genes from the gene expression data does not remove naturally highly expressed genes, it is expected to have a high number of

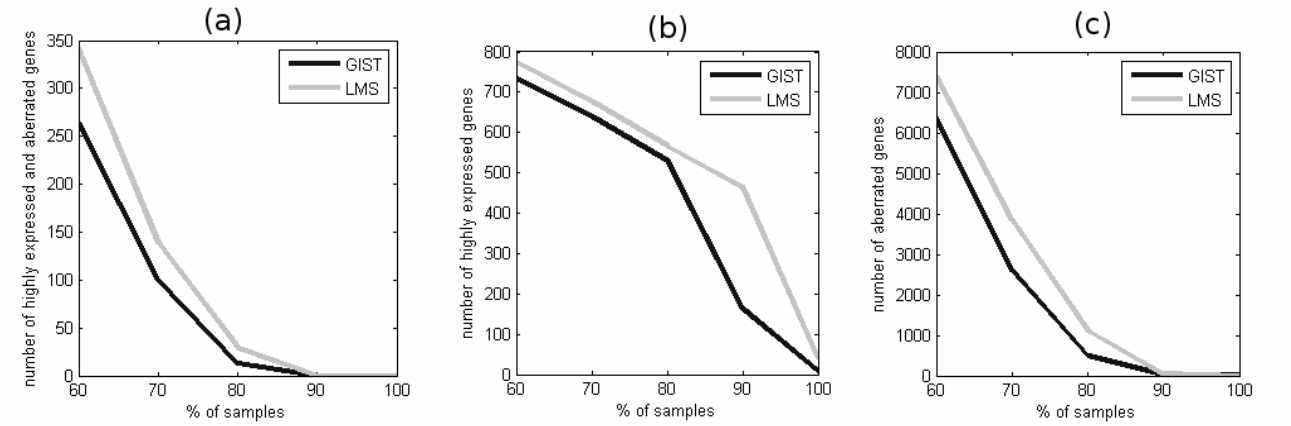


Figure 1: Number of genes identified as highly expressed in gene expression data (b), aberrated in aCGH data (c) and both highly expressed in gene expression data and aberrated in aCGH data (a) are shown as a function of the percentage of samples that share these common genes.

false positive on the list. Indeed, the number of genes increases very rapidly in the case of gene expression data (Figure 1(b)). Still Figure 1(b) shows only very small number of genes that are expressed on all the samples. This indicates that both GIST and LMS samples are highly heterogeneous.

Joint analysis of gene expression and aCGH data (Figure 1(a)) shows that only at the level of 80% of the samples, we start to observe a significant number of genes that are identified from the both sources of data. It should be noted that by combining the lists from gene expression and aCGH, we are able to remove naturally highly expressed genes that appear in gene expression data gene list. Thus, these genes should not affect our results or conclusions about relationships between gene expression and genetic aberrations.

Gene ontology enrichment analysis was applied to list of genes that was identified from both gene expression and aCGH data. Analysis with LMS uncovered several statistically significant GO categories (p<0.01) that are known to be related to cancer. These include histone modification and acetylation as well as regulation of apoptosis. This indicates that our analysis successfully captured genes that are related to this cancer type. For GIST the enrichment analysis uncovered categories that are related to different cellular complex disassembly and translation regulation. While these categories can also be linked to cancer, they are more general than categories from LMS analysis.

Figure 2(a) represents the correlations between aCGH and gene expression samples. Each column corresponds to an aCGH sample, and each row a gene expression sample. Samples number 1-20 are from LMS and samples number 21-37 are from GIST. The cells of the diagonal represent the correlations between aCGH and gene expression data of the same sample. It can clearly be seen, that the correlation between the aCGH and gene expression measurements of the same sample is higher on average than between any other samples. Furthermore, there is no difference in profiles between GIST and LMS. This indicates that the heterogeneity within GIST or LMS is more dominant than the differences between these two cancer types. This indicates that there is no clear global pattern in gene expression and aCGH profiles that would trivially separate these two classes of cancer. As for correlations computed using only aCGH (Figure 2(c)) and gene expression data (Figure 2(d)), the GIST and LMS classes are clearly observable.

4. CONCLUSIONS

Here, we have studied the relationships between gene expression and array CGH data, measured from a set of GIST and LMS tumors. Our results show that these cancer types show highly heterogeneous patters in their expression profiles. Comparison of lists of highly expressed and aberrated genes showed that significant number of common genes can be identified only in 80% of samples. Thus, the tumors genetic profiles are very different from each other even within a cancer type. Gene ontology enrichment analysis identified categories that are cancer related, indicating that genes that are identified in majority of the samples are in fact cancer related. Correlation analysis showed that the expression profiles measured from the same tumor using aCGH and gene expression arrays show significant correlation. However, significant correlation between these two data types within cancer type can not be observed.

In the light of this obvious heterogeneity of these cancer types, it is remarkable to observe that a very simple two gene classifier that accurately predicts the cancer type has been reported [15]. This observation outlines that fact that while the differences between cancer types are not evident, there are some underlying biological processes that give rise to these types of cancer. Based on this study, it is clear that in the case of highly heterogeneous cancer these processes can not be trivially uncovered just by naively analyzing high-throughput data. To understand cancer, significant amount of biological insight needs to be utilized in the analysis of data.

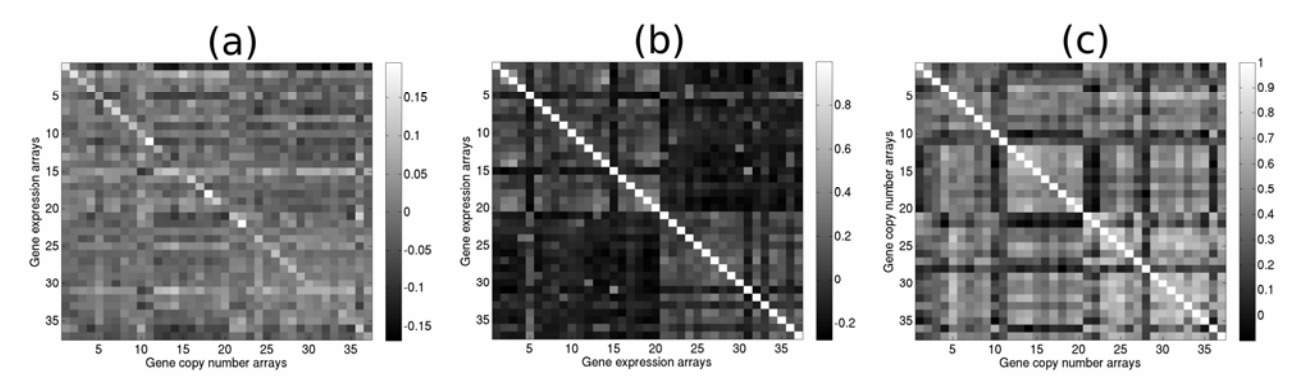


Figure 2: Correlation between the aCGH and gene expression profiles measured from the same tumors is shown (a). Correlations between gene expression and aCGH samples are shown in (b) and (c), respectively.

5. ACKNOWLEDGEMENTS

This work was supported by the Academy of Finland (project No. 122973 and 213462), National Technology Agency of Finland, and US National Institute of Health (Zhang RO1 CA098570).

6. REFERENCES

[1] W.R. Lai, M.D. Johnson, R. Kucherlapati, and P.J. Park, "Comparative analysis of algorithms for identifying amplifications and deletions in arrayCGH data," *Bioinformatics*, vol. 21, no. 19, pp. 3763-3770, 2005.

[2] D. Hanahan and R.A. Weinberg, "The hallmarks of cancer," *Cell*, vol. 100, no. 1, pp. 57-70, 2000.

[3] A. Kallioniemi, O.-P. Kallioniemi, D. Sudar, D. Rutovitz, J. Gray, F. Waldman, and D. Pinkel, "Comparative genomic hybridization for molecular cytogenetic analysis of solid tumors," *Science*, vol. 258, no. 5083, pp. 818-821, 1992.

[4] D. Pinkel, R. Segraves, D. Sudar, S. Clark, I. Poole, D. Kowbel, C. Collins, W.L. Kuo, C. Chen, Y. Zhai, S.H. Dairkee, B.M. Ljung, J.W. Gray, and D.G. Albertson, "High resolution analysis of dna copy number variation using comparative genomic hybridization to microarrays," *Nature genetics*, vol. 20, no. 2, pp. 207-211, 1998.

[5] S.C. Linn, R.B. West, J.R. Pollack, S. Zhu, T. Hernandez-Boussard, T.O. Nielsen, B.P. Rubin, R. Patel, J.R. Goldblum, D. Siegmund, D. Botstein, P.O. Brown, C.B. Gilks, and M. van de Rijn, "Gene expression patterns and gene copy number changes in dermatofibrosarcoma protuberans," *Am J Pathol.*, vol. 163, no. 6, pp. 2383-2395, 2003.

[6] D. Tsafrir, M. Bacolod, Z. Selvanayagam, I. Tsafrir, J. Shia, Z. Zeng, H. Liu, C. Krier, R.F. Stengel, F. Barany, W.L. Gerald, P.B. Paty, E. Domany, and D.A. Notterman, "Relationship of gene expression and chromosomal abnormalities in colorectal cancer," *Cancer Res.*, vol. 66, no. 4, pp. 2129-2137, 2006.

[7] K. Chin, S. DeVries, J. Fridlyand, P.T. Spellman, R. Roydasgupta, W.-L. Kuo, A. Lapuk, R.M. Neve, Z. Qian, T. Ryder, F. Chen, H. Feiler, T. Tokuyasu, C. Kingsley, S. Dairkee, Z. Meng, K. Chew, D. Pinkel, A. Jain, B.M. Ljung, L. Esserman, D.G. Albertson, F.M. Waldman, and J.W. Gray, "Genomic and transcriptional aberrations linked to breast cancer pathophysiologies," *Cancer Cell*, vol. 10, pp. 529-541, 2007.

[8] K.J. Bussey, K. Chin, S. Lababidi, M. Reimers, W.C. Reinhold, W.-L. Kuo, F. Gwadry, Ajay, H. Kouros-Mehr, J. Fridlyand, A. Jain, C. Collins, S. Nishizuka, G. Tonon, A. Roschke, K. Gehlhaus, I. Kirsch, D.A. Scudiero, J.W. Gray, and J.N. Weinstein, "Integrating data on DNA copy number with gene expression levels and drug sensitivities in the NCI-60 cell line panel," *Mol Cancer Ther*, vol. 5, no. 4, pp. 853-867, 2006.

[9] B.M. Clary, R.P. DeMatteo, J.J. Lewis, D. Leung, and M.F. Brennan, "Gastrointestinal stromal tumors and leiomyosarcoma of the abdomen and retroperitoneum: A clinical comparison," *Annals of Surgical Oncology*, vol. 8, pp. 290-299, 2001.

[10] J.C. Trent, J. Dupart, and W. Zhang, "Imatinib mesylate: Targeted therapy of gastrointestinal stromal tumor," *Current Cancer Therapy Reviews*, vol. 1, no. 1, pp. 93-108, 2005.

[11] M. Nykter, K.K. Hunt, R.E. Pollock, A.K. El-Naggar, E. Taylor, I. Shmulevich, O. Yli-Harja, W. Zhang, "Unsupervised analysis uncovers changes in histopathologic diagnosis in supervised genomic studies," *Technology in Cancer Research & Treatment*, vol. 5, no. 2, pp. 177-182, 2006.

[12] A.B. Olshen, E.S. Venkatraman, R. Lucito, and M. Wigler, "Circular binary segmentation for the analysis of array-based DNA copy number data," *Biostatistics*, vol. 5, no. 4, pp. 557-572, 2004.

[13] M.A. van de Wiel, K.I. Kim, S.J. Vosse, W.N. van Wieringen, S.M. Wilting, and B. Ylstra, "CGHcall: calling aberrations for array CGH tumor profiles," *Bioinformatics*, vol. 23, no. 7, pp. 892-894, 2007.

[14] M. Ashburner, C.A. Ball, J.A. Blake, D. Botstein, H. Butler, J.M. Cherry, A.P. Davis, K. Dolinski, S.S. Dwight, J.T. Eppig, M.A. Harris, D.P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J.C. Matese, J.E. Richardson, M. Ringwald, G.M. Rubin, and G. Sherlock, "Gene ontology: Tool for the unification of biology," *Nat. Genet.*, vol. 25, no. 1, pp. 25-29, 2000.

[15] N.D. Price, J. Trent, A.K. El-Naggar, D. Cogdell, E. Taylor, K.K. Hunt, R.E. Pollock, L. Hood, I. Shmulevich, and W. Zhang, "Highly accurate two-gene classifier for differentiating gastrointestinal stromal tumors and leiomyosarcomas," *Proc. Natl. Acad. Sci. USA*, vol. 104, no. 9, pp. 3414-3419, 2007.

COMPRESSION BASED CLASSIFICATION OF PRIMATE ENDOGENOUS RETROVIRUS SEQUENCES

Vladimir Kuryshev[1,2] *and Pavol Hanus*[2]

[1]Department of Behavioural Ecology and Evolutionary Genetics, Max Planck Inst. for Ornithology
[2]Institute for Communications Engineering, Technische Universität München

vkuryshev@tum.de

ABSTRACT

The highly divergent character of retrovirus sequences makes cross family alignment based classification of whole genomes difficult and unreliable. Standard methods thus focus on alignment based classification using only specific elements such as *pol* and *env* retroviral genes. In this paper a topology tree of exo- and primate endogenous retrovirus sequences based on whole genomes is presented. In order to avoid the necessity of making an alignment, compression was used to approximate a mutual information based distance between sequences.

1. INTRODUCTION

Endogenous retroviruses (ERVs) are remnants of ancient retroviral infections. Retroviruses in general are viruses capable of inserting their genome into the DNA of hosts. They become endogenous once they have been inserted into the germ-line. ERVs possess a similar genomic organization to present day exogenous retroviruses (XRVs) such as the human immunodeficiency virus (HIV). They are composed of *gag*, *pol*, and *env* coding regions placed between two long terminal repeats (LTRs). The LTRs possess nucleotide sequence motifs that are fundamental for the regulation of retroviral gene expression. The *gag* and *env* genes encode retroviral capsid and envelope proteins, respectively, whereas the *pol* gene encodes enzymes for viral replication, integration, and protein cleavage. About 8% of the human genome are ERVs. Although most of the proviruses (integrated copies of the virus genome) have undergone extensive deletions and mutations, some have retained the potential to produce viral products, even virus-like particles (reviewed in [1]).

In the current taxonomy, retroviruses are classified into seven genera: alpha-, beta-, gamma-, delta-, epsilon-, lenti-, and spuma-retroviruses [2]. However the diverse endogenous members remain relatively poorly incorporated into the classification scheme. Typically, ERVs are being classified using alignment of short conserved protein motifs of *pol* or *env* genes [3]. Obviously, by restricting the comparison to one gene a lot of information is being neglected. Usually the ERV phylogenetic trees are constructed using also representatives from XRVs.

In this work, we attempted to classify full-size genomes of both exo- and primate-specific endoretrovirus sequences. Compression was used to approximate a mutual information based distance between sequences in order to avoid the necessity of making an alignment. Shannon's Mutual Information quantifies the amount of information shared between stochastic processes. It is thus well suited to derive a distance measure quantifying their dissimilarity [4]. Genomic sequences can be regarded as realizations of such stochastic processes and compression can be used to approximate the distance measure from the genomic sequences. The use of compression for phylogenetic classification was first introduced in Li et al. [5]. The compression based distance does not require an alignment, it is capable of catching more subtle statistic similarities than simple sequence divergence and is largely independent of the lengths of the compared sequences [6].

2. METHODS

2.1. Mutual Information Distance

Information theory describes the relatedness of stochastic processes S_i and S_j as the mutual information $I(S_i;S_j)$ shared by these processes

$$I(S_i;S_j) = H(S_i) - H(S_i|S_j) = I(S_j;S_i), \quad (1)$$

where H is the entropy. Mutual information is an absolute measure of information common to both sources. It can be transformed to a bounded distance through normalization by the maximum entropy of both processes resulting in the following distance metric

$$d_{CL}(S_i,S_j) = 1 - \frac{I(S_i;S_j)}{\max(H(S_i),H(S_j))} \leq 1. \quad (2)$$

In order to achieve $d_{CL} = 0$ the two sources must not only share maximum possible mutual information, but need to have identical entropies as well. This distance has also been successfully applied to the clustering of SNPs in gene mapping [7]. Using conditional entropy the distance can be reformulated to

$$d_{CL}(S_i,S_j) = \frac{\max(H(S_i|S_j),H(S_j|S_i))}{\max(H(S_i),H(S_j))}. \quad (3)$$

2.2. Compression Based Entropy Approximation

The compression ratio achieved by an optimal compression algorithm designed for a given stochastic process S when compressing a message s generated by this process $s \mapsto |\mathrm{comp}(s)|$ is a good approximation of its actual entropy rate

$$H(S) \approx \frac{|\mathrm{comp}(s)|}{|s|}, \tag{4}$$

where $|.|$ denotes the size in bits or symbols. The compressors used in the scope of this work are so-called universal compression algorithms. They are universal in the sense that they gradually learn the statistics of the sequence while compressing. Therefore, we can approximate the conditional entropy $H(S_i|S_j)$ as the compression ratio achieved for message s_i when the compressor has been trained on the message s_j. This is achieved by compressing the concatenation $|\mathrm{comp}(s_j, s_i)|$ of the sequence s_j and s_i. Thus,

$$H(S_i|S_j) \approx \frac{|\mathrm{comp}(s_j, s_i)| - |\mathrm{comp}(s_j)|}{|s_i|}, \tag{5}$$

and for $|\mathrm{comp}(s_i)| > |\mathrm{comp}(s_j)|$ we obtain:

$$d_{CL} = \frac{|\mathrm{comp}(s_j, s_i)| - |\mathrm{comp}(s_j)|}{|\mathrm{comp}(s_i)|}. \tag{6}$$

This resembles the similarity metric based on Kolmogorov complexity proposed in [8]. Suitability of different compression algorithms for the purpose of classification is discussed in detail in [4]. The prediction by partial matching (PPM) compressor [9] was used in this work due to efficient implementation and good classification performance.

2.3. Classification and Results Analysis

After computing the distances between all sequences a phylogenetic classification is to be performed. The average neighbor joining method was used for this purpose. The results were visualized using MEGA4 [10]. For better comparison, the same clustering method was used to classify the test set sequences using alignment based distances computed by ClustalW [11]. The web-based tools *blast2seq* [12] and *blat* [13] were used to verify our findings.

3. DATASET

A test set of amino acid (aa) sequences corresponding to the *pol* region of 62 exo- and endogenous retroviruses was retrieved from the paper Jern *et al*. [14] and used for the comparison of the methods. For the compression based phylogeny presented in Figure 1 we used the currently available full set of 54 complete exogenous retrovirus genomes (*Retroviridae* from the NCBI Refseq collection http://www.ncbi.nlm.nih.gov/). The sequence of the Drosophila melanogaster gypsy virus (AF033821) was used as an outgroup to root the tree. The primate-specific ERV consensus sequences (84) corresponding to internal retrovirus regions (without LTRs) were fetched from the Repbase (http://www.girinst.org/). The XRV sequences vary in size from 2.6 to 14.0 kb, the ERVs from 1.8 to 11.2 kb.

4. RESULTS

4.1. Alignment vs. Compression Based Clustering

In order to compare the performance of alignment and compression based distance measures for retrovirus classification, we built topology trees of aa-sequences of the *pol* gene described and classified by Jern *et al*. [14] using the ClustalW based alignment distance and the compression distance computed using the PPM compressor. Visual inspection of the resulting trees (data not shown) suggested that, in general, the sequence clustering obtained by both methods is similar with slight differences in the branch distributions within major clades. In addition, the obtained clustering was in both cases in agreement with the observations of Jern *et al*. However, the aa-sequence corresponding to the Mason-Pfizer monkey virus (MPMV) was misleadingly first classified as an outlier by PPM. Further inspection has revealed that the used sequence downloaded from Jern *et al*. contained in addition to the *pol* protein sequence a frame shifted protein snippet of about the same length. Since PPM as opposed to the ClustalW based approach used also this portion of the sequence to compute the distance, it correctly considered the sequence to be distant from all the others.

In conclusion, it can be stated that the results of the alignment and the compression based classification largely agree. Moreover, the reliability of the compression based classification can be improved if nucleotide instead of aa-sequences are being used. This is due to the increased sequence length and also due to the reduced alphabet size giving the universal compressor a better chance to learn the statistics of shorter sequences.

4.2. Compression Based Primate ERV Classification

Figure 1 depicts the whole genome based tree of both exo- and primate endogenous retroviruses obtained using compression. The observed clustering of XRVs corresponds to the established taxonomy (http://www.ncbi.nlm.nih.gov). The only unexpected finding is the assignment of the squirrel monkey retrovirus (SMRV) to the delta genera viruses. The SMRV is believed to be closely related to the beta like mouse mammary tumor virus (MMTV). In order to investigate this disagreement we have increased the classification weight of MMTV by incorporating additional MMTV-like sequences. As a consequence SMRV was assigned to the MMTV-like family. This suggests that there is a close relation between SMRV and both retrovirus genera delta and beta. In addition, it indicates that care needs to be taken when interpreting classification results of cross-family related sequences.

The salmon swim bladder sarcoma virus (SSSV) is annotated in Refseq as unclassified. First described by Paul *et al.* [15] it represents the only fish-specific XRV in our dataset. The author's findings based on the reverse transcriptase suggest to place SSSV between the gamma

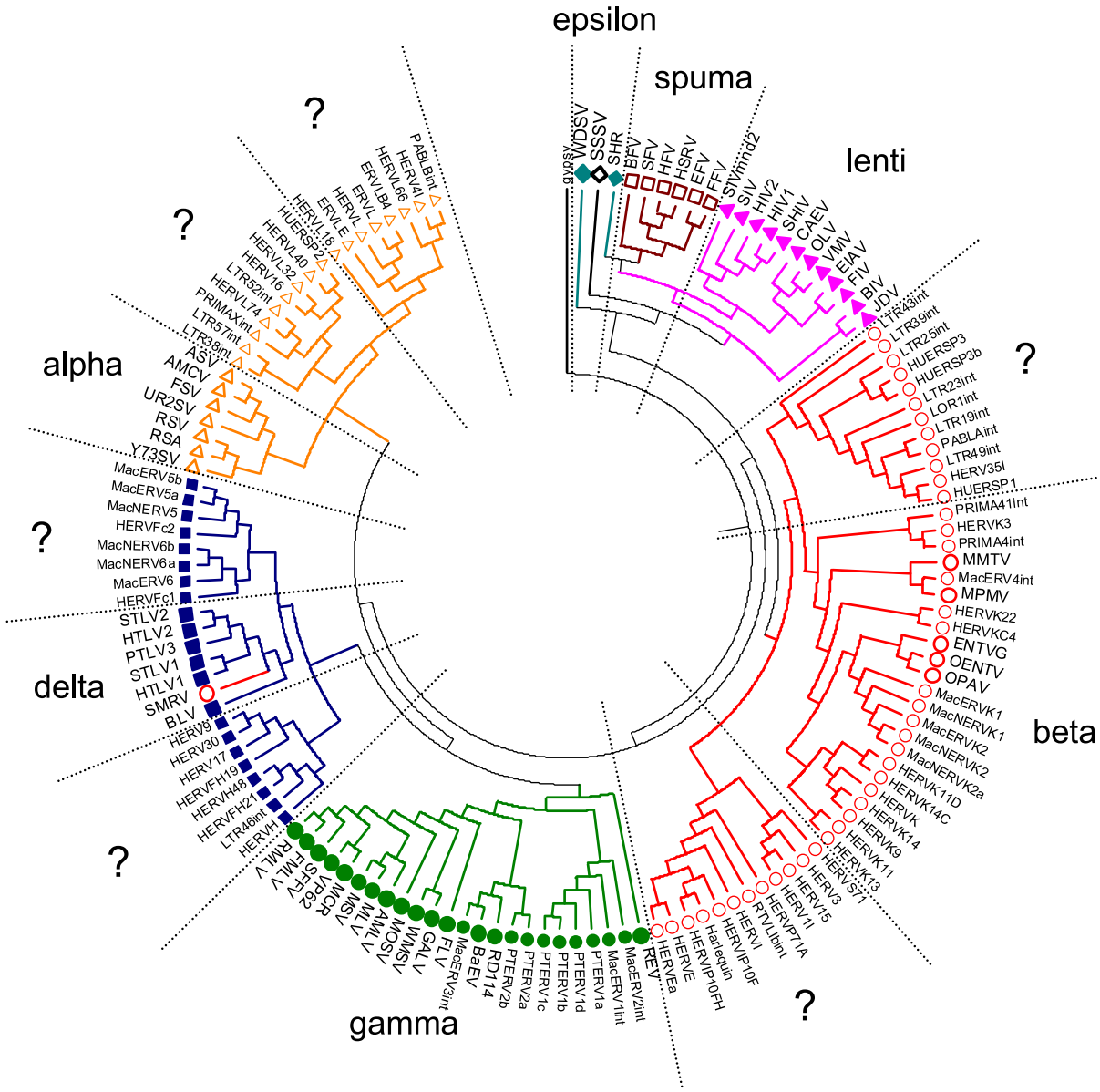


Figure 1. **Topology tree of ERVs and XRVs based on compression similarity.** Names of the leaves correspond to standard exogenous and endogenous retrovirus abbreviations according to the International Committee on Taxonomy of Viruses (ICTV) and the Repbase nomenclatures, respectively (dashes and underscores were omitted). XRV names are large in size. Presumably distinct ERV clades are denoted by a "?". The tree is rooted using the gypsy retroelement (indicated on the top of the tree). Different symbols were used for sequences related to different clades: △ - alpha-like, ■ - delta-like, ● - gamma-like, ○ - beta-like, ▲ - lenti, □ - spuma, ♦ - epsilon, ◇ - unclassified.

and epsilon genera but in a distinct branch, what is consistent with our results.

Based on the obtained XRV clustering, the assignment of ERVs found on the corresponding subtrees of the XRV genera was attempted. It could be observed that only the XRVs from the beta and gamma genera have closely related primate ERV counterparts. The remaining primate ERVs seem to cluster in distinct genera with distant relationship to the alpha, delta and beta XRV clades (they are depicted by a "?"). The epsilon, spuma and lenti family do not seem to have any primate ERV relatives. The clustering of ERVs into clades distant from the established XRV genera was also suggested by Han *et al.* [16].

According to Baillie *et al.* [17] the Mason-Pfizer monkey virus (MPMV) exists only in exogenous form. However, the topology tree shows the primate-specific endogenous consensus MacERV4int at close proximity to MPMV. An alignment of both sequences revealed that they are highly homologous (76% identity). A blat scan of the *Macaca mulatta* genome (rheMac2) for the MacERV4int consensus returned dozens of highly similar hits. All this implies that MacERV4int consensus represents a group of endogenous retroviruses in the macaque genome that is closely related to the exogenous MPMV form, also con-

firmed by findings of Han *et al.* [16].

5. DISCUSSION

Since the alignment based classification is limited to alignable parts and the alignment of highly heterogenous retrovirus genomes is unreliable the scope of this work was to test whether compression based classification can be applied to the relatively short but complete retrovirus genomes. After verifying the suitability of the approach on exogenous retroviruses, classification of recently published primate endogenous retrovirus sequences was attempted. The resulting tree indicates that most primate exogenous retroviruses represent own genera and that only the beta and gamma exogenous retrovirus genera have close primate endogenous relatives. Most endogenous viruses seem to cluster in distinct separate clades and are likely remnants from infections by ancient extinct retroviruses.

6. CONCLUSION

Using compression based approximation of a mutual information based distance we were able to classify sequences of complete exogenous retrovirus genomes with good agreement to published data. Moreover, using this method we proposed a classification (topology tree) for a collection of primate-specific ERVs. The biological meaning of our observations needs to be further investigated and the robustness of the compression based approach remains to be thoroughly tested.

7. ACKNOWLEDGMENTS

This work was supported by the DFG research grant MU 1479/1-2.

8. REFERENCES

[1] N. Bannert and R. Kurth, "The Evolutionary Dynamics of Human Endogenous Retroviral Families," *Annu Rev Genomics Hum Genet*, vol. 7, pp. 149–173, 2006.

[2] M. Van Regenmortel et al., *Virus Taxonomy: Classification and Nomenclature of Viruses: Seventh Report of the International Committee on Taxonomy of Viruses*, Academic Press, 2000.

[3] L. Benit, P. Dessen, and T. Heidmann, "Identification, Phylogeny, and Evolution of Retroviral Elements Based on Their Envelope Genes," *Journal of Virology*, vol. 75, no. 23, pp. 11709, 2001.

[4] Z. Dawy, J. Hagenauer, P. Hanus, and J. C. Mueller, "Mutual information based distance measures for classification and content recognition with applications to genetics," in *Proc. of the ICC 2005*, 2005.

[5] M. Li, J. H. Badger, X. Chen, S. Kwong, P. Kearney, and H. Zhang, "An information-based sequence distance and its application to whole mitochondrial genome phylogeny ," *Bioinformatics*, vol. 17, no. 2, pp. 149–154, 2001.

[6] R. Cilibrasi and P. Vitani, "Clustering by Compression," *Information Theory, IEEE Transactions on*, vol. 51, no. 4, pp. 1523–1545, 2005.

[7] P. Hanus, B. Goebel, J. Dingel, J. Weindl, J. Zech, Z. Dawy, J. Hagenauer, and J. Mueller, "Information and communication theory in molecular biology," *Electrical Engineering (Archiv fur Elektrotechnik)*, pp. 161–173, 2007.

[8] M. Li, X. Chen, X. Li, B. Ma, and P. Vitanyi, "The similarity metric," in *Proc. of the 14th annual ACM-SIAM symposium on Discrete algorithms (SODA)*, Baltimore, Maryland, 2003, pp. 863–872.

[9] J. G. Cleary and I. H. Witten, "Data compression using adaptive coding and partial string matching," *IEEE Transactions on Communications*, vol. COM-32, no. 4, pp. 396–402, April 1984.

[10] K. Tamura, J. Dudley, M. Nei, and S. Kumar, "MEGA4: Molecular Evolutionary Genetics Analysis (MEGA) Software Version 4.0," *Molecular Biology and Evolution*, vol. 24, no. 8, pp. 1596, 2007.

[11] J. Thompson, D. Higgins, and T. Gibson, "CLUSTAL W," *Nucleic Acids Res*, vol. 22, no. 4673, pp. 80, 1994.

[12] T. Tatusova and T. Madden, "BLAST 2 Sequences, a new tool for comparing protein and nucleotide sequences," *FEMS Microbiology Letters*, vol. 174, no. 2, pp. 247–250, 1999.

[13] W. Kent et al., "BLAT—The BLAST-Like Alignment Tool," *Genome Research*, vol. 12, no. 4, pp. 656–664, 2002.

[14] P. Jern, G. Sperber, and J. Blomberg, "Use of Endogenous Retroviral Sequences (ERVs) and structural markers for retroviral phylogenetic inference and taxonomy," *Retrovirology*, vol. 2, pp. 50, 2005.

[15] T. Paul, S. Quackenbush, C. Sutton, R. Casey, P. Bowser, and J. Casey, "Identification and Characterization of an Exogenous Retrovirus from Atlantic Salmon Swim Bladder Sarcomas," *Journal of Virology*, vol. 80, no. 6, pp. 2941–2948, 2006.

[16] K. Han, M. Konkel, J. Xing, H. Wang, J. Lee, T. Meyer, C. Huang, E. Sandifer, K. Hebert, E. Barnes, et al., "Mobile DNA in Old World Monkeys: A Glimpse Through the Rhesus Macaque Genome," *Science*, vol. 316, no. 5822, pp. 238, 2007.

[17] G. Baillie, L. Lagemaat, C. Baust, and D. Mager, "Multiple groups of endogenous betaretroviruses in mice, rats, and other mammals.," *Journal of Virology*, vol. 78, no. 11, pp. 5784–5798, 2004.

ACTIVE LEARNING OF BAYESIAN NETWORK STRUCTURE IN A REALISTIC SETTING

Antti Larjo[1], *Harri Lähdesmäki*[1], *Marc Facciotti*[2], *Nitin Baliga*[2], *Olli Yli-Harja*[1], *and Ilya Shmulevich*[2]

[1]Department of Signal Processing, Tampere University of Technology,
P.O. Box 553, FI-33101 Tampere, Finland
[2]Institute for Systems Biology,
1441 North 34th Street, Seattle, WA 98103, USA
antti.larjo@tut.fi, harri.lahdesmaki@tut.fi

ABSTRACT

Bayesian networks (BNs) are frequently used for modeling genetic regulatory networks. The structure of a static BN cannot in general be learnt unambiguously from observational data alone but interventions (i.e. knock-outs or over-expressions) are also required. These interventions can be difficult and costly to perform, thus calling for careful planning of experiments. Active learning methods can be used to suggest which interventions should be performed in order to increase our knowledge about the network structure maximally. Here, we utilize such a method for the first time in a realistic setting with measured wild-type and perturbed gene-expression and protein data and show the applicability and usability of the approach for designing biological experiments with maximal expected utility.

1. INTRODUCTION

Choosing which biological experiments to perform in order to benefit maximally from them is a highly non-trivial problem. The solutions to such a problem are context dependent: Trying to infer the dynamics of a system sets different demands on experimental design than when inferring the structure of a system, and will thus need to be addressed by different methods. Here we are interested in the problem of finding the structure of a biochemical subnetwork as efficiently as possible when the used model class is (causal) Bayesian networks. For demonstration, we consider learning both gene regulatory network and signaling network structures. We demonstrate the usability of a method to suggest maximally informative experiments.

2. METHODS

2.1. Bayesian networks

Given a set of random variables $\mathcal{X} = \{X_1, ..., X_n\}$, a Bayesian network is defined as a pair (G, θ), where G is a directed acyclic graph (DAG), which is a graphical representation of the conditional independencies between variables in $\mathcal{X}$, and θ is the set of parameters for the conditional probability distributions of these variables. The joint distribution over $\mathcal{X}$ factorizes according to G as

$$P\left(X_1, ..., X_n|G, \theta\right) = \prod_{i=1}^{n} P\left(X_i|Pa_G(X_i), \theta_i\right), \quad (1)$$

where $Pa_G(X_i)$ is the set of parents of node X_i in G, and θ_i the parameters for the distribution of X_i conditional on its parents.

In searching for the structure that most probably generated the data, of main interest is the posterior probability of a DAG given the data $P(G|D) = P(D|G)P(G)/P(D)$, where $P(G)$ is the prior probability of G, $P(D) = \sum_{G'} P\left(D|G'\right) P(G')$ is the prior probability of data (sum goes over all possible DAG structures), and

$$P\left(D|G\right) = \int_{\theta} P\left(D|G, \theta\right) P\left(\theta|G\right) d\theta. \quad (2)$$

In this paper we only consider BNs having all the variables observed, discrete-valued and have multinomial conditional probability distributions (CPDs). We use uniform Dirichlet parameter priors since Dirichlet distribution is the conjugate prior of multinomials and makes it possible to obtain the closed form solution for Equation (2), which now becomes [1]

$$P(D|G) = \prod_{i=1}^{n} \prod_{j=1}^{q_i} \frac{\Gamma(N'_{ij})}{\Gamma(N'_{ij} + N_{ij})} \prod_{k=1}^{r_i} \frac{\Gamma(N'_{ijk} + N_{ijk})}{\Gamma(N'_{ijk})}, \quad (3)$$

where N_{ijk} is the number of times the configuration $(X_i = k, Pa_G(X_i) = j)$ occurs in data D, N'_{ijk} are hyperparameters (a.k.a. pseudo-counts) of the Dirichlet distributions, $N_{ij} = \sum_k N_{ijk}$ and $N'_{ij} = \sum_k N'_{ijk}$, q_i is the number of different parent configurations, and r_i is the number of different states that node i can take.

Ideally, we would like to have the whole posterior distribution of DAGs and calculate our further analyses based on that (i.e. perform full Bayesian analysis). But since the number of different DAGs grows super-exponentially

with n, evaluating the score (Equation (3)) for all possible structures is prohibitive for all but smallest of n ($n \leq 6$ or so). Instead, one is forced to resort to taking a sample of the posterior distribution with MCMC, as is done in this study.

An assumption we have to make is that the data is sampled from a probability distribution which can be represented with a Bayesian network (so called faithfulness assumption). In many real cases, this assumption is not likely to hold, especially for data from genetic networks (like in this study), where feedback loops are present. Still, we are obliged to make this approximation in order to be able to use a rigorous modeling approach.

2.2. Equivalence classes and interventional data

Given only observational (i.e. no interventions) data, it is generally impossible to learn the structure of a BN unambiguously because there is more than one structure producing the same combined probability distribution. Such sets of inseparable DAGs constitute equivalence classes, each of which consists of all the DAGs having the same v-structures[1] and otherwise the same structure when edge directions are ignored [2].

With interventions (i.e. forcing or "clamping" a node or set of nodes to a certain value) we can break these classes, by inducing bias towards some of the alternatively possible structure(s). Forcing the value of a node determines the directions of edges adjacent to it and thus splits the equivalence classes into transition sequence (TS) equivalent structures [3]. With enough interventions, the size of the most probable TS equivalent class should reduce to one.

In gene networks, interventions can be either over-expressions, meaning that a gene is set to state "on", or knock-outs, corresponding to setting the gene "off". Since these interventions are based on biological mechanisms that are inherently stochastic, there is uncertainty in how well the intervention succeeds. However, here we take the interventions to be ideal.

2.3. Active learning

Active learning methods are designed to suggest which interventions should be made in order to maximally benefit from their effect of breaking equivalence classes or, more generally, to learn the structure of a BN with minimal cost of experiments.

Basically, two different approaches to selecting the perturbations have been presented: (i) those that break equivalence classes [4] and (ii) decision theoretic that aim to diminish our uncertainty (or increase information maximally) about some edges [5, 6]. These approaches are in fact closely related and complementary, since within an equivalence class the inability to say which direction an edge takes is, in other words, uncertainty about that edge.

We use the method presented by Murphy [5], where the expected utility of making an intervention a (which can be a plain observation, i.e. "empty" intervention, or consist of setting the value of one or more nodes at a time) is defined as

$$V(a) = \sum_{G \in \mathcal{G}} \sum_{y \in \mathcal{Y}_{G,a}} P(y|G,a,D)P(G|D)U(G,a,y,D), \tag{4}$$

where $\mathcal{G}$ is our set of possible DAGs, $\mathcal{Y}_{G,a}$ denotes the set of possible observations that G can produce given that intervention a has been made. For the utility function $U(G,a,y,D)$ we use (assuming equivalent cost for each intervention) $\log P\left(G|a,y,D\right)$.

The best action is chosen from the set of possible actions $\mathcal{A}$ as the one with maximal utility $a^* = \arg\max_{a \in \mathcal{A}} V(a)$. The optimal way of finding this action is by exhaustive enumeration.

Since the number of DAGs grows super-exponentially with the number of nodes, the exhaustive approach is practically unusable when $n > 6$. Therefore, stochastic sampling is used to obtain a sample from the posterior $P(G|D)$ which is then used in the above calculations.

Also, since the number of different observations a BN can produce is $\prod_{i=1}^{n} r_i$ (r_i is the number of discretization levels for node i), it quickly becomes too expensive to evaluate the above algorithm for all of them. Thus, we must again resort to sampling to keep the computing times reasonable. Sampling is done in this study in the same way as discussed in [5], by using importance sampling and drawing observations from a uniform distribution. The number of possible actions is rather small in our case so sampling is not needed for them.

3. RESULTS

3.1. Data

Our first dataset, which we refer to as the Halo dataset, consists of 242 gene expression measurements of 7 different transcription factors in *Halobacterium salinarum* [7, 8]. These transcription factors form the core of the transcriptional network in *H. salinarum* and are also believed to largely control the expression of each other, thus forming a small regulatory subnetwork. The dataset contains interventions (over-expressions) for all the 7 genes as well as normal observations (i.e. expression measurements without over-expressions). Therefore, this is an ideal dataset for our purposes.

The data was discretized into ternary values using a likelihood ratio statistic based model for detecting under- and over-expressed genes (with significance level 0.15) [9]. Some interventional measurements (8 in total) were removed due to having wrong discretization levels, implying most probably unsuccessful interventions.

The second dataset, which we call the Sachs dataset, consists of flow cytometry measurements from a signaling network with 11 nodes, of which 5 have been perturbed in some measurements [10]. These interventions contain both inhibitions and activations of the nodes, which should intuitively give the active learning a greater advantage over non-active learning than with the Halo dataset. The data

[1] a v-structure is a triplet (a,b,c) where $a \rightarrow b \leftarrow c$ and $a \nsim c$ (i.e. a and c are not joined).

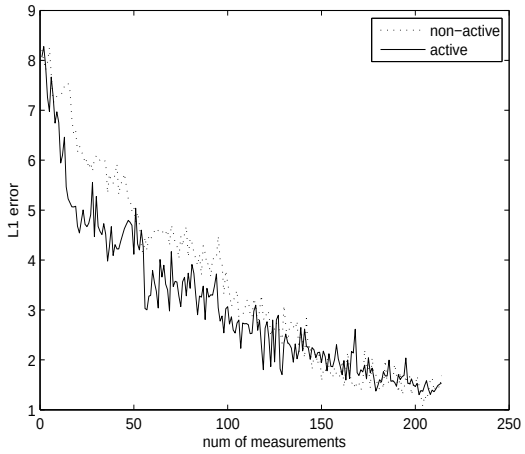


Figure 1. Using the Halo dataset, L_1 error was calculated for active and non-active learning methods by comparing to the structure derived from the "true" posterior by taking edges with posterior probability > 0.5. Number of measurements are in addition to the initial 20 observations. Initial burn-in was $4 \cdot 10^5$, between-measurement burn-in was $2 \cdot 10^4$, graph sample size 10^4, and sampled observations 100. Results averaged from five different runs.

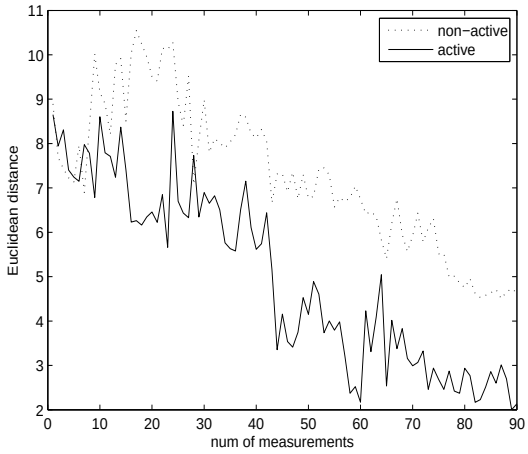


Figure 2. Euclidean distance between edge posterior probabilities calculated from the "true" posterior distribution and either active or non-active learning methods when using the Sachs dataset. Number of measurements are in addition to the initial 40 observations. Initial burn-in was $2 \cdot 10^5$, between-measurement burn-in was 5000, graph sample size 5000, and sampled observations 300. Results averaged from four different runs.

was discretized into ternary values in the same way as in [10]. From the whole dataset we took a sample with 100 observational data points and 20 data points per intervention, totaling 220 measurements.

3.2. Active learning and random interventions

Instead of using the particle filter based updating done in [5], we used normal MCMC since it can be argued that it is what one would preferably use when there is plenty of computational time available between consequent measurements, as in, e.g., performing studies involving microarray measurements.

To compare the performances of the active and non-active learning methods, so called L_1 edge error was used [5, 6]

$$\begin{aligned} L_1(P_t) = \sum_{i=1}^{n} \sum_{j=i+1}^{n} & I_{G^*}(X_i \to X_j)(1 - P_t(X_i \to X_j)) \\ & + I_{G^*}(X_i \leftarrow X_j)(1 - P_t(X_i \leftarrow X_j)) \\ & + I_{G^*}(X_i \nsim X_j)(1 - P_t(X_i \nsim X_j)), \end{aligned} \quad (5)$$

where $P_t(\cdot) = P(\cdot|D_{1:t})$ is the posterior marginal probability of an edge given data points up to index t, and $I_{G^*}(c)$ is the indicatior function which takes value 1 if c is present in the true structure G^* and 0 otherwise. We also used the normal Euclidean distance between edge posterior probabilities as a measure of convergence towards the "true" posterior distribution.

Each trial was initiated by taking a set of observations as initial data and, using this data, by running two MCMC chains in parallel for a long initial burn-in period. After this, samples were taken from both chains and the convergence of the chains was checked by comparing distributions of edge posterior probabilities calculated from both samples. When the distributions were similar, either sample was used as the initial sample for both active and non-active learners.

The active learning method proceeds by making at each step the measurement (intervention or observation) suggested by the active learning algorithm based on the sampled graphs, available measurements, sampled observations, and data collected so far. After each new measurement the chain is run for a between-measurement burn-in period and a new sample of graphs is taken. The non-active learning proceeds in the exact same way but instead of using an algorithm to suggest the next measurement, it just makes one randomly without replacement (i.e. takes one of the available measurements from the dataset).

To approximate the true posterior distributions, normal batch-style MCMC chains were run for the whole datasets and using very long burn-ins ($8 \cdot 10^5$ for the Sachs dataset and $2 \cdot 10^6$ for the Halo dataset) and big sample sizes ($2 \cdot 10^5$ for the Sachs dataset and $5 \cdot 10^5$ for the Halo dataset).

Figure 1 shows the results when using the Halo dataset. L_1 edge errors for both non-active and active learning methods were calculated by using as the reference structure the graph obtained by including only edges with posterior probability over 0.5 in the "true" posterior distribution. Parameter values (sample sizes etc.) used are shown in the caption. Figure 2 shows the same using the Sachs dataset, except now Euclidean distances to the edge posterior probabilities of the "true" posterior distribution were

calculated for non-active and active learners. In this experiment we also took two similar measurements simultaneously instead of just one.

4. DISCUSSION

As can be deduced from Figures 1 and 2, the convergence towards the final results is faster with active learning than with non-active learning. Thus, using an active learning method to guide experimentation can result in savings in time and costs.

The performance of active learning methods has usually been assessed with simulated data. As shown here, the methods do not perform as convincingly with real data, due to possibly existing factors outside the targeted subsystem and the real systems containing cyclic regulatory relationships. Thus, it would be one step closer to reality if, e.g., simulated data from systems with hidden variables were used when comparing the methods.

Looking at the sequence of actions suggested by the active learning algorithm tells us what is probably intuitively clear: The most beneficial way is to mostly make interventional measurements rather than obtaining a lot of observational data. In the beginning of the investigations, however, it pays off to acquire (usually less costly) observations in order to get a solid basis for deciding which interventions to make. Even though part of the better performance of active learning over non-active can be explained by the fact that active learning suggests mostly interventions in the beginning while non-active learning samples uniformly from the set of interventional and observational measurements, the active learning should still (in the long run) overperform non-active due to choosing the order in which to make the interventions. This was also validated using simulated data (results not shown).

In order to be able to tell how many experiments to perform and when making more experiments produces no more benefit, a stopping criterion should be developed. A simple heuristic could be checking for the changes in posterior distribution between measurements and if there is no trend or bigger jumps in change, then it can be concluded at that point that more measurements tell us nothing new.

An alternative method of active learning by Pournara [4] approaches the problem by considering how to split the equivalence classes most efficiently. Although this is much faster than Murphy's method [5], the latter can perhaps be deemed to be more Bayesian, since it takes into account the distributions of generating observations. It is also not restricted to splitting equivalence classes but aims to minimize the conditional entropy of the posterior (or any other utility function). This reason, in particular, makes this method more general and precise by allowing it to, for example, suggest particular interventions several times if needed, instead of only suggesting the node with which to intervene without saying anything about how many measurements to take. However, because Murphy's method is computationally demanding and since sampling can affect the reliability/precision of the method, using the equivalence class based method becomes more attractive after about $n > 12$.

The active learning methods could also be developed towards better realistic applicability by making the cost of actions uneven and especially making the observations cheaper than interventions. The methods should also take into account the possibility of imperfect interventions. The idea of extending the methods to being able to suggest measurements from multiple different sources in an active learning fashion (for example by encoding them in priors) is also worth exploring.

5. ACKNOWLEDGMENTS

This work was supported by the Academy of Finland (application number 213462, Finnish Programme for Centres of Excellence in Research 2006-2011) and by grants R01 GM072855 and P50 GM076547 from NIH/NIGMS.

6. REFERENCES

[1] D. Heckerman, D. Geiger, and D. M. Chickering, "Learning Bayesian networks: The combination of knowledge and statistical data," *Mach. Learn.*, vol. 20, no. 3, pp. 197–243, 1995.

[2] T. Verma and J. Pearl, "Equivalence and synthesis of causal models," in *UAI '90: Proceedings of the Sixth Annual Conference on Uncertainty in Artificial Intelligence*, New York, NY, USA, 1991, pp. 255–270, Elsevier Science Inc.

[3] J. Tian and J. Pearl, "Causal discovery from changes," in *UAI '01: Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence*, San Francisco, CA, USA, 2001, pp. 512–521, Morgan Kaufmann Publishers Inc.

[4] I. Pournara and L. Wernisch, "Reconstruction of gene networks using Bayesian learning and manipulation experiments.," *Bioinformatics*, vol. 20, no. 17, pp. 2934–2942, Nov 2004.

[5] K. Murphy, "Active learning of causal Bayes net structure," Technical Report, University of California, Berkeley, USA, 2001.

[6] S. Tong and D. Koller, "Active learning for structure in Bayesian networks," in *Proceedings of the International Joint Conference on Artificial Intelligence*, 2001.

[7] R. Bonneau, M. T. Facciotti, D. J. Reiss, et al., "A predictive model for transcriptional control of physiology in a free living cell.," *Cell*, vol. 131, no. 7, pp. 1354–1365, Dec 2007.

[8] M. T. Facciotti, D. J. Reiss, M. Pan, et al., "General transcription factor specified global gene regulation in archaea," *Proceedings of the National Academy of Sciences*, vol. 104, no. 11, pp. 4630–4635, 2007.

[9] T. Ideker, V. Thorsson, A. F. Siegel, and L. E. Hood, "Testing for differentially-expressed genes by maximum-likelihood analysis of microarray data," *Journal of Computational Biology*, vol. 7, no. 6, pp. 805–817, 2000.

[10] K. Sachs, O. Perez, D. Pe'er, et al., "Causal protein-signaling networks derived from multiparameter single-cell data.," *Science*, vol. 308, no. 5721, pp. 523–529, Apr 2005.

EFFECTS OF DISEASE-RELATED MUTATIONS ON TRANSCRIPTION FACTOR BINDING

Kirsti Laurila and Harri Lähdesmäki

Department of Signal Processing, Tampere University of Technology,
P.O. Box 553, FI-33101 Tampere, Finland
kirsti.laurila@tut.fi, harri.lahdesmaki@tut.fi

ABSTRACT

Many diseases are caused by hereditary mutations. So far, most of the identified mutations affect the coded protein sequence. However, an increasing number of the identified disease-related mutations occur in gene regulatory sequences. These mutations pose a threat to influence the mechanism by which a cell regulates the transcription of its genes. Here we have studied the effect of mutations on transcription factor binding affinity computationally. We have compared our results with experimentally verified cases where a mutation in the gene regulatory region either creates a new transcription factor binding site or deletes a previously existing one. We have also investigated the statistical properties of the changes on transcription factor binding affinity according to mutation type. Although accurate binding site prediction is difficult in general, our results demonstrate that computational analysis can provide valuable information about the effect of mutations on transcription factor binding sites. The analysis results also give a useful test set for the in vitro studies of regulatory mutation effects.

1. INTRODUCTION

Millions of single nucleotide polymorphisms (SNPs) are identified in the human genome. The majority of these SNPs are neutral but some of them are linked to hereditary diseases. Most of these disease-causing mutations alter the protein sequence, but a set of mutations are identified to occur in gene regulatory sequences. These mutations may cause a significant change in individuals phenotype by increasing or decreasing the gene expression levels. Some examples of this have been verified experimentally. The expression level of the gene for a 91-kD glycoprotein component of the phagocyte oxidase (gp91-*phox*) are decreased because of promoter region mutations that are are associated with X-linked chronic granulomatous disease [1]. Moreover, with Alzheimer disease patients, abnormal high expression levels of the amyloid precursor protein (APP) were measured in vitro when studying three point mutations in the APP promoter region [2].

Although all the mechanisms of gene expression regulation are not known, the mutations in the promoter regions may cause wrong transcription factor (TF) binding and this may in turn have effect on transcription levels. For instance, it has been shown experimentally that the point mutation T→C at 77 nucleotides upstream of the transcription starting site (TSS) of the δ-globin gene (HBG) changes the binding affinity of the TF GATA-1 and this also alters the expression levels of the gene. This mutation is associated with a hereditary disease δ-thalassemia [3]. Another example is described in [4] where a point mutation in 292 nucleotides upstream of the TSS of the reticulocyte-type 15-lipoxygenase-1 (ALOX15) gene causes a new transcription factor binding site (TFBS) for the TF SPI1. This again causes three-fold expression levels compared with wild type gene expression. ALOX15 has a role in the development of asthma and some other diseases.

Mutation effect on TF binding has been studied computationally in [5], where authors have used the change of a score computed based on position specific scoring matrixes (PSSMs) to infer if the binding of some TF changes. This can be problematic since a single nucleotide change usually causes a very small change in score and one cannot directly say that whether this change of score is significant or not. This fact was found when they compared the scores of mutations that are known to affect TF binding with the scores of background substitutions [5].

In this paper, we use a similar approach as in [5] to analyze the regulatory mutations and how they affect the TF binding. However, we use the p-values to compare the wildtype and mutated cases and to get the results of different genes and TFs comparable. We also study if some type of mutation is more significant than the others. This is because the DNA bending ability is known to be different for separate dinucleotide steps [6], [7]. Further, it has been found that contacts between TFs and purines are especially important and because the bending of DNA has an effect on TF binding [8], [9], [10].

2. METHODS

The mutations used in this study were the regulatory mutations from Human Gene Mutation Database (HGMD) [11]. The regulatory mutations dataset was filtered to contain only those mutations that occur upstream from transcription or translation starting sites. Altogether we used 474 mutations in 256 genes.

PSSMs are a widely used in predicting TFBSs and we

apply them in our analysis as well [12], [13]. PSSMs were collected from Transfac (Release 10.3) [14] and Jaspar [15],[16]. Only those matrixes that have been built (at least partially) using human sequences were used. After this selection, we had 496 matrixes for 343 different TFs.

The score for TF binding to the DNA sequence x_1^n was computed by

$$S(x_1^n) = \frac{P_{TF}(x_1^n)}{P_{bg}(x_1^n)}, \quad (1)$$

where $P_{TF}(x_1^n)$ is the probability computed by PSSM and $P_{bg}(x_1^n)$ is the background probability. We added a small pseudo count (0.005) to all elements in PSSM to prevent zero probabilities. As a background model, we used a third order Markov model whose parameters were computed from the promoter sequences of all human genes. As a promoter sequence, we considered commonly used 5000 bases upstream from the start of the first (according to 5' end) annotated mRNA sequence of the gene. However, promoter sequences were not allowed to overlap. The promoter sequences we used were collected from annotated sequence files (gbk-files) of human chromosomes. These files were downloaded from ftp-site of National Center for Biotechnology Information.

We computed the scores for the wildtype and the mutated sequences of our regulatory mutations dataset. Since location of mutation in putative binding sites is not known, we computed the scores for all locations within PSSM. In view of the fact that the distributions were very different for each PSSM, we did not compare the scores but computed the p-values for each mutation. To get the reference distribution for the p-value estimation, the scores were computed for each position of each promoter sequence.

Nucleotides can be divided in purines (denoted by R, consists of bases A and G) and pyrimidines (denoted by Y, bases C and T). By these classifications the dinucleotides can be divided into four classes, RR, YY, YR and RY. Further, for single point mutation the mutations can be divided into 8 groups whether the mutation is in the first or second nucleotide. We divided mutations into these classes, so that each mutation occurred both in the first and in the second nucleotide. Each mutation class was studied separately.

We made a literature search for known mutations affecting TF binding. We collected 6 experimentally proven mutations from articles and rSNP_DB [17]. These mutations were used to set a threshold for a relevant change in binding affinity.

3. RESULTS

We evaluated the effect of the experimentally verified mutations on TF binding by PSSM scores. The list of mutations and their p-values are at Table 1. All mutations showed a big change in p-value (over 0.2) between the wildtype and mutated sequence. However, the p-values of the sequence which has stronger affinity to TF were quite high in some cases i.e. the binding site was quite not statistically significant. Nevertheless, even weaker binding sites can be important, since it has been recently shown

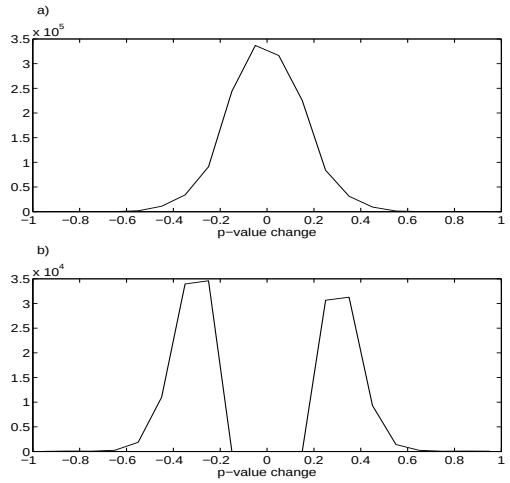


Figure 1. Distributions of the p-value changes. a) All changes. b) Only changes that exceeded the thresholds.

that models which include weak binding sites predict the expression patterns better than those models from which the weak binding sites are excluded [21].

For the experimentally verified mutations, big p-value changes are found for several PSSMs of a single TF. For example for the mutation in hemoglobin gamma G(HBG2) promoter, the p-values corresponding to 4 out of 7 PSSMs for TF SP1 showed a difference in binding affinity. However, one of the matrixes showed the change in two different matrix positions, which suggests that all of the matrixes are not very specific to the binding site.

We computed the change in p-value for each mutation (the wildtype sequence − the mutated sequence) for each TF. This p-value change was considered as a score to measure the change in TF's affinity to bind. The distribution of changes are shown in Figure 1a). Based on the experimentally verified cases we considered the change to be relevant if the p-value change (absolute value) was over 0.3 or the change was over 0.2 and p-value of either the wildtype or the mutated sequence was under 0.3. Approximately 11% of changes exceeded these boundaries. The set of experimentally verified mutations is relatively small and that prevents us from inferring more conservative thresholds without loosing too many verified cases. Current knowledge does not allow us to discriminate true and false changes more carefully (see e.g.[5]). This choice of thresholds, however, results in a set of predicted binding changes that is enriched for true binding affinity changes. Consequently, despite some false positives, our analysis results provide insights into true mutation effects. Our analysis provides a list of testable hypothesis, ordered according to the significance of mutation effect, that can be readily tested in laboratory to verify the real mutation effect in vitro. Besides, if a particular TF is known to regulate some gene and our analysis provides a big p-value change for the affinity of that TF due to mutation, this provides a strong evidence for the mutation effect and this should be taken into account when studying the disease

Table 1. Experimentally verified mutations and their effect on TF binding. p-values are presented only for those PSSMs that show relevant changes. wt=wildtype, Δp-value=(p-value of wt) $-$ (p-value of mutated sequence), mutation position is relative to TSS, MW=matrix width, POM= mutation position on matrix

gene symbol	mutation	mutation position	TF	MW	POM	effect on binding	Δp-value	p-value of wt	disease	reference
ALOX	A→G	-292	SPI1	6	2	increase	0.356	0.592	(anti)inflammatory effects	[4]
HBD	T→C	-77	GATA1	13	12	decrease	-0.386	0.553	δ-thalassemia	[3]
HBG2	C→G	-202	SP1	10	4	increase	0.274	0.540	hereditary persistence of fetal hemoglobin	[18]
HBG2	C→G	-202	SP1	10	5	increase	0.402	0.702	"	[18]
HBG2	C→G	-202	SP1	13	6	increase	0.658	0.861	"	[18]
HBG2	C→G	-202	SP1	10	4	increase	0.373	0.653	"	[18]
HBG2	C→G	-202	SP1	10	4	increase	0.206	0.420	"	[18]
PROC	T→C	-14	HNF-1	15	7	decrease	-0.216	0.265	protein C deficiency	[19]
UROS	C→A	-90	CP2	18	13	decrease	-0.207	0.143	congenital erythro-poietic porphyria	[20]
UROS	C→A	-90	CP2	11	11	decrease	-0.274	0.164	"	[20]
UROS	T→C	-70	GATA1	14	8	decrease	-0.317	0.085	"	[20]
UROS	T→C	-70	GATA1	13	7	decrease	-0.206	0.038	"	[20]

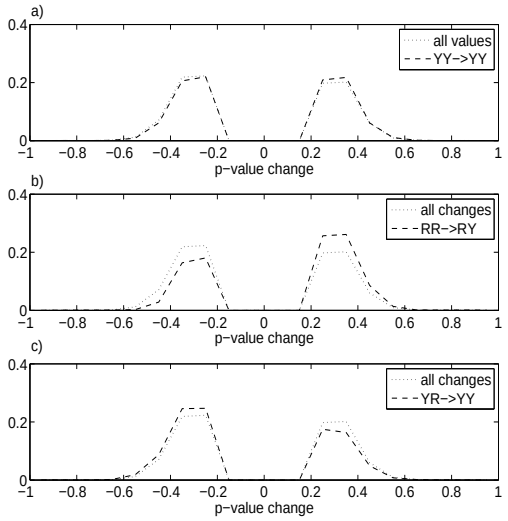


Figure 2. Distributions of the p-value changes in three different dinucleotide mutation types. a) YY→YY b) RR→RY c) YR→YR, Y=pyrimidine, R=purine

mechanisms on molecular level.

The distribution of the p-value changes of the relevant dataset can be seen in Figure 1b). It can be seen that the left side of the bimodal distribution has somewhat larger area than the right side i.e. the mutations cause more often the loss of TF binding affinity than create a new TFBS.

We computed the distributions of the p-value changes for each mutation type (16 dinucleotide classes). The distributions for different classes varied remarkably. In the Figure 2 is the distribution of the p-value changes in three different cases where mutation is in the second nucleotide. In all of the three plots there is also the distribution of the all p-value changes that exceeded the thresholds, as a reference distribution. It can be inferred based on the plots that the mutation type affects the binding affinity change differently. For mutations YY→YY (Figure 2a), the probability of formation a new TFBS is as probable as a disruption of an old binding site. This was also the case for mutation type RR→RR when mutation occurred in the second nucleotide and for RY→YY, RY→RY, YR→YR and YY→YY, if the mutation was in the first nucleotide. For mutations RR→RY and YY→YR, RR→YR and YY→RY the mutation more often caused a new binding site than disrupted an existing one (Figure 2b)). The rest of the mutations caused more likely the removal of an old binding site than making a new one as can be seen in an example in Figure 2c). The results suggest that purine-pyrimidine and pyrimidine-purine dinucleotides are in important roles in TF binding. It has been previously shown that pyrimidine-purine steps are flexible allowing the DNA strands to form sharp kinks [6]. This is important for TF which usually bends the DNA or binds to a bent DNA. Nevertheless, such flexibility is not shown to occur with all purine-pyrimidine steps. However, an RY step GC can also form more conformations than for example the AA and TT steps [7].

4. CONCLUSION

We have shown that regulatory mutations can change the TF binding affinity remarkably. This does not originate only from a single nucleotide mutation but also the type the surrounding nucleotides.

PSSMs are a widely used method to model TF binding. A big problem of PSSMs is, however, the number of false positives in predicting TFBSs. As our studies with experimentally verified TFBSs and the mutations affecting them showed, the PSSM modeling does not assign an extremely high p-values to TFBSs. This can be because of PSSM matrixes which does not have any corre-

lation between different bases. Our studies have shown that the dinucleotides in TFBSs affect the binding significantly. This is most likely caused by the ability of DNA strands to bend. Since different DNA-binding domains of TFs have different binding mechanisms and demands for DNA bending it could be more appropriate to study each TF family separately.

In the future it is important to incorporate additional knowledge into TF binding prediction. Previously, models that combine the nucleosome positions or Chromatin ImmunoPrecipitation on chip (ChIP-chip) data are shown to predict TF binding better than pure PSSMs [22], [23]. Other additional data sources can be also combined to models, for example DNase hypersensitive sites or conservation data. It should be also taken into account that in the cell, there is not just a single TF type present at a certain time, but the situation can be thought to be a competition between different TFs and other molecules to bind the DNA strand [21]. Thus, the TF binding differs in different states of the cell depending on the TFs present and their concentrations.

5. REFERENCES

[1] P. E. Newburger, D. G. Skalnik, P. J. Hopkins, A. A. Eklund, and J. T. Curnutte, "Mutations in the promoter region of the gene for gp91-phox in x-linked chronic granulomatous disease with decreased expression of cytochrome b558," *J Clin Invest*, vol. 94, pp. 1205–11, Feb 1994.

[2] J. Theuns, N. Brouwers, S. Engelborghs, K. Sleegers, V. B. V, E. Corsmit, T. de Pooter, C. M. van Duijn, P. P. de Deyn, and C. van Broeckhoven, "Promoter mutations that increase amyloid precursor-protein expression are associated with Alzheimer disease," *Am J Hum Genet*, vol. 26, pp. 936–46, Jun 2006.

[3] M. Matsuda, N. Sakamoto, and Y. Fukunaki, "δ-thalassemia caused by disruption of the site for an erythroid-specific transcription factor, GATA-1, in the δ-globin gene promoter," *Blood*, vol. 80, pp. 1347–51, 1992.

[4] J. Wittwer, J. Marti-Juan, and M. Hersberg, "Functional polymorphism in ALOX15 results in increased allele-spesific transcription in macrophages through binding of the transcription factor SPI1," *Hum Mutat*, vol. 27, no. 2, pp. 78–87, 2006.

[5] M. C. Andersen, P. G. Engström, S. Lithwick, D. Arenillas, P. Eriksson, B. Lenhard, W. W. Wasserman, and J. Odeberg, "In silico detection of sequence variations modifying transcriptional regulation," *PLoS Comput Biol*, vol. 4, pp. e5, Jan 2008.

[6] M. Suzuki, D. Loakes, and N. Yagi, "DNA conformation and its changes upon binding transcription factors," *Adv Biophys*, vol. 32, pp. 53–72, 1996.

[7] A. A. Travers, "The structural basis of DNA flexibity," *Philos Transact A Math Phys Eng Sci*, vol. 15, pp. 1423–38, Jul 2004.

[8] A. Sarai and H. Kono, "Protein-DNA recognition patterns and predictions," *Annu Rev Biophys Biomol Struct*, vol. 34, pp. 379–98, 2005.

[9] R. E. Harrington, "DNA curving and bending in protein-dna recognition," *Mol Microbiol*, vol. 6, pp. 2549–55, Sep 1992.

[10] C. O. Pabo and R. T. Sauer, "Transcription factors: Structural families and principles of DNA recognition," *Annu Rev Biochem*, vol. 61, pp. 1053–95, 1992.

[11] P. D. Stenson, E. V. Ball, M. Mort, A. D. Phillips, J. A. Shiel, N. S. Thomas, S. Abeysinghe, M. Krawczak, and D. N. Cooper, "The Human Gene Mutation Database (HGMD ®): 2003 update," *Hum Mutat*, vol. 21, pp. 577–581, 2003.

[12] G. D. Stormo, "DNA binding sites:representation and discovery," *Bioinformatics*, vol. 16, pp. 1416–23, Jan 2000.

[13] R. Staden, "Computer methods to locate signals in nucleic acid sequences," *Nucleic Acids Res*, vol. 12, pp. 505–519, Jan 1984.

[14] V. Matys, E. Fricke, R. Geffers, E. Gossling, M. Haubrock, R. Hehl, K. Hornischer, D. Karas, A. E. Kel, O. V. Kel-Margoulis, D. U. Kloos, S. Land, B. Lewicki-Potapov, H. Michael, P. Munch, I. Reuter, S. Rotert, H. Saxel, M. Scheer, S. Thiele, and E. Wingender, "TRANSFAC: transcriptional regulation, from patterns to profiles," *Nucleic Acids Res*, vol. 31, pp. 374–8, 2003.

[15] A. Sandelin, Walkema, P. Engström, W. Wasserman, and B. Lenhard, "JASPAR: an open access database for eukaryotic transcription factor binding profiles," *Nucleic Acids Res*, vol. 32, pp. D95–7, 2004.

[16] B. Lenhard and W. Wasserman, "TFBS: Computational framework for transcription factor binding site analysis," *Bioinformatics*, vol. 18, pp. 1135–6, 2002.

[17] J. V. Ponomarenko, G. V. Orlova, M. P. Ponomarenko, S. V. Lavryushev, and T. I. Merkulova, "rSNP_Guide: a database documenting influence of substitutions in regulatory gene regions onto their interaction with nuclear proteins and predicting protein binding sites, damaged or appeared de novo due to these substitutions," in *Proceedings of BGRS'2000*, 2000, pp. 69–72.

[18] F. S. Collins, C. J. jr Stoeckert, G. R. Serjeant, B. G. Forger, and S. M. Weissman, "G gamma beta+ hereditary persistence of fetal hemoglobin: cosmid cloning and identification of a specific mutation 5' to the G gamma gene," *Proc Natl Acad Sci U S A*, vol. 81, pp. 4898–8, Aug 1984.

[19] L. P. Berg, D. A. Scopes, A. Alhaq, V. V. Kakkar, and D. N. Cooper, "Disruption of a binding site for hepatocyte nuclear factor 1 in the protein C gene promoter is associated with hereditary thrombophilia," *Hum Mol Genet*, vol. 3, pp. 2147–52, Dec 1994.

[20] C. Solis, G. I. Aizencan, K. H. Astrin, D. F. Bishop, and R. J. Desnick, "Uroporphyrinogen III synthase erythroid promoter mutations in adjacent GATA1 and CP2 elements cause congenital erythropoietic porphyria," *J Clin Invest*, vol. 107, pp. 753–62, Mar 2001.

[21] E. Segal, T. Raveh-Sadka, M. Schroeder, U. Unnerstall, and U. Gaul, "Predicting expression patterns from regulatory sequence in Drosophila segmentation," *Nature*, vol. 451, pp. 535–540, Jan 2008.

[22] L. Narlikar, Raluca, and A. J. Hartemink, "A nucleosome-guided map of transcription factor binding sites in yeast," *PLoS Comput Biol*, pp. 2199–208, 2007.

[23] H. Kim, K. J. Kechris, and L. Hunter, "Mining discriminative distance context of transcription factor binding sites on ChIP enriched regions," in *ISBRA*, 2007, pp. 338–49.

SBML ODE SOLVER LIBRARY: EXTENSIONS FOR INVERSE ANALYSIS

James Lu[1], *Stefan Müller*[1], *Rainer Machné*[2], *Christoph Flamm*[2]

[1]Johann Radon Institute for Computational and Applied Mathematics (RICAM),
Austrian Academy of Sciences,
Altenbergerstrasse 69, 4040 Linz, Austria
james.lu@oeaw.ac.at, stefan.mueller@oeaw.ac.at
[2]Institute of Theoretical Chemistry, University of Vienna,
Währingerstrasse 17, 1090 Wien, Austria
raim@tbi.univie.ac.at, xtof@tbi.univie.ac.at

ABSTRACT

The SBML ODE Solver Library (SOSlib) [1] is a C/C++ programming library for the symbolic and numerical analysis of ODE systems derived from biochemical reaction networks encoded in the Systems Biology Markup Language (SBML) [2]. It is written in ANSI/ISO C and distributed under the terms of the GNU Lesser General Public License (LGPL).

Recent efforts in the development of SOSlib have been focused on extensions that allow one to perform not only the *forward analysis* but also the *inverse analysis* of biochemical models. In particular, SOSlib has been extended with forward and adjoint capabilities to enable the identification of model parameters and initial conditions from (noisy) experimental data, measured either continuously or at discrete time points. Via on-the-fly compilation of right-hand-side functions and Jacobian routines, a significant speed-up in numerical integration has been achieved.

1. SIMULATION AND SENSITIVITY ANALYSES

We denote the underlying ODE system and initial condition as, respectively,

$$\begin{aligned} \dot{x}(t) &= f(x,\alpha), \\ x(0) &= x_0, \end{aligned} \tag{1}$$

where $x \in \mathbb{R}^n$ is the state variable, $\alpha \in \mathbb{R}^m$ the parameters and $f(x,\alpha) : \mathbb{R}^{n+m} \to \mathbb{R}^n$ the parameter-dependent vector field. To allow for the sensitivity analysis of (1), we assume the differentiability of $f(x,\alpha)$ with respect to both x and α.

In studying many biological models, one would like to not only obtain the solution $x(t)$ for a given set of nominal parameter values but also to examine its parametric dependence. This can be computationally studied by solving the forward sensitivty equations as discussed in Section 1.1. For applications such as parameter identification and optimization of biological systems, one is interested in computing the parametric dependence not for the whole time-series but only for certain *functionals* that map solutions to real numbers. For these applications, the adjoint approach to sensitivity analysis as discussed in Section 1.2 is the preferred method in terms of the computational efficiency.

1.1. Forward Sensitivity Analysis

The simulation of the ODE system (1) can be thought of as applying an operator $\mathcal{F}$ which takes as input the initial condition and parameter values, mapping it to the ODE solution. That is, we have $\mathcal{F} : (x_0,\alpha) \in \mathbb{R}^{n+m} \to x(t) \in C^1([0,T],\mathbb{R}^n)$, where C^1 denotes the space of continuously differentiable functions. One can then consider differentiations of the operator $\mathcal{F}$ either at an algorithmic level, or at a mathematical level. For the former, one would symbolically "differentiate" the steps taken in the chosen numerical algorithm in going from the input data, x_0, α, to the N-point numerical approximation over the requested time interval, $\{x(t_0),\cdots,x(t_N)\}$. Such an approach is known as *automatic differentiation* (AD) [3].

In our work, we take the forward sensitivity equations approach whereby one formally differentiates the operator $\mathcal{F}$ with respect to the initial concentrations, x_0, and the parameter values, α. In this case, the differential equations are first derived and then arbitrary numerical methods can be applied to solve the resulting system of equations. This is the approach that we have implemented in SOSlib and discussed in Sections 1.1.1 and 1.1.2. In Section 1.1.3, we discuss an application of the forward sensitivity solver, namely for computing the Fisher Information Matrix from which a lower-bound for the standard deviation in the estimated parameters can be derived [4].

1.1.1. Initial condition sensitivity

First, we consider variations in the solution arising from variations in the initial conditions. One supposes that the parameters α are fixed, and differentiates the original ODE system with respect to the initial conditions, which is then reflected in the initialization for the sensitivity variables. If we denote s^i as the set of sensitivity variables (of dimension n) corresponding to the perturbation of the system (1) with respect to the ith component of x_0, then one obtains the following equations for the $n \times n$ sensitivity

system $\{s^1(t), \cdots, s^n(t)\}$:

$$\begin{aligned} \dot{s}^i(t) &= f_x(x(t), \alpha) s^i(t), \\ (s^i)_j(0) &= \delta_{ij}, \end{aligned}$$

where the notation $(s^i)_j$ denotes the j-th component of s^i, f_x is the Jacobian matrix and δ_{ij} is the Kronecker delta defined as $\delta_{ii} = 1$ and $\delta_{ij} = 0$ otherwise (for $1 \leq i, j \leq n$).

1.1.2. Parameter sensitivity

Next, we consider variations in the solution arising from variation in the parameters. Since no perturbation in the initial state is introduced, it is easy to see that the parametric sensitivity variables have the homogeneous initial condition. One thus obtains the following $n \times m$ linear ODE system for $\{s^1(t), \cdots, s^m(t)\}$:

$$\begin{aligned} \dot{s}^i(t) &= f_x(x(t), \alpha) s^i(t) + f_{\alpha_i}(x(t), \alpha), \\ s^i(0) &= 0. \end{aligned}$$

Using the symbolic differentiation capability of SOSlib, the expressions f_x, f_{α_i} are computed and passed to be called from the CVODES solver.

1.1.3. Application: Fisher Information Matrix

Let us consider the problem of estimating m parameters from time-course data. For each data-point t_i, let us denote $S(t_i)$ as the $n \times m$ matrix of sensitivity solutions:

$$S(t_i) = \begin{pmatrix} s_1^1(t_i) & \cdots & s_1^m(t_i) \\ \vdots & \ddots & \vdots \\ s_n^1(t_i) & \cdots & s_n^m(t_i) \end{pmatrix}.$$

If we denote the covariance matrix of (discrete) measurement errors as V, then the Fisher Information Matrix (F) is given by the following formula:

$$F = \sum_{t_i=1}^{N} S(t_i)^T V^{-1} S(t_i).$$

Thus, using the forward sensitivity solver, F can be computed by simply summing matrix products over experimental time points. For deriving parameter confidence intervals from F, refer to [4].

1.2. Adjoint Sensitivity Analysis

Given a functional of interest, $J : C^1([0,T], \mathbb{R}^n) \to \mathbb{R}$, we consider the following Lagrangian,

$$L(x, \psi) = J(x) + \langle \psi, \dot{x} - f(x, \alpha) \rangle_{L_2},$$

where $\psi(t) \in C^*([0,T], \mathbb{R}^n)$ is the associated adjoint variable (of bounded variation) in the dual space of continuous functions, serving as Lagrange multiplier to the ODE constraint $\dot{x} - f(x, \alpha) = 0$. Integration by parts of the above gives

$$\begin{aligned} L(x, \psi) = J(x) + \langle -\dot{\psi}, x \rangle_{L_2} - \langle \psi, f(x, \alpha) \rangle_{L_2} \\ + \psi(T) x(T) - \psi(0) x(0). \end{aligned} \tag{2}$$

The equations for the adjoint variable are obtained by considering the variational equations $\delta L(x, \psi; \delta x) = 0$, for all variations: $\delta x \in C^1([0,T], \mathbb{R}^n)$, $\delta x(0) = 0$. In Sections 1.2.1 and 1.2.2, we show the adjoint ODE systems for cases where the objective corresponds to either continuously or discretely measured experimental data respectively. For a discussion on the adjoint equations and its numerical solution in the general context of differential-algebraic equations (DAEs), refer to [5].

After the adjoint system is solved, the objective gradients with respect to the parameters and initial conditions are simply obtained as (refer to eqn. (2), noting the implicit dependency of $x(t)$ on x_0 and α):

$$\begin{aligned} \frac{dJ(x(\alpha, x_0))}{d\alpha_j} &= \frac{\partial L}{\partial \alpha_j} = -\langle \psi, f_{\alpha_j}(x, \alpha) \rangle_{L_2} \\ \frac{dJ(x(\alpha, x_0))}{d(x_0)_j} &= \frac{\partial L}{\partial (x_0)_j} = -\psi(0)_j. \end{aligned} \tag{3}$$

We remark that the adjoint approach to computing the objective gradient is especially attractive for biological systems of high parameter dimensions. In particular, the dimension of the adjoint variable is the same as that of the state, independent of the number of parameters. After this adjoint system has been numerically integrated, equation (3) shows that gradients of the given objective can then be computed by simply taking inner products over the time domain, or evaluating the adjoint variable at time $t = 0$. Thus, gradient calculations can be done essentially at constant time, independent of the number of parameters present in the model.

1.2.1. Continuous data

Without the loss of generality but for the simplicity of presentation, in what follows we assume a specific form of the objective function. Namely, we consider parameter identification applications where one tries to minimize objectives measuring the data mis-match. That is, if no regularization is used, such an objective may take the form:

$$J_{cont}(x) = \frac{1}{2} \int_0^T (x(t) - x_{data}(t))^2 dt. \tag{4}$$

where $x_{data}(t) \in C^1([0,T], \mathbb{R}^n)$ is some given experimental time-series.

From the Lagrangian expression in (2), setting $\delta L(x, \psi; \delta x) = 0$ gives rise to the following terminal-value problem for the adjoint variable, $\psi(t)$:

$$\begin{aligned} \psi(T) &= 0, \\ \dot{\psi}(t) &= -f_x(x(t), \alpha)^T \psi(t) \\ &\quad + (x(t) - x_{data}(t)). \end{aligned} \tag{5}$$

Once the expression for the objective has been provided to SOSlib and the data $x_{data}(t)$ is read in, the system (5) can again be numerically integrated (backwards in time) using the adjoint solver provided by CVODES.

1.2.2. Discrete data

Here, we consider objectives of the following form:

$$J_{disc}(x) = \frac{1}{2}\sum_{k=1}^{N}(x(t_k) - x_{data}(t_k))^2, \qquad (6)$$

consisting of the sum of the data mis-match over the (discrete) time points, $\{t_1, \cdots, t_N\}$. The objective (6) may be rewritten as:

$$J_{disc}(x) = \frac{1}{2}\sum_{k=1}^{N}\int_0^T \delta(t - t_k)(x(t) - x_{data}(t))^2 dt, \quad (7)$$

where $\delta(t - t_k)$ is the delta distribution with the sifting property that for all continuous functions $g(t)$,

$$\int_{-\infty}^{\infty} g(t)\delta(t - t_k)dt = g(t_k).$$

Now that the objective (1.2.2) is of the integral form, one might attempt to write down the adjoint system analogous to (5):

$$\begin{aligned} \psi(T) &= 0, \\ \dot{\psi}(t) &= -f_x(x(t), \alpha)^T \psi(t) \\ &\quad + \sum_{i=k}^{N} \delta(t - t_k)(x(t) - x_{data}(t_k)). \end{aligned}$$

The above ODE system only has meaning in the sense of distributions and no ODE solver can be applied directly without taking special care at the data time points, $\{t_i\}$. Instead, one can solve it by treating it as a concatenation of piecewise continuous trajectories. More specifically, with the terminal condition being $\psi(T) = 0$, we integrate over time intervals in between the set of data time points and introduce jumps at the times when data is given:

$$\begin{aligned} \text{FOR} \quad : \quad & k = N, N-1, \cdots, 1 \\ & \dot{\psi}(t) = -f_x(x(t), \alpha)^T \psi(t), t \in [t_k, t_{k+1}) \\ & \psi(t_k^-) = \psi(t_k^+) - (x(t) - x_{data}(t_k)). \end{aligned}$$

Thus, the adjoint profile can be computed by providing start- and stop-time points $\{t_i\}$ to the CVODES adjoint solver to integrate it piecewise and adding to the adjoint variable in between the integration calls.

2. COMPILATION

For parameter identification and optimal control applications, the ODE and sensitivity solvers typically need to be called many times. In order to study systems with high dimensional parameter space within reasonable compute time, it is important to be able to evaluate the right-hand-side functions and Jacobians of the ODE systems efficiently.

Motivated by a need to speed up the solvers, we have implemented two different versions of on-the-fly compilation of these functions. First, we take use of libSBMLs abstract-syntax-tree representation of model equations to directly construct machine code for all equations of the model. As this approach is highly platform-specific (currently, we provide 32 and 64 bit architecture machine code for both Windows and Unix systems), SOSlib also allows for the conversion from the vector-field and associated functions to C source code and makes use of a preinstalled C/C++ compiler (e.g. `gcc`). Both approaches result in around an order of magnitude decrease in the compute time for some test cases; see Section 3.2.

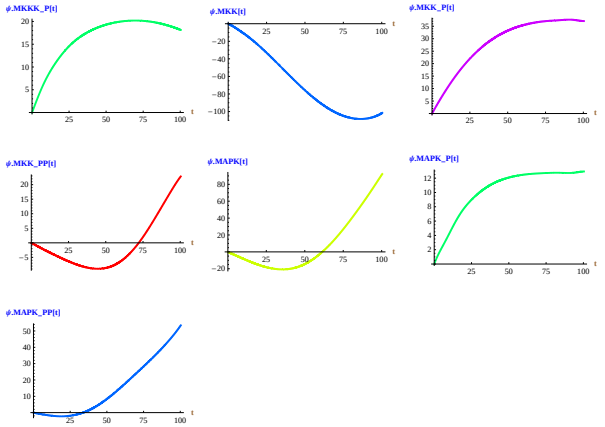


Figure 1. Adjoint solution profile for parameter estimation: using continuous data

3. NUMERICAL DEMONSTRATIONS

Here, we consider parameter idenfication examples formulated as finding the minimizer of the data mis-match. In Section 3.1, we illustrate the difference in the adjoint solution profiles using continuous and discrete data. In Section 3.2, we show the objective convergence using an interior-point optimization solver and demonstrate the speed-up gained by model compilation.

3.1. Adjoint profiles

Here we examine adjoint solutions at the first step of the parameter identification procedure. Figures 1 and 2 illustrate the adjoint solution profiles corresponding to using continuous and discrete data, for a simple oscillatory model of a signaling cascade taken from the BioModels database [1]. In both cases, we use artificial data obtained by simulating the model at its nominal parameters. In Figure 2, one can easily spot the jumps in the adjoint profiles at the 10 data points. Despite this, one can observe some similarity in the general shapes of the profiles given in Figures 1 and 2. In fact, as the number of (discretely) sampled data points increases, one would expect the adjoint solutions to converge in the L_1 norm.

3.2. Convergence and speed

Here we consider the identification of 36 parameters in the 3-gene model as used in [6]. In particular, we use noiseless, artificial data corresponding to the original parameters and start the parameter identification procedure from

[1] `http://www.ebi.ac.uk/biomodels/`

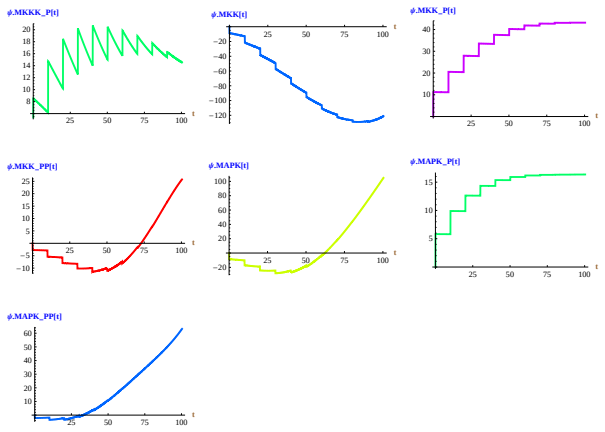


Figure 2. Adjoint solution profile for parameter estimation: using discrete data

parameter values being an order of magnitude smaller than the true ones. To carry out the minimization of the above objective we employ IpOpt [7], an interior-point (local) optimization algorithm. The values for the objective and gradient are provided by the forward and adjoint solvers of SOSlib, respectively.

The convergence in the objective is shown in Figure 3. We observe a 6 orders of magnitude decrease in the objective over 500 optimization iterations using IpOpt [7]. Table 1 gives the number of function evaluation calls to SOSlib as well as the CPU time taken in the numerical integrations. We see that around 1200 forward ODE and 500 adjoint integrations were carried out in the optimization process. The observed difference in the number of objective calls between the compiled and non-compiled results is due to small numerical discrepancies. If the right-hand side and Jacobian functions are not compiled, the time taken for these calculations take 21.54 seconds; when these functions are compiled with `gcc` only 2.64 seconds are needed, thereby achieving nearly an order of magnitude decrease in the computing time.

Table 1: IpOpt calls to SOSlib

	No compilation	`gcc` compilation
# obj. eval.	1222	1251
# grad. eval	500	500
CPU: SOSlib	**21.54 sec.**	**2.64 sec.**

4. CONCLUSIONS

We have demonstrated extensions to SOSlib that allow one to perform inverse analyses of biological models efficiently. In combination with regularization methods [8], these tools enable one to tackle *ill-posed* parameter identification problems that arise in systems biology.

5. ACKNOWLEDGMENTS

We gratefully acknowledge financial support by the Vienna Science and Technology Fund (WWTF), Project Number MA05.

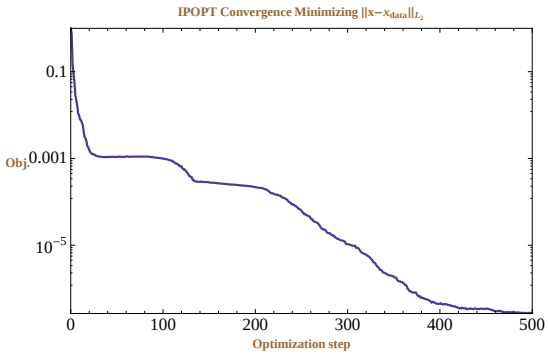


Figure 3. Convergence of objective using the interior point solver, IpOpt

6. REFERENCES

[1] R. Machné, A. Finney, S. Müller, J. Lu, S. Widder, and C. Flamm, "The SBML ODE Solver Library: a native API for symbolic and fast numerical analysis of reaction networks.," *Bioinformatics*, Mar 9 2006.

[2] A. Finney and M. Hucka, "Systems biology markup language: Level 2 and beyond," *Biochem Soc Trans*, vol. 31, no. Pt 6, pp. 1472–1473, Dec 2003.

[3] A. Griewank, "A mathematical view of automatic differentiation," in *Acta Numerica*, vol. 12, pp. 321–398. Cambridge University Press, 2003.

[4] A. Kremling, S. Fischer, K. Gadkar, F. J. Doyle, T. Sauter, E. Bullinger, F. Allgower, and E. D. Gilles, "A benchmark for methods in reverse engineering and model discrimination: problem formulation and solutions.," *Genome Res*, vol. 14, no. 9, pp. 1773–85, 2004.

[5] Y. Cao, S. Li, L. Petzold, and R. Serban, "Adjoint sensitivity analysis for differential-algebraic equations: the adjoint DAE system and its numerical solution," *SIAM J. Sci. Comput.*, vol. 24, no. 3, pp. 1076–1089 (electronic), 2002.

[6] C. G. Moles, P. Mendes, and J. R. Banga, "Parameter estimation in biochemical pathways: a comparison of global optimization methods.," *Genome Res*, vol. 13, no. 11, pp. 2467–74, 2003.

[7] A. Waechter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Math. Program.*, vol. 106, no. 1, pp. 25–57, 2006.

[8] H. W. Engl, M. Hanke, and A. Neubauer, *Regularization of inverse problems*, vol. 375 of *Mathematics and its Applications*, Kluwer Academic Publishers Group, Dordrecht, 1996.

MODELING MOTION OF CONTAMINANT BAP IN CYTOPLASM

Juliane Mai and Sabine Attinger

Department Computational Hydrosystems, Helmholtz-Centre for Environmental Research - UFZ
Permoserstraße 15, 04318 Leipzig, Germany
juliane.mai@ufz.de

ABSTRACT

The response of cells to contaminant stressors like nicotine is of great importance for human health. The focus of the project is to model the way of the contaminant until the entrance of the nucleus. Therefore, in the first step the cell culture surrounded by the fluorescent contaminant is imaged by a laser microscope. Filters and contour extracting algorithms are used to extract the cell geometry. Finally the movement of the contaminant is modeled using reaction-diffusion-equations and random-walk-processes. The long term goal of the project is to understand the influence of contaminant molecules on biological cell functions.

1. INTRODUCTION

Our first step is to model the motion of the contaminant Benzo[a]pyrene BaP within the cytoplasm.
It is well known that a fraction of contaminants is able to react with receptors (AhR) and the motion of larger particles like this complexes is slowed up compared to unbound contaminants. [1] Further, the flow of a receptor-bounded complex is more directed towards the nucleus of the cell than the molecules which are unbounded. [2] So we model two kinds of motion: the normal diffusion via Random-Walk-Process and the directed diffusion via Random-Walk-Process with drift. [3]
It is assumed that there is an equilibrium of the bounded and the unbounded fraction. This means that the association and dissociation rates have to be modeled as well.
The reason of modeling the motion of contaminants is that we need all parameters which describe the behaviour of our substance BaP. In practice, you have the possibility to accomplish FRAP experiments to get all these parameters. [4, 5] At a later date we want to compare our model-parameters with the calculated ones out of the FRAP data. At that time we don't have such data, so we simulate FRAP experiments with different input values and analyse the results we get.

2. MATERIALS AND METHODS

We describe now the steps of modeling: motion of unbounded and bounded particles, association and dissociation rates at equilibrium and FRAP experiments.

2.1. Motion of unbounded Contaminants

The motion of a free particle is modeled by a Random-Walk-Process.
Let the position of the particle at a given time t be (x_t, y_t). The particle jumps within 1 timestep 1 or -1 unit in x-direction and in y-direction. This yields 4 possible positions of the particle after 1 timestep:

$$(x_{t+1}, y_{t+1}) = \begin{cases} (x_t - 1, y_t - 1) \\ (x_t - 1, y_t + 1) \\ (x_t + 1, y_t - 1) \\ (x_t + 1, y_t + 1) \end{cases}$$

The probability of the incidence is the same for every point.

$$P\left[(x_{t+1}, y_{t+1}) = (x_t \pm 1, y_t \pm 1)\right] = \frac{1}{4} \qquad (1)$$

2.2. Motion of bounded Contaminants

The motion of a bounded particle is modeled by a Random-Walk-Process with drift. This means that the compound have a preferred direction.
Let the preferred direction be a point (x_{Dir}, y_{Dir}) and the position of the particle at a given time t be (x_t, y_t). The possible positions of the compound are the same as in section 2.1, the probabilities on the other hand are different. A step in the preferred direction is more probable than a step in the opposite direction. Let the probability of a jump in preferred direction be $p, p \geq \frac{1}{2}$. This yields the probabilities of the 2 possible positions in x-direction:

$$P[x_{t+1} = x_t + s_t] = p \qquad (2a)$$
$$P[x_{t+1} = x_t - s_t] = 1 - p \qquad (2b)$$

whereas:

$$s_t = \text{Sign}[x_{Dir} - x_t] = \begin{cases} 1 & , x_{Dir} > x_t \\ 0 & , x_{Dir} = x_t \\ -1 & , x_{Dir} < x_t \end{cases}$$

The same equations apply to y-direction as well.

2.3. Association and Dissociation

Note, that compounds can unbind and free particles can be bind.

Let B_t the fraction of bounded particles at time t and F_t the fraction of free particles at time t.
The rate of unbinding molecules k_{off} per timestep describes this process of dissociation. On the other hand the parameter k_{on}, which specifies the rate of new bounded molecules per timestep, characterises the process of association.
Now, we can calculate the fractions of the different particles at time $t+1$ out of the fractions at the time t:

$$B_{t+1} = B_t + k_{on} \cdot F_t - k_{off} \cdot B_t \quad (3a)$$
$$F_{t+1} = F_t - k_{on} \cdot F_t + k_{off} \cdot B_t \quad (3b)$$

We assume an equilibrium of free and bounded particles at initial time $t = 0$. The equilibrium situation yields:

$$B_t = const. \quad \forall t \geq 0 \quad (4a)$$
$$F_t = const. \quad \forall t \geq 0 \quad (4b)$$

Making use of the equations (4) equations (3) are simplified:

$$\frac{k_{on}}{k_{off}} = \frac{B_t}{F_t} \quad (5)$$

Futher, we assume that the sum of bounded and unbounded fraction is 1. This yields:

$$B_t = \frac{k_{on}}{k_{on} + k_{off}} \quad (6a)$$
$$F_t = \frac{k_{off}}{k_{on} + k_{off}} \quad (6b)$$

The relationship between dissociation rate and mean binding time BT is well known [4]:

$$k_{off} = \frac{1}{BT} \quad (7)$$

Equation (7) and equation (5) yields :

$$k_{on} = \frac{B_t}{BT \cdot (1 - B_t)} \quad (8)$$

2.4. Simulation of FRAP experiments

As we show in the sections above, we only need a few input parameters to simulate the motion of the contaminant molecules.
First, to guarantee the equilibrium situation during the whole simulation, we need

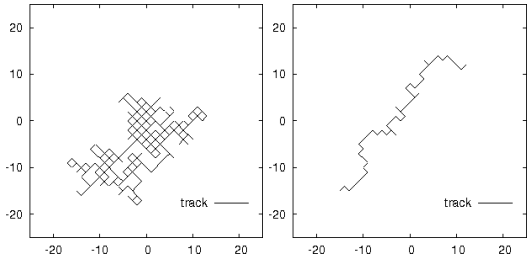
- the mean binding time BT and the fraction of bounded particles B_t or
- the association rate k_{on} and the dissociation rate k_{off}

We choose the first possibility.
Second, we need a direction (x_{Dir}, y_{Dir}) and a probability p for the Random-Walk motion with drift. In the case of modeling contaminants the preferred direction of bounded particles is the position of nucleus. So we modeled the nucleus as a circle with centre (x_{Dir}, y_{Dir}) and radius r_{Dir}.
In case of a particle enters the nucleus we modeled two different kinds of behavior. We assume on the one hand that only bounded particles can be captured by the nucleus and on the other hand that all (bounded and free) particles are captured by the nucleus.
As an application of the motion-model we simulate Fluorescence Recovery After Photobleaching (FRAP) experiments. FRAP is a method of the confocal laser scanning microscopy (cLSM). You are able to assign parameters of diffusion and binding by these experiments. The proceeding of FRAP is to bleach fluorescent particles irreversible within a bleaching spot. Afterwards you monitor the recovery of fluorescent molecules from the outer part of the bleaching spot. They enter the bleaching spot by their motion.
So, we define the radius of a circular bleaching spot r_{Spot} and the centre of the spot $(x_{Spot}, y_{Spot}) = (0, 0)$ as well as the size of the monitored square area a.
For simulation we initialize the model with the number of tracked particles $Samples$, the time steps of simulation $TimeSteps$ and the number of simulations $SimSteps$ we used to create an average recovery.

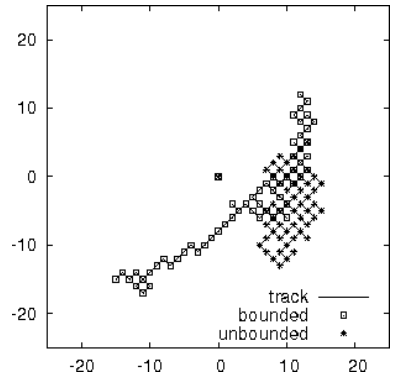
3. RESULTS

3.1. Motion of unbounded und bounded Contaminants

The motion of unbounded contaminants is a diffusion process. A track of on particle is shown in Figure 1(a). On the other hand the motion of contaminants which are bounded by another particle modeled as a diffusion with a drift as you can see in Figure 1(b).



(a) free particle (b) bounded particle



(c) free and bounded particle

Figure 1. track of single particles

3.2. Association and Dissociation

The next step is to integrate the fact that unbounded contaminants can be bounded and the other way around. Now the track is a combination of directed and undirected walk as you can see in Figure 1(c).

3.3. Simulation of a FRAP experiment

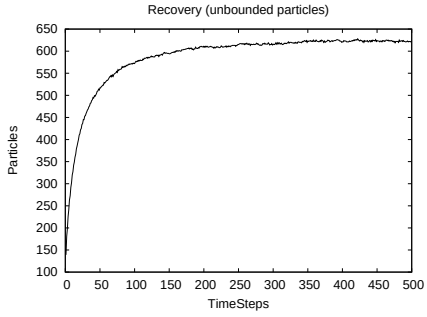
The following simulations are set up with the same parameters unless otherwise noted (see Table 1).

Samples	# of particles	20000
TimeSteps	# of simulated time steps	2000
SimSteps	# of Simulations for averaging	100
a	length of square monitored area	50
r_{Spot}	radius of bleaching spot	5
x_{Dir}	x-coord. of nucleus	15
y_{Dir}	y-coord. of nucleus	15
r_{Dir}	radius of nucleus	5
p	prob. of jump to nucleus	0.55

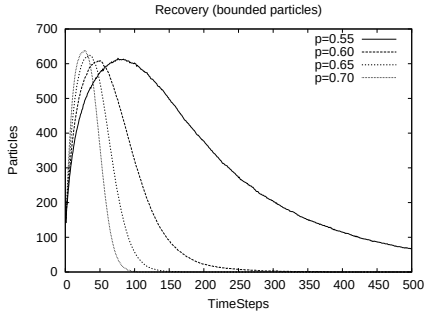
Table 1. simulation parameters

First, we simulate a FRAP experiment with particles which are unbounded, walk undirected and can not enter the nucleus (see Figure 2(a)).

Second, we simulate several FRAP experiments of particles which are bounded and walk with a drift (see Figure 2(b)). We vary the probabilities of a jump into the direction of the nucleus. Note, particles that enter the nucleus are captured in this case.



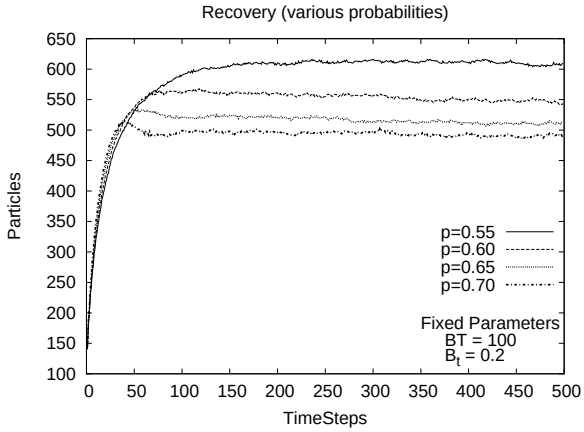
(a) only unbounded particles



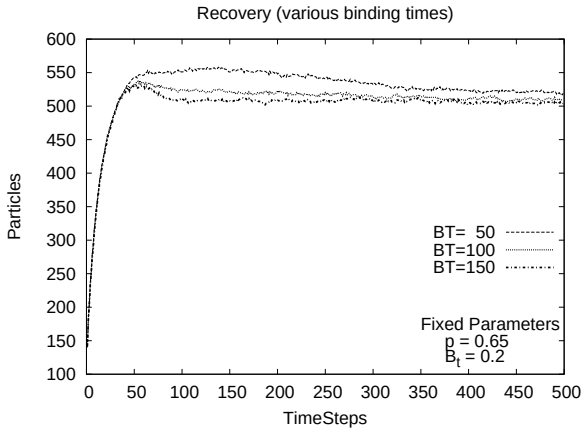
(b) only bounded particles

Figure 2. FRAP simulations

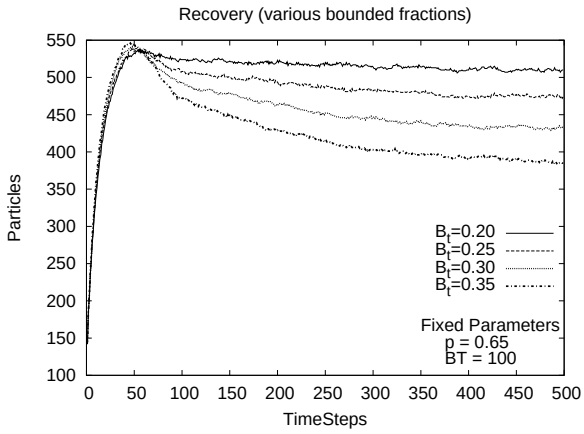
Third, we simulate three different types of FRAP experiments by varying the parameters of the probability of a jump towards the preferred direction p, the mean time of binding BT and the bounded fraction B_t. On the one hand we assumed that only the bounded fraction can be captured by the nucleus (Figure 3) and on the other hand all particles can be capured by the nucleus (Figure 4).



(a) different probabilities

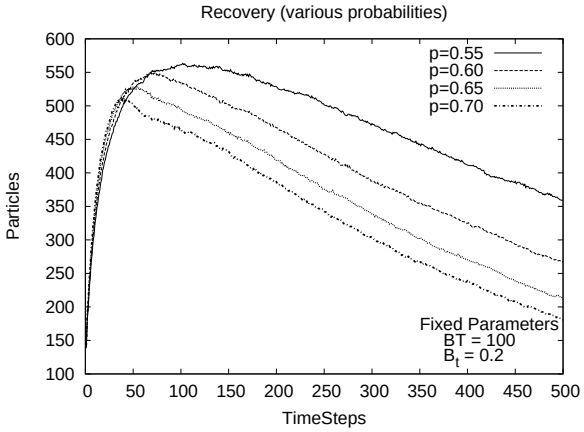


(b) different binding times

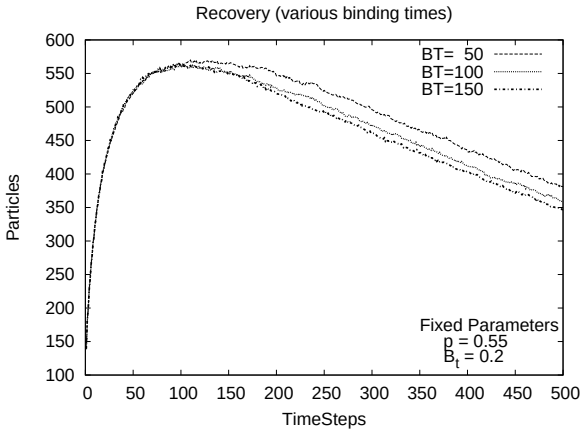


(c) different bounded fractions

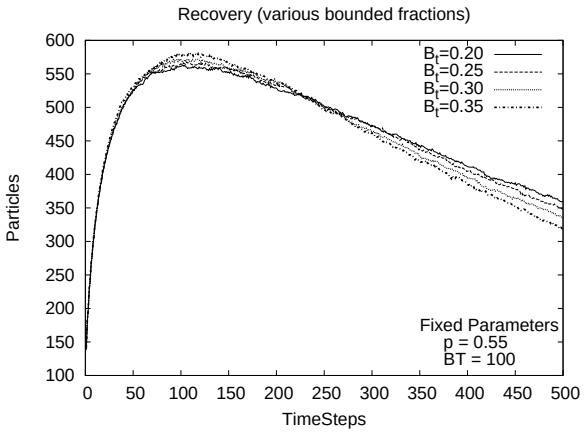
Figure 3. FRAP simulations (capture of bounded particles)



(a) different probabilities



(b) different binding times



(c) different bounded fractions

Figure 4. FRAP simulations (capture of all particles)

4. DISCUSSION AND FUTURE WORK

The influences on the recovery of FRAP experiments of directed particle movement and the possibility of particle capture in cell membranes are rarely described in the literature. A standard figure found in the literature is Figure 2(a) which correspondends in our simulation to unbounded particle movement. The recovery converges towards a non-zero value because no particle sink like capturing by the nucleus is modeled. In contrast, particle capture by the cell nucleus causes a zero limit in the recovery as displayed in Figure 2(b).
Further, the recovery is fastened and shifted to smaller values

1. by definition of a higher probability value for steps towards the sink (Figure 3(a), 4(a))
2. by definition of a higher mean binding time (Figure 3(b), 4(b))
3. by definition of a higher fraction of bounded particles (Figure 3(c), 4(c))

In the future we plan to derive an analytical solution for the FRAP recovery to change this qualitive conclusions into quantitive. This solution will allow to infer all parameters which describe the diffusion and binding processes from real FRAP data. Therefor different diffusion coefficients have to be modeled in a next step.

5. REFERENCES

[1] J. Braga, J. G. McNally, and M. Carmo-Fonseca, "A reaction-diffusion model to study rna motion by quantitative fluorescence recovery after photobleaching," *Biophysical Journal*, vol. 92, pp. 2694 – 2703, April 2007.

[2] R. D. Phair, S. A. Gorski, and T. Misteli, "Measurement of dynamic protein binding to chromatin in vivo, using photobleaching microscopy," *Methods in Enzymology*, vol. 375, pp. 393 – 414, 2004.

[3] C. P. Fall, E. S. Marland, J. J. Tyson, and J. M. Wagner, *Computational Cell Biology*, Springer Verlag, 2005.

[4] B. L. Sprague, R. L. Pego, D. A. Stavreva, and J. G. McNally, "Analysis of binding reaction by fluorescence recovery after photobleaching," *Biophysical Journal*, vol. 86, pp. 3473–3495, June 2004.

[5] B. L. Sprague and J. G. McNally, "Frap analysis of binding: proper and fitting," *Trends in Cell Biology*, vol. 15, no. 2, pp. 84–91, February 2005.

STOCHASTIC KINETIC SIMULATIONS OF ACTIVITY-DEPENDENT PLASTIC MODIFICATIONS IN NEURONS

Tiina Manninen[1,2] *and Marja-Leena Linne*[2]

[1] Department of Mathematics and [2] Department of Signal Processing
Tampere University of Technology, P.O. Box 553, FI-33101 Tampere, Finland
tiina.manninen@tut.fi, marja-leena.linne@tut.fi

ABSTRACT

Neuronal phosphorylation-dephosphorylation cycles have been shown to be important in the induction and maintenance of activity-dependent plastic modifications. In these cycles, protein kinases add phosphates to proteins and, on the other hand, phosphatases remove phosphates. Long-lasting, activity-dependent plastic modifications may provide the basis for cellular-level memory and learning. In this study, two different systems describing phosphorylation and dephosphorylation are studied and their behavior is simulated with deterministic and stochastic methods. The results of this study support the previously reported new principle for information storage in neurons where a single neuronal process controls both induction and maintenance of activity-dependent plastic modifications.

1. INTRODUCTION

Neurons respond to variations in their vicinity by modifying their synaptic and intrinsic membrane properties. Long-lasting and activity-dependent plastic modifications may provide the basis for cellular-level memory and learning. Neuronal phosphorylation-dephosphorylation cycles (protein kinase and phosphatase cycles) have been shown to be important in the induction and maintenance of the activity-dependent plastic modifications (see for a review [1, 2]). In this study, the behavior of two systems describing phosphorylation-dephosphorylation cycles [1, 2] are studied by the ordinary differential equation (ODE) model, the Gillespie stochastic simulation algorithm (SSA) [3, 4], and the stochastic differential equation (SDE) model, where stochasticity is incorporated into rate constants [5, 6, 7]. With these simple models that describe the minimal conditions required to generate plasticity, it is shown that phosphorylation-dephosphorylation cycles may play an important role in information storage.

2. SYSTEMS AND METHODS

In this study, two alternative systems based on the modulation of α-amino-3-hydroxy-5-methylisoxazole-4-propionic acid receptor (AMPA-R) activity through phosphorylation-dephosphorylation cycles are considered (System A and System B) and simulated with different methods.

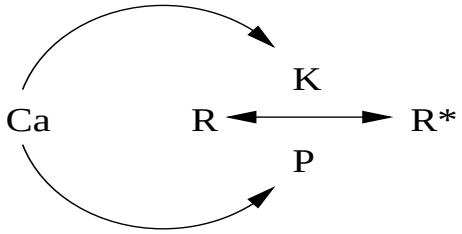


Figure 1. Graphical design model for System A, which was originally published in [2].

2.1. System A

A graphical design model for calcium (Ca^{2+})-controlled phosphorylation-dephosphorylation cycle of, for example, AMPA-R [2] is given in Figure 1. The reactions, reaction rates, rate constants, and initial values and description for each variable are given in Tables 1 – 2 (Model A). Rate constants are based on experimental data [2].

2.2. System B

A graphical design model for Ca^{2+}-controlled AMPA-R phosphorylation-dephosphorylation cycle [1] is given in Figure 2. The reactions, reaction rates, rate constants, and initial values and description for each variable are given in Tables 3 – 4 (Model B). Rate constants are based on experimental data or given a sophisticated guess [1]. Model B consists of six parts: 1) calmodulin (CaM, marked as S) activation, 2) Ca^{2+}/calmodulin-dependent protein kinase II (CaMKII, marked as K) activation, 3) calcineurin (CaN) or protein phosphatase 2B (PP2B) (marked as N) activation, 4) dopamine- and cyclic adenosine monophosphate (cAMP) -regulated phosphoprotein (DARPP-32) or I-1 inhibitor (I-1) (marked as D) activation, 5) protein phosphatase 1 (PP1, marked as P^*) inactivation, and 6) AMPA-R (marked as R) activation.

2.3. Deterministic differential equation model

The ODE model can be presented as

$$\mathrm{d}\mathbf{X} = \mathbf{S}\mathbf{v}\mathrm{d}t, \tag{1}$$

where $\mathbf{X}$ describes the variables (concentrations for chemical species) and $\mathbf{S}$ is the stoichiometric matrix. The function $\mathbf{v}$ describes the reaction rates and depends on the rate

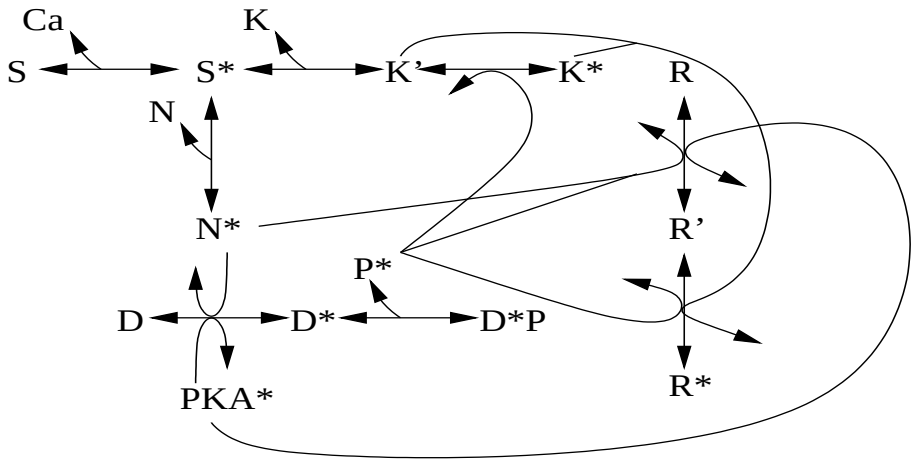


Figure 2. Graphical design model for System B, which was originally published in [1].

Table 1. Reversible reactions, reaction rates, and rate constants in Model A [2].

Reaction	Reaction rates		Rate constants	
$\mathrm{R} \underset{P}{\overset{K}{\rightleftharpoons}} \mathrm{R}^*$	$v_1 = K[\mathrm{R}]$	$v_2 = P[\mathrm{R}^*]$	$K = \frac{0.31[\mathrm{Ca}^{2+}]^4}{(6\times10^{-6})^4+[\mathrm{Ca}^{2+}]^4}\ \frac{1}{\mathrm{s}}$	$P = \frac{0.31[\mathrm{Ca}^{2+}]^4}{(3\times10^{-6})^4+[\mathrm{Ca}^{2+}]^4}\ \frac{1}{\mathrm{s}}$

Table 2. Descriptions and different initial concentrations for each variable in Model A [2]. The total number for $\mathrm{R} + \mathrm{R}^*$ is 100 in volume 6.5450×10^{-17} l. Different concentrations for Ca^{2+} are used in simulations.

Chemical species	Description	Initial concentrations
R	Dephosphorylated molecule	$[9, 7, 5, 3, 1] \times 2.5371 \times 10^{-7}$ M
R^*	Phosphorylated molecule	$[1, 3, 5, 7, 9] \times 2.5371 \times 10^{-7}$ M

Table 3. Reversible reactions, reaction rates, and rate constants in Model B [1].

Reaction	Reaction rates		Rate constants	
$\mathrm{Ca}^{2+} + \mathrm{S} \underset{k_2}{\overset{k_1}{\rightleftharpoons}} \mathrm{S}^*$	$v_1 = k_1[\mathrm{Ca}^{2+}][\mathrm{S}]$	$v_2 = k_2[\mathrm{S}^*]$	$k_1 = 10^6\ \frac{1}{\mathrm{Ms}}$	$k_2 = 7\ \frac{1}{\mathrm{s}}$
$\mathrm{S}^* + \mathrm{K} \underset{k_4}{\overset{k_3}{\rightleftharpoons}} \mathrm{K'}$	$v_3 = k_3[\mathrm{S}^*][\mathrm{K}]$	$v_4 = k_4[\mathrm{K'}]$	$k_3 = 1.5 \times 10^8\ \frac{1}{\mathrm{Ms}}$	$k_4 = 2.17\ \frac{1}{\mathrm{s}}$
$\mathrm{K'} \underset{k_6\times\mathrm{P}^*}{\overset{k_5}{\rightleftharpoons}} \mathrm{K}^*$	$v_5 = k_5[\mathrm{K'}]$	$v_6 = k_6[\mathrm{K}^*][\mathrm{P}^*]$	$k_5 = 0.50\ \frac{1}{\mathrm{s}}$	$k_6 = 3 \times 10^4\ \frac{1}{\mathrm{Ms}}$
$\mathrm{S}^* + \mathrm{N} \underset{k_8}{\overset{k_7}{\rightleftharpoons}} \mathrm{N}^*$	$v_7 = k_7[\mathrm{S}^*][\mathrm{N}]$	$v_8 = k_8[\mathrm{N}^*]$	$k_7 = 1.67 \times 10^9\ \frac{1}{\mathrm{Ms}}$	$k_8 = 10\ \frac{1}{\mathrm{s}}$
$\mathrm{D} \underset{k_{10}\times\mathrm{N}^*}{\overset{k_9\times\mathrm{PKA}^*}{\rightleftharpoons}} \mathrm{D}^*$	$v_9 = k_9[\mathrm{D}][\mathrm{PKA}^*]$	$v_{10} = k_{10}[\mathrm{D}^*][\mathrm{N}^*]$	$k_9[\mathrm{PKA}^*] = 0.002\ \frac{1}{\mathrm{s}}$	$k_{10} = 2 \times 10^9\ \frac{1}{\mathrm{Ms}}$
$\mathrm{D}^* + \mathrm{P}^* \underset{k_{12}}{\overset{k_{11}}{\rightleftharpoons}} \mathrm{D}^*\mathrm{P}$	$v_{11} = k_{11}[\mathrm{D}^*][\mathrm{P}^*]$	$v_{12} = k_{12}[\mathrm{D}^*\mathrm{P}]$	$k_{11} = 3 \times 10^7\ \frac{1}{\mathrm{Ms}}$	$k_{12} = 0.03\ \frac{1}{\mathrm{s}}$
$\mathrm{R} \underset{k_{14}(\mathrm{P}^*+\mathrm{N}^*)}{\overset{k_{13}\times\mathrm{PKA}^*}{\rightleftharpoons}} \mathrm{R'}$	$v_{13} = k_{13}[\mathrm{R}][\mathrm{PKA}^*]$	$v_{14} = k_{14}[\mathrm{R'}]([\mathrm{P}^*] + [\mathrm{N}^*])$	$k_{13}[\mathrm{PKA}^*] = 0.1\ \frac{1}{\mathrm{s}}$	$k_{14} = 10^6\ \frac{1}{\mathrm{Ms}}$
$\mathrm{R'} \underset{k_{16}\times\mathrm{P}^*}{\overset{k_{15}\times(\mathrm{K'}+\mathrm{K}^*)}{\rightleftharpoons}} \mathrm{R}^*$	$v_{15} = k_{15}[\mathrm{R'}]([\mathrm{K'}] + [\mathrm{K}^*])$	$v_{16} = k_{16}[\mathrm{R}^*][\mathrm{P}^*]$	$k_{15} = 10^7\ \frac{1}{\mathrm{Ms}}$	$k_{16} = 10^5\ \frac{1}{\mathrm{Ms}}$

constants, variables, and model inputs (concentrations for second messengers).

2.4. Itô stochastic differential equation model

In order to obtain SDE models, stochasticity is incorporated into ODE models [5, 6]. In this study, stochasticity is incorporated into rate constants. The Itô SDE is

$$\mathrm{d}\mathbf{X} = \mathbf{S}\mathbf{v}\mathrm{d}t + \mathbf{S}\mathbf{B}\mathbf{V}_{rc}\mathrm{d}\mathbf{W}, \tag{2}$$

where $\mathbf{V}_{rc}$ has reaction rates $\mathbf{v}$ without the rate constants (rc) as its diagonal elements and other elements are zero [5, 6]. The parameter values in the diagonal matrix $\mathbf{B}$ can be estimated using the responses of the SSA as the measurement data [8]. $\mathbf{W}$ is the Brownian motion.

2.5. Gillespie stochastic simulation algorithm

In the SSA, the length of the time step and which reaction happens during that time step are randomly selected based

Table 4. Descriptions and initial concentrations for each variable in Model B calculated by setting the concentration for Ca^{2+} to 10^{-7} M and using the steady state model given in [1]. The initial concentrations for R, R', and R^* are modified to be similar to Model A. The total number for R + R' + R^* is 100 in volume 6.5450×10^{-17} l. Different concentrations for Ca^{2+} are used in simulations. Rate constants times the concentration for protein kinase A (PKA) are given in Table 3.

Chemical species	Description	Initial concentrations
S	Calmodulin (CaM)	2.9997×10^{-5} M
S^*	Ca^{2+}/calmodulin complex (CaMCa4)	1.2494×10^{-12} M
K	Ca^{2+}/calmodulin-dependent protein kinase II (CaMKII)	4.9741×10^{-7} M
K'	Bound CaMKII	4.2957×10^{-11} M
K^*	Trapped CaMKII, where threonine Th^{286} is phosphorylated	2.5461×10^{-9} M
N	Protein phosphatase 2B (PP2B), also known as calcineurin (CaN)	9.9979×10^{-7} M
N^*	Active PP2B	2.0860×10^{-10} M
D	Dopamine- and cAMP-regulated phosphoprotein (DARPP-32) or I-1 inhibitor (I-1)	1.2751×10^{-6} M
D^*	Phosphorylated DARPP-32 or I-1	6.1126×10^{-9} M
P^*	Protein phosphatase 1 (PP1)	2.8119×10^{-7} M
D^*P	DARPP-32/PP1 complex or I-1/PP1 complex	1.7188×10^{-6} M
R	α-amino-3-hydroxy-5-methylisoxazole-4-propionic acid receptor (AMPA-R)	$[9, 7, 5, 3, 1] \times 2.5371 \times 10^{-7}$ M
R'	Naive AMPA-R, where only serine S^{845} is phosphorylated	$[0.5, 1.5, 2.5, 3.5, 4.5] \times 2.5371 \times 10^{-7}$ M
R^*	Completely phosphorylated AMPA-R	$[0.5, 1.5, 2.5, 3.5, 4.5] \times 2.5371 \times 10^{-7}$ M

on the propensity functions (reaction rates), and then the system and time are updated and simulation proceeds [3, 4]. In this study, the direct method is used [3].

3. RESULTS

With Model A and B, the effects of Ca^{2+} changes are studied. The concentration for Ca^{2+} is kept constant in each simulation. The numbers for R + R^* in Model A and R + R' + R^* in Model B are equal to 100 in a spine having volume $\frac{4}{3}\pi(0.25 \times 10^{-6})^3 \times 1000\,\mathrm{l} = 6.5450 \times 10^{-17}$ l, giving 2.5371×10^{-6} M. The active fraction (f) of the receptor is calculated by R^*/(R + R^*) in Model A and (R' + R^*)/(R + R' + R^*) in Model B.

Figure 3 shows that Model A and Model B behave differently when the concentration for Ca^{2+} and the initial active fraction are changed. Both deterministic and stochastic methods need to be used as small volume is involved [7]. Simulation results of the SDE are not shown since the SSA is computationally faster than the SDE. In Model A, the active fraction converges to a steady state after several months when the concentration for Ca^{2+} is 10^{-7} M. When the concentration for Ca^{2+} increases, the active fraction converges faster to a higher steady state value. In the case of 3×10^{-6} M and 6×10^{-6} M, it converges after 30 s and 10 s, respectively. Simulation results of Model B do not depend much on different concentrations for Ca^{2+}. The active fraction converges to a steady state after 150 s in all cases. The mean of values for steady state active fraction (f_∞) are given in Table 5. f_∞ is calculated for each deterministic simulation and then the mean of f_∞ is calculated for each subfigure in Figure 3. In Model A, the mean values change with different concentration for Ca^{2+}, but, on the other hand, in Model B, the values are about the same. The sample mean of 1000 simulations with the SSA and the SDE are the same but there are small differences in the beginning of the stochastic simulations compared to the deterministic results. Results mean that Model A is able to explain both induction and maintenance of plastic modifications, whereas Model B is only able to explain the induction [1, 2].

Table 5. Mean values for f_∞ calculated for deterministic simulations presented in Figure 3.

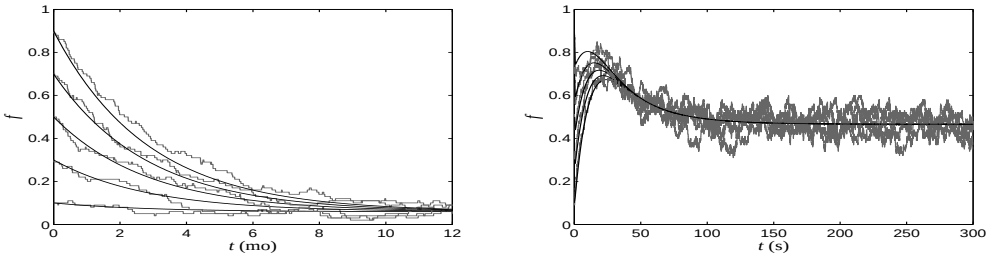
Ca^{2+}	Model A	Model B
10^{-7} M	0.0651	0.4659
3×10^{-6} M	0.1049	0.4898
6×10^{-6} M	0.3461	0.4266

4. CONCLUSIONS

Neuronal phosphorylation-dephosphorylation cycles have been shown to be important in the induction and maintenance of activity-dependent plastic modifications [1, 2]. In this study, the behavior of two systems for describing phosphorylation-dephosphorylation cycles are simulated with deterministic and stochastic methods for long periods of time. Even though Model A is very simple, it is able to explain both induction and maintenance of plastic modifications, whereas Model B is only able to explain the induction (as also noted in [1, 2]).

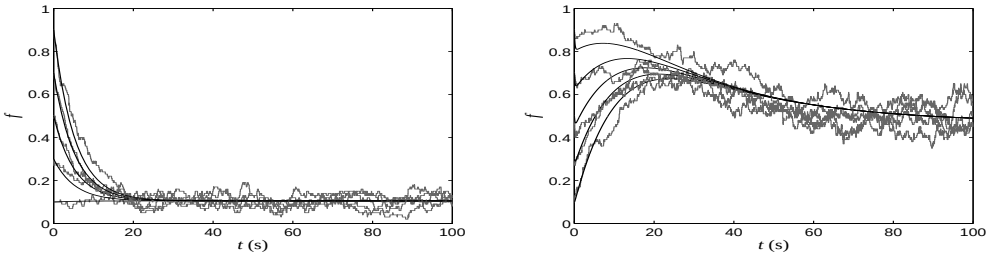
5. ACKNOWLEDGEMENTS

This work was supported by the Academy of Finland, project nos 213462 (Finnish Programme for Centres of Excellence in Research 2006-2011), 106030, and 124615, as well as Tampere University of Technology Graduate



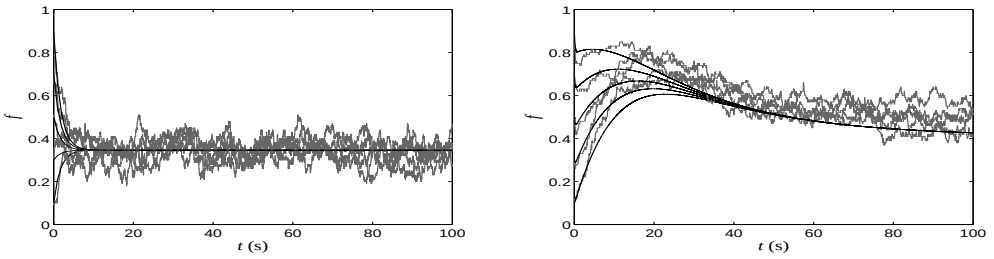
(a) Model A, Concentration for Ca^{2+} is 10^{-7} M.

(b) Model B, Concentration for Ca^{2+} is 10^{-7} M.



(c) Model A, Concentration for Ca^{2+} is 3×10^{-6} M.

(d) Model B, Concentration for Ca^{2+} is 3×10^{-6} M.



(e) Model A, Concentration for Ca^{2+} is 6×10^{-6} M.

(f) Model B, Concentration for Ca^{2+} is 6×10^{-6} M.

Figure 3. Active fractions from deterministic simulations (black) and stochastic simulations with the SSA (gray) of Model A (a, c, and e) and Model B (b, d, and f) using different initial active fractions and concentrations for Ca^{2+}.

School, the Jenny and Antti Wihuri Foundation, the Foundation of Technology, the Finnish Foundation for Economic and Technology Sciences – KAUTE, the Alfred Kordelin Foundation, and the Ulla Tuominen Foundation.

6. REFERENCES

[1] P. d'Alcantara, S. N. Schiffmann, and S. Swillens, "Bidirectional synaptic plasticity as a consequence of interdependent Ca^{2+}-controlled phosphorylation and dephosphorylation pathways," *Eur. J. Neurosci.*, vol. 17, pp. 2521–2528, 2003.

[2] B. Delord, H. Berry, E. Guigon, and S. Genet, "A new principle for information storage in an enzymatic pathway model," *PLoS Comput. Biol.*, vol. 3, no. 6, e124, 2007.

[3] D. T. Gillespie, "A general method for numerically simulating the stochastic time evolution of coupled chemical reactions," *J. Comput. Phys.*, vol. 22, no. 4, pp. 403–434, 1976.

[4] D. T. Gillespie, "Exact stochastic simulation of coupled chemical reactions," *J. Phys. Chem.*, vol. 81, no. 25, pp. 2340–2361, 1977.

[5] T. Manninen, M.-L. Linne, and K. Ruohonen, "A novel approach to model neuronal signal transduction using stochastic differential equations," *Neurocomputing*, vol. 69, no. 10-12, pp. 1066–1069, 2006.

[6] T. Manninen, M.-L. Linne, and K. Ruohonen, "Developing Itô stochastic differential equation models for neuronal signal transduction pathways," *Comput. Biol. Chem.*, vol. 30, no. 4, pp. 280–291, 2006.

[7] T. Manninen, *Stochastic methods for modeling intracellular signaling*, Ph.D. thesis, Department of Science and Engineering, Tampere University of Technology, Tampere, Finland, 2007.

[8] T. Manninen, A. Saarinen, A. Ylipää, M.-L. Linne, O. Yli-Harja, and K. Ruohonen, "Sequential Monte Carlo based maximum likelihood estimation for calcium binding reactions," in *Proc. of the 2nd Conference on Foundations of Systems Biology in Engineering (FOSBE 2007)*, Stuttgart, Germany, 2007, pp. 189–194.

EXPLORING PROTEIN INTERACTOME FEATURES

Elisabetta Marras and Enrico Capobianco

CRS4 Bioinformatics Laboratory,
Science and Technology Park of Sardinia,
Pula (CA) 09010, Italy
lisa@crs4.it, ecapob@crs4.it

ABSTRACT

The aim of our work is to explore topological features in terms of how sufficient they can be for the analysis of protein interactomes. One question concerning sufficiency is whether it might be possible to encapsulate in a few salient dimensions or components, or otherwise in a low dimensional bulk of interactions, all the relevant biological information and minimal residual noise. We show that this task can in principle be accomplished by statistical tools performing dimensionality reduction and feature selection. We are currently validating our results over the *Yeast* interactome as the reference model organism while we plan successive extensions to more complex contexts (*Homo Sapiens*).

1. INTRODUCTION

Protein-Protein Interaction Networks (PPIN) are becoming more and more important for understanding cellular functions and causal-effect relationships inside the cell. Interactomes are defined as the complete list of physical interactions mediated by all proteins of an organism [1].

A PPIN can be characterized through a wide variety of measures, each describing particular aspects according to different types of criteria, such as connectivity, distance, cliquishness, and so on. Dealing with biological networks which are simply dichotomous (binary interactions, depending on presence of absence of connectivity) generates several problems due to the impact of both coverage and accuracy of data.

As a result, the observed links are affected by both false positives (bad measurements) and false negatives (missing information). The strongest limitation is that *gold standard datasets* used to build knowledge from new experiments are also incomplete, and various biases exist as well, say towards proteins of high abundance or cellular localization.

Furthermore, there is not yet widespread consensus with regard to the *inference methods* for complex networks, as the highest parameter estimation and/or prediction accuracy are far from being achieved by one specific method. This is simply a consequence of the complexity and dimensionality of PPIN.

When weighted instead of unweighted interactomes are considered, problems arise in relation to what *statistical confidence* should be attached to the the assigned weights, reason why both scoring systems and randomization techniques are pursued.

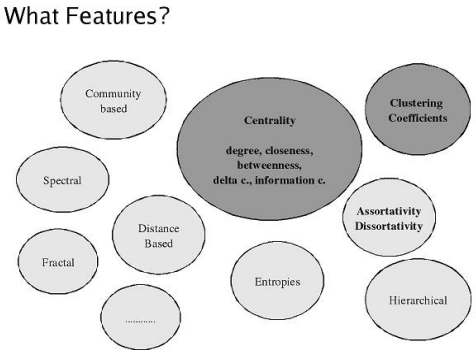


Figure 1. Classes of interactome features: examples elucidating aspects of the interactome topology.

A very common approach is the description of PPIN through *topological features* (*TOPfeatures*, to be distinguished from biological features used to explain interactions [2]). They are measures which characterize global (net-wise) and local (node- and edge-wise) aspects of interactions, such as connectivity, coreness, cohesiveness, centrality etc.

These features (listed in Figure 1) are usually studied in association with more general network properties. For instance, the usually observed scale-free property [3] implies that the connectivity level (or number of links per node) follow a power law (see Figure 2, where the pattern is confirmed at variable sub-sampling rate):

$$p(k) = ck^{-\gamma}, c > 0, \gamma > 0 \quad (1)$$

This fact in turn implies that there are some nodes in the network that are highly connected ("sticky" proteins) and most are poorly connected ("non-sticky" proteins) instead, thus there is a preferential attachment of new nodes to well-connected hubs.

We also look at features, and consider the statistics related to them, in particular correlation and distribution aspects (see for instance Figure 3). Each feature repre-

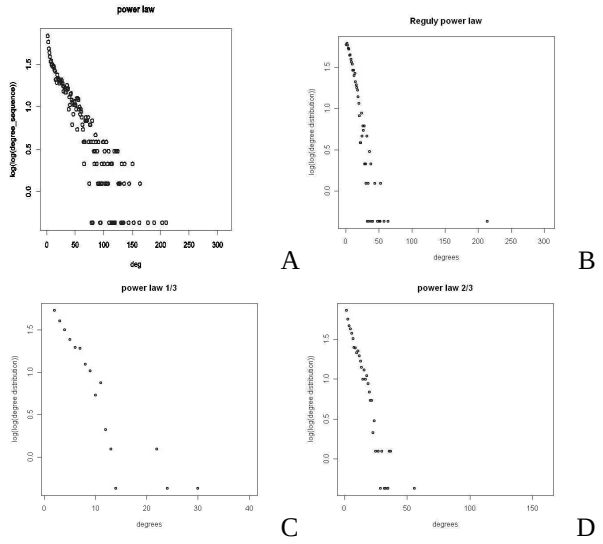


Figure 2. Power laws: A. Bader's [3]; B. Reguly's [4]. Sub-sampled power laws: C. (one third); D. (two thirds). The pattern is preserved under variable sub-sampling rate.

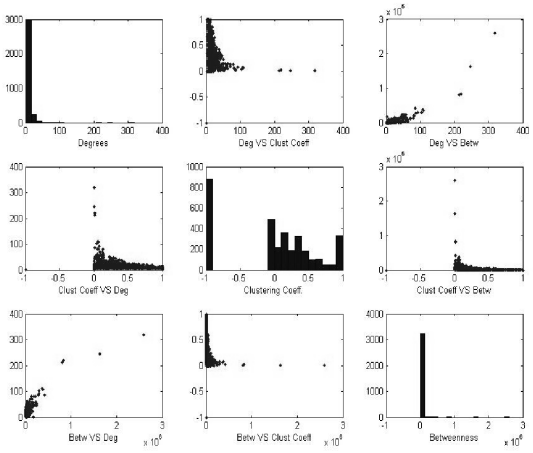


Figure 3. Distributional aspects: Bader's [3] features. Correlation patterns at variable degree appear between the chosen *TOPfeatures*.

sents a summary of relevant biological information in the network. We pick some node-wise valued *TOPfeatures* to support our analysis which are targeted to the search of some kind of sufficient information core within the interactome.

The datasets employed in this work are two popular sources. One, from Bader et al [3], is a combination of protein networks constructed from published Y2H and Co-IP data, thus resulting in a set of 5787 high-confidence high-throughput interactions. The other dataset is from Reguly et al [4] who built a database of genetic and protein interactions manually curated from 31793 abstracts and online publications, resulting in a set of 31311 interactions. We have considered only the literature-curated part (with 3289 proteins and 11314 interactions).

2. METHOD

A natural mapping from an interactome graph to a vector space (of large dimension, in this case) is provided by the adjacency matrix A ($A_{ij} = 1$, iff $\exists$ an edge between i and j nodes). With undirected matrices this is a symmetric matrix, and connectivity, clustering and centrality measures can be established by computing degree-degree distribution, clustering coefficient and betweenness, respectively.

Given $G = (V, E)$ network (no self-loops and multiple edges), with an $n-$set of V vertices, and an $m-$set of E edges, we have the following definitions:

- The node v degree distribution $D(v)$ is defined as the number of interacting partners of a protein (with no distinction between in- and out-degree due to the undirect nature of the graph), thus according to:

$$Deg(v) = k \tag{2}$$

where k is the number of nodes directly connected with v.

- Given a node v degree $d(v)$ defined as the number of nodes adjacent to v, the clustering coefficient [5] establishes the likeliness that a link between the nodes a and c exists when both a with b and b with c are linked. Taken from a function in the *R* package *graph*, for a node v in an undirected graph and with m adjacent nodes, when self loops are not to be included we have:

$$ClustCoef(v) = \frac{N}{(m * (m-1))} \tag{3}$$

where N is the number of edges between these nodes. If self loops are instead allowed, the clustering coefficent is $N/(m * m)$, where N is the number of edges between these nodes, including self loops.

- Betweenness [6, 7] is a centrality measure of a vertex, and is higher for some vertices depending on the fact that they are present on many shortest paths between other vertices. It is computed by a function from the *R* package *igraph* that defines the number of geodesics (shortest paths) going from an origin to a destination through a vertex relatively to the total number of geodesics observed between start and end node. For vertex v it holds that:

$$Betw(v) = \sum_{s \neq v \neq t \in V, s \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}} \tag{4}$$

for s and t belonging to V.

3. FEATURE IMPORTANCE

With two kinds of sets available to study interactome features, we have some heterogeneity due to the fact that from literature-curated data the sources of interactions can be more compared to high-throughput data. We thus focus more closely on the latter dataset.

Given the available *TOPfeatures* discussed in the literature, an important question is to what extent their redundancy can lead to complementary information [8].

It is likely that a small number of features is sufficient to describe the interaction map, each with a certain limited precision but relative importance. We aim to measure somehow this importance by further decomposing and denoising each feature, as shown in the next section.

This consideration holds without even looking at the specific characterization of them, say, gene-specific, more than sequence based or domain-domain oriented, just to mention some possibilities.

A point to stress is that each *TOPfeature* synthesizes underlying physical, genetic, evolutionary aspects which can be emphasized and in turn represent relatively strong or weak predictors of protein interactions.

Feature selection is thus required not to account for weak and noisy data characteristics, and decipher their interdependencies. While the former aspect is unavoidable due to experimental limitations in terms of accuracy and to real interactome coverage, the latter aspect can be investigated further.

4. FEATURE DECOMPOSITION

Principal Component Analysis (PCA) [9] is aimed to obtain the smallest possible signal subspace (S) from a noisy space (Y) where the data lie, $Y = S + \epsilon$. The data space rank, N is split into an $M-$component (signal part) and a residual $(N - M)-$component (noise part) depending on the relative magnitude of the singular values which are identified.

PCA determines an orthogonal projection which allows for decorrelation of the structure present in the original signal space. The idea behind the application of PCA to the *TOPfeatures* (see Figure 4) refers to possible changes in the correlation structure, in particular the interdependency links.

The addition of a *TOPfeature* (clustering coefficient) in the subplot B of Figure 4 indicates that when an extra component is extracted, there is a change between the correlation of the previously considered *TOPfeatures*. The extra component significantly affects the signal-to-noise ratio, quite likely, which makes hard to distinguish pure correlation changes from spurious correlation reduction.

We then look at Independent Component Analysis (ICA) [10, 11, 12, 13] in order to demix possible convoluted structure characterizing each *TOPfeatures*. Usually, ICA is used to extract independent and non-Gaussian signal sources from observed (noisy) mixtures with unknown mixing mechanism.

ICA is a very effective exploratory tool and compared to PCA, which is targeted to Gaussian data and linear dependence, it exploits high-order distributional information (from moments and cumulants).

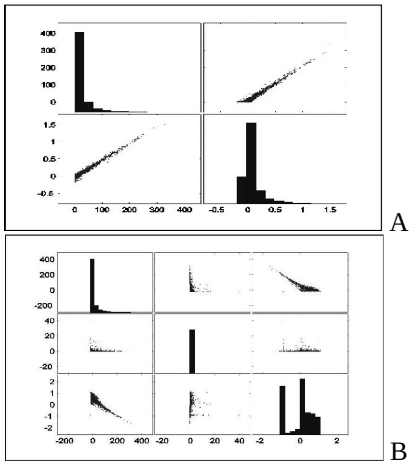


Figure 4. PCA decomposition with two Bader's [3] features (A), and with one additional feature (B). More components change the cross-*TOPfeature* relations.

In our applications we have observed that ICA suggests something about the inner structure of the signals at hand, and reveal the distributional properties of its informative bulk. It appears from Figure 5 that a strong non-Gaussian characterization is underlying these signals, which together with possible statistical independence of the extracted components are the strengths of ICA.

5. CONCLUSIONS

We have provided examples of features that can analyzed statistically because quantified at a local level in the network, i.e. node-wise.

Their correlation and distributional aspects suggest that interdependencies are quite strong among the *TOPfeatures* here analyzed.

Further decomposition through PCA and ICA deserves care: PCA shows that decorrelation cannot suffice to disentangle the dependence structure among *TOPfeatures*, while ICA reveals a strong non-Gaussian characterization for them.

Our final remarks, consequently, are that it might be better to rely on a small rather than a big (or redundant) set of *TOPfeatures*, because in passing from the graph (i.e. a dichotomous space) to the features (i.e. a multivariate space), the task of dealing with correlation remains hard even after dimensionality reduction and denoising.

Next, we plan to perform shrinkage estimation and calibration in null and/or complementary graph creation.

6. ACKNOWLEDGMENTS

Support from Sardegna Ricerche and CRS4 is gratefully credited and acknowledged.

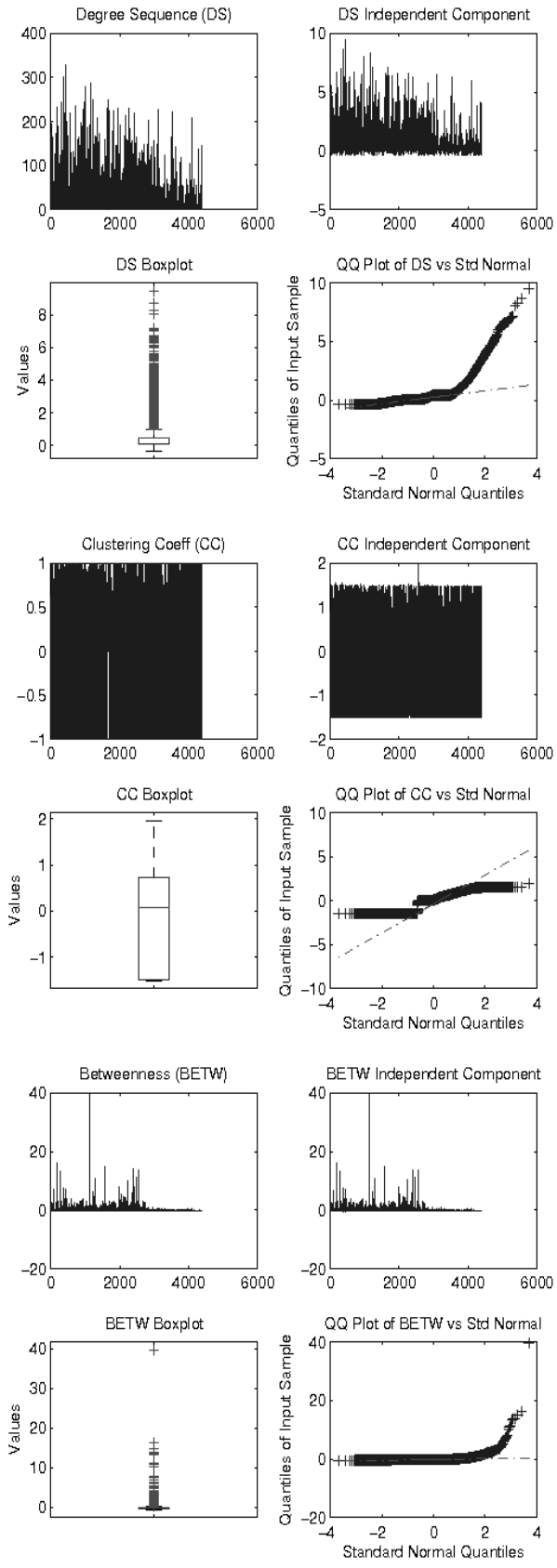


Figure 5. ICA demixing of Bader's [3] degree (top four plots), clustering coefficient (mid four plots), and betweenness (bottom four plots). Diagnostic plots to emphasize outlying distributional aspects.

7. REFERENCES

[1] M. Vidal, "Interactome modeling," *FEBS Letters*, vol. 579, pp. 1834 – 1838, 2005.

[2] L. Lu, Y. Xia, A. Paccanaro, and M. Gerstein, "Assessing the limits of genomic data integration for predicting protein networks," *Genome Research*, vol. 15, pp. 945 – 953, 2005.

[3] A. Barabasi and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, pp. 509 – 512, 1999.

[4] T. Reguly, A. Breitkreutz, L. Boucher, B. Breitkreutz, G. Hon, C. Myers, A. Parsons, H. Friesen, R. Oughtred, A. Tong, C. Stark, Y. Ho, D. Botstein, B. Andrews, C. Boone, O. Troyanskya, T. Ideker, K. Dolinski, N. Batada, , and M. Tyers, "Comprehensive curation and analysis of global interaction networks in saccharomyces cerevisiae," *Journal of Biology*, vol. 5:11, 2006.

[5] D. Watts and S. Strogatz, "Collective dynamics of "small-world" networks," *Nature*, vol. 393, pp. 440 – 442, 1998.

[6] U. Brandes, "A faster algorithm for betweenness centrality," *Journal of Mathematical Sociology*, vol. 25(2), pp. 163 – 177, 2001.

[7] L. C. Freeman, "A set of measures of centrality based on betweenness," *Proteins: Structure, Fuction, and Bioinformatics*, vol. 40, pp. 35 – 41, 1977.

[8] Z. B.-J. Y. Qi and J. Klein-Seetharaman, "Evaluation of different biological data and computational classification methods for use in protein interaction prediction," *Proteins: Structure, Fuction, and Bioinformatics*, vol. 63, pp. 490 – 500, 2006.

[9] I. Jolliffe, *Principal Component Analysis*, NY: Springer, New York, 1986.

[10] J. Cardoso, "Source separation using higher order moments," in *Proc. ICASSP*, 1989, pp. 2109–2112.

[11] P. Comon, "Independent component analysis - a new concept?," *Sig. Proces.*, vol. 36(3), pp. 287–314, 1994.

[12] J. Cardoso and A. Souloumiac, "Blind beamforming for non-gaussian signals," in *IEE Proc. F.*, 1993, pp. 771–774.

[13] A. Hyvarinen and E. Oja, "A fast fixed-point algorithm for independent component analysis," *Neural Comput.*, vol. 9(7), pp. 1483–1492, 1997.

[14] J. Bader, A. Chaudhuri, J. Rothberg, and J. Chant, "Gaining confidence in high-throuhput protein interaction networks," *Nature Biotechnology*, vol. 22(1), pp. 78 – 85, 2004.

FINDING SNP INTERACTIONS

Tina Mueller[1,2]*, Holger Schwender*[1,2] *and Katja Ickstadt*[1,2]

[1]Fakultät Statistik, Technische Universität Dortmund,
[2]Sonderforschungsbereich 475, Technische Universität Dortmund,
Vogelpothsweg 87, D-44227 Dortmund, Germany.
tmueller@statistik.uni-dortmund.de, holger.schwender@udo.edu

ABSTRACT

Genetic association studies investigating the relationship between complex diseases and Single Nucleotide Polymorphisms (SNPs) have become popular recently, as both costs and time of genotyping have decreased dramatically. However, reliable tools of extracting relevant results from them are yet to be established, as the studies usually contain more variables than observations and only small signals. Moreover, the interest focuses not on the identification of single SNPs, but (high-order) SNP interactions.
To face this challenge, we use adaptations of frequent itemsets and association rules as well as tree-based methods to detect such SNP interactions and to classify the observations as cases or controls. These methods contain great potential for the analysis of SNP data as our applications to both simulated association data and real-world whole-genome data show.

1. INTRODUCTION

The etiology of complex diseases like cancer is still not fully understood. Yet it is evident that a person's susceptibility to cancer is due to a combination of genetic predisposition and environmental influences [1]. A major problem in the analysis of genetic SNP data is the abundance of possible factors to be considered. Association studies contain between 50 specifically selected SNPs in candidate SNP studies and up to hundreds of thousands SNPs in genome-wide studies. As the majority of the observed study variables does not contribute to the disease risk, and study size is usually small [2], we have to try to detect low signals caused possibly only by interactions of variables rather than by single variables in a lot of noise. Standard statistical methods fail to solve this problem.
We will show that analysis tools based on frequent itemsets and association rules are suitable to fulfill the difficult task as good as possible.
By assessing the misclassification rate, we compared our adapted methods with each other and to logic regression and further tree-based discrimination methods.
The data we analyse will be described in the next section, followed by Section 3 which presents the methods used. The results of the application to the simulated data are given in Section 4, while the key issues, observed drawbacks and future progress are discussed in the last section.

2. DATA

A SNP refers to a single base exchange at a specific locus on the genome that is present in at least 1 % of the population. There are three possible genotypes at each loci: the *homozygous reference* genotype (if both chromosomes show the more frequent base), the *heterozygous* genotype (if one chromosome shows the more frequent and the other the less frequent genotype) and the *homozygous variant* (if both chromosomes show the less frequent variant).
Interacting SNPs are assumed to influence the risk of developing diseases. Thus, they can help to identify high risk groups of patients and to classify new observations into cases and controls.
To compare the different classification approaches, we analyse different real and simulated data sets.

1. **HapMap.** We use a subset of the HapMap data [3] comprising 45 unrelated Han Chinese and 45 unrelated Japanese. Ethnicity is used as a class label in this case. The feature subset consists of 157 SNP variables which express all three genotypes, have a minor allele frequency greater than 0.1 and are preselected using the Significance Analysis of Microarrays [4] adapted to categorical data [5].

2. **GENICA.** The GENICA study on genetic and environmental interactions and sporadic breast cancer [6] was designed as an age-matched population-based candidate SNP association study. The specific data used in this article is built up by 63 SNP variables and 1191 observations (561 cases and 630 controls). The few missing values in this data set are replaced SNP-wise by random draws from the marginal distribution of the respective SNP.

3. **SNaP.** The simulated data are created using the software package SNaP [7]. For four different scenarios 1-4 (status "case" is caused either by one two-way interaction, or by one of two two-way interactions, or by one of three two-way interactions, or by one of two three-way interactions), ten data sets are generated, each of which contains 1000 observations (divided into 500 cases and 500 controls) with categorical values for 40 SNP variables. We specify

Table 1. Penetrances of the simulation study

two-way interactions		three-way interactions	
P(case | A = a, B = b)	(A = a) * (B = b)	P(case | A = a, B = b, C = c)	(A = a) * (B = b) * (C = c)
1	$2*2$ $2*1$	1	$2*2*2$ $2*2*1$
0.6	$2*0$ $1*1$	0.6	$2*2*0$ $2*1*1$ $1*1*1$
0.3	$1*0$	0.3	$2*1*0$ $1*1*0$
0	$0*0$	0	$2*0*0$ $1*0*0$ $0*0*0$

different minor allele frequencies $f \in [0.1, 0.3]$ for the causative SNPs. For each interaction, the chosen penetrance values can be found in Table 1. Note that they are symmetric, i.e. (A = 0) * (B = 1) is equivalent to (A = 1) * (B = 0). Additionally, nine categorical epidemiological variables with different numbers of levels are simulated. They, however, do not have an impact on the disease status. For classification purpose, each data set is used once as training data and once as test data.

3. METHODS

To mine frequent itemsets and association rules, the statistical software `R 2.5.1` [8] and the package `arules 0.6-3` [9] are used, where `arules` is based on the well-known apriori algorithm introduced by [10].
Background on the algorithm and its usage can be found in the articles cited above. In this paper, we employ frequent itemsets and association rules for classification purposes in genetic association studies.
Frequent itemsets and association rules have been applied successfully in various ways to several kinds of genetic data (e.g., [11], [12], [13]), but to our knowledge, the methods presented in this section have not been used in this specific context of disease status classification based on SNP data.
Frequent itemsets can be employed to define subgroups of observations by allocating a person to the group corresponding to the first itemset of an ordered list that is contained in the person's transaction. A separate classification model can be built in each of the created subgroups. A model (in our comparisons: CART) is built on the training data within each group and validated using the respective test data. We call this approach localCART.
Furthermore, frequent itemsets can be used for feature construction (FC). The frequent itemsets of the training set serve as new input variables, i.e. the new training and test sets consist of a binary vector for each observation indicating if the respective frequent itemset is contained in the observation's transaction or not. Subsequently, a classification based on these new binary features can be carried out.
A possible extension to the second kind of supportive tools, association rules, is to employ them in a classification framework (associative classification, [14]). All discovered rules are only allowed to contain one element in their consequent which has to be one of the class labels.
The resulting rule set can be employed in different ways for classification purposes. We take a voting approach (AC Vote): All rules applicable to a new observation contribute to the labelling of the respective observation. According to their vote, the new observation is assigned to the voted class [15].
Currently, the status "case" is chosen if a fraction of at least 0.1 of the votes suggests it. Besides these methods based on frequent itemsets and association rules, a tree-based discrimination and regression procedure called logic regression (LogReg)[16] is considered. Logic regression has been especially designed for SNP data and therefore shows good results when applied to SNP data [5]. Moreover, we also employ Random Forests (RF) [17], Bagging [18] and CART [19] in the comparisons presented in this paper.

4. ANALYSIS AND COMPARISON

For each scenario, each of the SNaP data sets is used once as training and once as test set. The misclassification rates (MCRs) shown in Figure 1 are averaged over all data sets from each scenario. For the HapMap and the GENICA data set, crossvalidation (9fold and 10fold, respectively) is used to estimate the misclassification rate. The different parameter specifications for the applications of the apriori algorithm are summarized in Table 2.
In the simulated settings, the MCRs for the tree-based methods, AC Vote and localCART increase with more two-way interactions. The best MCRs rise from 0.232 in Scenario 1 to 0.327 in Scenario 2 and 0.452 in Scenario 3 (cf. Figure 1).
FC shows an irregular, but constantly unsatisfying behaviour across all scenarios. The corresponding MCRs lie between 0.432 and 0.494, the latter meaning that tossing a fair coin to assign a class label to a new observation yields almost as good results as FC.

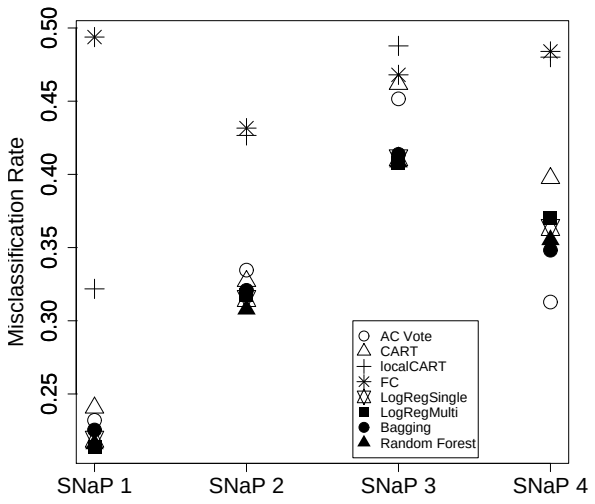


Figure 1. Misclassification rates for the four simulation scenarios using different classification approaches

For two-way interactions, the tree-based methods besides CART dominate the itemset-based methods. But in Scenario 4 (causative three-way interactions), AC Vote clearly outperforms the other methods and gives even better MCRs than it does for two or three causative two-way interactions.

localCART, though not suitable for the simulated case yielding MCRs about nearly 0.500, gives the second lowest MCR for the GENICA data (cf. Table 3). The best MCR (0.405) is achieved by logic regression. Again, the MCR of FC is highest with 0.498.

The HapMap data could be discriminated best. The MCRs lie between 0.011 and 0.356, with ACVote and Random Forest giving the lowest MCRs.

5. DISCUSSION

SNP data sets confront the analyst with several challenges: They can be very large (or at least consist of more variables than observations), the signal within the data can be quite small and is presumably caused by interacting factors rather than by single variables. With genome-wide studies in view, we investigate possible classification adaptations of frequent itemsets and association rules which are suitable for large data sets. In particular, we compared an approach based on local subgroups (localCART), a feature construction classification, an associative classification variant (AC Vote) with each other and with standard tree-based methods (CART, Bagging, Random Forest) as well as logic regression.

A general statement of which methods performs best was not expected and cannot be made. If the tree-based methods and especially logic regression are viewed as the current standard, we found that AC Vote is usually close or in one setting even better than the standard methods in terms of a smaller misclassification rate. The other two suggested methods did not show a satisfying behaviour, even though localCart performed second best on the GENICA data.

Besides the overall problem of a low signal and high noise within the data, all methods applied suffer from individual weaknesses. There is the miscellaneous group created by localCART which contains objects that do not share similar characteristics (as is the idea of localCART) with each other. This is due to the fact that all observations that did not fit into a group of a suitable size were merged into this class. If this group is neglected during the analysis, misclassification rates improve in each scenario. On the other hand, identifying objects which cannot be grouped in a sensible way with other observations can be important for application as well.

The feature construction approach usually yields more new variables than old ones. The redundancy of features reflects the initial problem in a different way. Not only do many of the variables not contribute to the outcome, but some of the mined frequent itemsets contain redundant information.

The voting scheme of AC Vote has to reflect the data characteristics as well as possible. The current scheme is still quite simple.

The major advantage of the methods based on frequent itemsets and association rules is their potential for a better performance on this special kind of genetic data. Further adaptations and different sorts of fine tuning will make them more suitable. In particular, we want to fit different models in the local subgroups, also answering the special characteristics in the miscellaneous group. Furthermore, a

Table 2. Parameters for the apriori algorithm for different data sets: Support and confidence for the cases (controls)

data	support	confidence
method	$1/2/3$ items	
HapMap		
AC Vote	0.07(0.05)	0.8
localCART/FC	$0.8/0.8/0.8$	–
GENICA		
AC Vote	0.065 (0.05)	0.65
localCART/FC	$0.8/0.75/0.75$	–
SNaP		
AC Vote	0.07 (0.05)	0.7 (0.065)
localCART/FC	$0.8/0.75/0.7$	–

Table 3. The misclassification rates for the real-world data sets HapMap and GENICA

method	HapMap	GENICA
ACVote	0.011	0.432
localCART	0.356	0.417
FC	0.244	0.498
LogReg	0.144	0.405
CART	0.356	0.437
Bagging	0.022	0.453
RF	0.011	0.450

variable selection on the constructed features might eliminate useless information and help to decrease the misclassification rate. By adjusting the voting scheme in AC Vote [20], the classification can be improved.
Therefore, we are convinced that the adjusted versions of the methods will help classify cases and controls more accurately in near future.

Acknowledgements

Financial support of the Deutsche Forschungsgemeinschaft (SFB 475, "Reduction of complexity in multivariate data structures") is gratefully acknowledged. Thanks to the GENICA Network for the supply of the data.

6. REFERENCES

[1] F. Perera, "Molecular epidemiology: Insights into cancer susceptibility, risk assessment, and prevention," *Journal of the National Cancer Institute*, vol. 88, pp. 496 – 509, Apr. 1996.

[2] W. Wang, B. Barratt, D. Clayton, and J. Todd, "Genome-wide association studies: Theoretical and practical concerns," *Nature Reviews Genetics*, vol. 6, pp. 109–118, Feb. 2005.

[3] The International HapMap Consortium, "The International HapMap Project," *Nature*, vol. 426, pp. 789–796, Dec. 2003.

[4] V. Tusher, R. Tibshirani, and G. Chu, "Significance analysis of microarrays applied to the ionizing radiation response," *Proceedings of the National Academy of Sciences*, vol. 98, pp. 5116 – 5124, 2001.

[5] H. Schwender, "Statistical analysis of genotype and gene expression data," *Ph.D. Thesis, University of Dortmund, Germany*, 2007.

[6] C. Justenhoven, U. Hamann, B. Pesch, V. Harth, S. Rabstein, C. Baisch, C. Vollmert, T. Illig, Y. Ko, T. Brüning, and H. Brauch, "ERCC2 genotypes and a corresponding haplotype are linked with breast cancer risk in a German population," *Cancer Epidemiol Biomarker Prev*, vol. 13, pp. 2059 – 2064, 2004.

[7] M. Nothnagel, "Simulation of ld block-structured snp haplotype data and its use for the analysis of case-control data by supervised learning methods," *American Journal of Human Genetics*, vol. 71, pp. A2363, Oct. 2002.

[8] R Development Core Team, "R: A language and environment for statistical computing," *R Foundation for Statistical Computing, Vienna, Austria*, URL http://www.R-project.org 2004.

[9] M. Hahsler, B. Grun, and K. Hornik, "arules - a computational environment for mining association rules and frequent item sets," *Journal of Statistical Software*, vol. 14, pp. 1 – 25, Oct. 2005.

[10] R. Agrawal, T. Imielinski, and A. Swami, "Mining association rules between sets of items in large databases," in *Proc. ACM SIGMOD*, USA, Columbia, Washington, May 1993, pp. 207–216.

[11] C. Shoemaker and C. Ruiz, "Association rule mining algorithms for set-valued data," in *IDEAL*, China, Hong Kong, Mar. 2003, pp. 669–676.

[12] D. Wimalasuriya, S. Ramachandran, and D. Dou, "Clustering zebrafish genes based on frequent-itemsets and frequency levels," in *Advances in Knowledge Discovery and Data Mining 11th Pacific-Asia Conference (PAKDD)*, China, Nanjing, May 2007, pp. 912–920.

[13] J. de Graaf, R. de Menezes, J. Boer, and W. Kosters, "Frequent itemsets for genomic profiling," in *Proc. CompLife*, Germany, Konstanz, Sep. 2005, pp. 104 – 116.

[14] B. Liu, W. Hsu, and Y. Ma, "Integrating classification and association rule mining," in *Proc. 4th KDD*, USA, New York, New York City, Aug. 1998, pp. 80 – 86.

[15] E. Baralis and P. Garza, "Majority classification by means of association rules," in *Proc. PKDD*, Croatia, Cavtat-Dubrovnik, Sep. 2003, pp. 35–46.

[16] I. Ruczinski, C. Kooperberg, and M. LeBlanc, "Logic regression," *Jorunal of Computational and Graphical Statistics*, vol. 12, pp. 475 – 511, 2003.

[17] L. Breiman, "Random forests," *Machine Learning*, vol. 45, pp. 5 – 32, Nov. 2001.

[18] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 26, pp. 123 – 140, Aug. 1996.

[19] L. Breiman, J. Friedman, R. Olshen, and C. Stone, *Classification and Regression Trees*, Chapman and Hall, New York, 1984.

[20] J.Ting, T.-C. Fu, and F.-L. Chung, "Mining of stock data: Intra- and inter-stock pattern associative classification," in *Proc. DMIN 2006*, USA, Nevada, Las Vegas, Jun. 2006, pp. 30–36.

CALCIUM CHANGES INDUCED BY AMYLOID-β-NEUROTRANSMITTER INTERACTIONS IN ASTROCYTES: MODEL FORMATION

Eeva Mäkiraatikka[1], *Amit K. Nahata*[2], *Tuula O. Jalonen*[3], *and Marja-Leena Linne*[1]

[1]Department of Signal Processing, Tampere University of Technology,
P.O. Box 553, FI-33101 Tampere, Finland
[2]Kidney Care Center at DCI, Washington, PA, USA
[3] Department of Biological and Environmental Science
University of Jyväskylä, FI-40014 Jyväskylä, Finland
eeva.makiraatikka@tut.fi, marja-leena.linne@tut.fi

ABSTRACT

The goal of this work is to present a computational framework on how to combine experimentally obtained data from calcium oscillations with modeling studies to understand the mechanisms leading to complex interactions between amyloid-β peptide and neurotransmitters in glial cells. Experimental work has provided evidence that a failure in the proteolytic processing of the amyloid precursor protein results in increased production of amyloid-β peptide. Aggregates of amyloid-β have been shown to destabilize the cellular calcium homeostasis in brain, a phenomenon associated with Alzheimer's disease. In normal cellular microenvironment in central nervous system, the level of intracellular calcium is transiently increased by neurotransmitters, such as serotonin. By adding serotonin and amyloid-β together, the enhancing effect on the intracellular calcium levels is multiplied. The model discussed here for the amyloid-β-neurotransmitter interactions in cortical astrocytes describes this synergistic effect of amyloid-β and serotonin. The complete computational model can be used to study pathological phenomena associated with Alzheimer's disease.

1. INTRODUCTION

Alzheimer's disease (AD) is a progressive and irreversible neurodegenerative disorder that leads to cognitive impairment and emotional disturbances. Symptoms result from the degeneration of brain tissue, seen as a shrinkage of brain regions, such as temporal and frontal lobes, which are involved in cognitive processes, learning and memory formation [1]. Pathological changes in a living patient's brain can be detected by using MRI and PET imaging techniques. In addition to brain shrinkage, an AD patient suffers from so called "plaques" and "neurofibrillary tangles", which are thought to hinder the transmission of nerve impulses. A definite diagnosis of AD can be confirmed by pathological studies, *post mortem*.

1.1. APP processing

Amyloid precursor protein (APP) is a type I transmembrane glycoprotein which consists of 695-770 amino acids, thus the C-terminal is in the cell cytosol. Secretases (α-, β-, and γ-secretase) are enzymes, which catalyze the proteolytic processing of APP. Each of them has its own characteristic cleavage site, and thus generates different peptide fragments (Fig. 1).

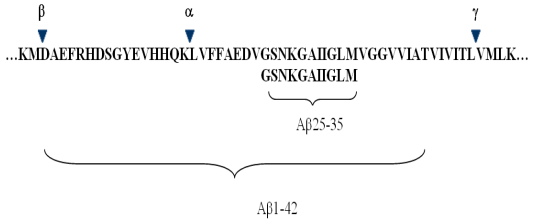


Figure 1. A section of amyloid precursor protein sequence listed with single-letter-abbreviations of amino acids. The triangles represent secretase (α, β, γ) cleavage sites.

Figure 2 shows the different possibilities of APP processing. Some of the formed intra- and extracellular fragments are considered to be neurotoxic. The non-neurotoxic fragments are produced with α-secretase. The extracellular fragment is a soluble peptide (sAPPα) and the intracellular C-terminal fragment consists of 83 amino acids, and is called C83. This above described APP processing represents the APP processing found in normal healthy brain.

In contrast, β-secretase cuts the transmembrane protein APP in such a way that a soluble N-terminal fragment (sAPPβ) is released outside the cell. The rest of the precursor protein stays attached to the membrane, and the γ-secretase can then process the remaining transmembrane peptide producing an intracellular, C-terminal fragment (C99). This fragment can be transported into the cell nucleus, where it participates to gene expression and, e.g. promotes apoptosis. Together with the intracellular C99, an extracellular amyloid-β (Aβ) peptide, is also formed.

Aβ consists of 39-42 amino acids. Based on the classification of amino acids by Branden and Tooze [2], 25 amino acids out of the 42 have hydrophobic side chains in Aβ42 (glysine is classified to be hydrophobic). Therefore, Aβ42 tends to aggregate easier than the shorter Aβ

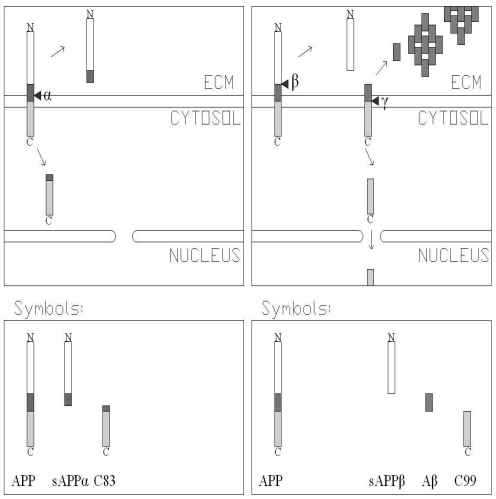


Figure 2. APP processing by α-, β-, and γ-secretases. The figure is a modification of the figure in [1]. Amyloid precursor protein (APP), the cleavage region for a secretase (◀), soluble extracellular fragment cleaved by α-secretase (sAPPα), intracellular fragment cleaved by α-secretase (C83), N-terminal fragment cleaved by β-secretase (sAPPβ), C-terminal fragment cleaved by γ-secretase (C99), and extracellular amyloid β peptide cleaved by γ-secretase (Aβ).

fragments. Neuritic plaques found in AD patients consist mostly on Aβ42. Both fragments, the 42 amino acids long (Aβ42) and the shorter 11 amino acids long synthetic derivative (Aβ25-35), are widely used in Alzheimer's disease research. Here, the shorter Aβ25-35 fragment is used together with a neurotransmitter, serotonin. The sequences and surfaces of these two specific fragments are shown in Figures 1 and 3. The longer the peptide is, the more the peptide gets aggregated, and thus working with Aβ42 is more difficult than with Aβ25-35. As is Aβ42, also Aβ25-35 is thought to be neurotoxic [3]. Details about the structures of Aβ25-35 and Aβ42 can be found from [3, 4], respectively.

The Aβ42 structure consists of two α-helices and a β-turn between them. Figure 3 shows the molecular surfaces for Aβ42 and Aβ25-35 (PDB IDs 1IYT and 1QWP, respectively). The hypothesis is that the Aβ's neurotoxicity is related to different peptide-cell membrane interactions and destabilization processes, culminating in membrane pore formation and membrane/cell disruption. According to this hypothesis, an α-helical peptide induces the formation of membrane channels, permitting neuronal death inducing reactants to penetrate [3].

1.2. Signaling pathways associated with Alzheimer's disease

Pathological studies of AD patients' brain reveal plaques and neurofibrillary tangles with Aβ and hyperfosforylated τ-protein, respectively. There is a consensus about the central role of Aβ in AD. However, there is only a weak correlation between fibrillary amyloid load and measures

Figure 3. Tertiary structures of Aβ42 (on the left-hand side) and Aβ25-35 (on the right-hand side). The light gray areas are hydrophobic, and black hydrophilic. The picture is drawn using Swiss-PdbViewer 3.7 [5].

of neurological dysfunction [3]. A hypothesis for AD formation and generation is as follows: **(1)** Abnormal amyloid precursor protein (APP) processing in the plasma membrane results in Aβ formation. **(2)** Extracellular Aβ aggregation induces oxidative stress in surrounding cells. **(3)** Oxidative stress causes alterations in the plasma membrane. These alterations affect the cell membrane potential and intake of ions. For example, the intracellular calcium ion concentration ([Ca^{2+}]) will rise. **(4)** Ca^{2+} functions as an intracellular second messenger, activating e.g. protein kinases, which are phosphorylation catalyzing enzymes. **(5)** Due to activation of protein kinases, τ-protein gets hyperphosphorylated and begins to form neurofibrillary tangles, common to AD patients. **(6)** Both Aβ aggregates and neurofibrillary tangles hinder the normal transfer of neuroimpulses. **(7)** Synaptic dysfunction causes degeneration in central nervous system (CNS). Cells initiate an apoptosis cascade and finally die.

Whatever the initiator for Aβ formation and aggregation is, the ionic equilibrium in the CNS microenvironment gets disturbed. Also the interactions of various neurotransmitters and cellular signaling pathways are disturbed, as has been suggested in studies on cellular calcium dysregulation [6, 7, 8, 9, 10].

In the following, both the experimental measurements and the computational framework for the model are discussed. The model will focus on the synergistic effects of Aβ and the neurotransmitter serotonin on the intracellular [Ca^{2+}] in rat cortical astrocytes. The aim is to formulate a computational model which can reproduce the experimental data and can be further developed to study and explain also the pathological phenomena associated with human AD.

2. METHODS

The experiments presented in this study use rat cortical astrocytes as a model organism. The main idea is to follow the changes in the level of Ca^{2+} concentration following additions of Aβ25-35 and serotonin.

2.1. Cell culture

Astrocytes are distributed throughout the CNS and constitute 20-50 % of the volume of most brain areas. They have unique cytological and immunological properties which make them easy to identify. One of the functions of astrocytes is neurotransmitter release and uptake in a synaptic cleft. Thus, astrocytes have a significant role in the function of the brain. [11]

Primary astrocyte cultures were prepared from newborn Sprague-Dawley rat pups, and grown on coverslips in culture dishes and kept at 37°C in an air-ventilated humified incubator containing 5 % CO_2 for 1-4 weeks. Identification of astrocytes was made by using immunohistological staining for glial fibrillary acidic protein (GFAP).

2.2. [Ca^{2+}] measurements in astrocytes

Astrocytes were exposed to both Aβ25-35 and serotonin. In addition, the cells on the coverslips were loaded for 30 min in a 4 μM Fura-2AM buffer before imaging, which was performed using a monochromator-based spectrophotofluorimetric system with dual excitation at the 340 and 380 nm wavelengths, bandpass of 2 nm, and the fluorescence emission measurements at 510 nm wavelength. Results are shown as a ratio of the emissions obtained by the two wavelengths (340/380). As Fura-2AM is a fluorescent dye which binds to free intracellular calcium, the ratio of the emissions at 340 and 380 nm wavelengths is directly correlated to the amount of intracellular calcium (as presented in [7]).

3. MODEL FORMATION

Before beginning to formulate a model for any phenomenon, it is essential to agree on the level of simplification in the model. Certain level of simplification is always necessary for computational models, as the number of variables and equations must be limited in view of obtaining reasonable computation times. However, some players in the phenomenon can be mimicked without being explicitly modeled in the system.

3.1. Model components

Here, the aim is to model the changes in both the intracellularly released calcium and the calcium flux through membranes due to additions of the neurotransmitter serotonin and Aβ. The model takes into account three physiological phenomena known to be the major contributors in describing the intracellular calcium oscillations, namely (1) the flux of Ca^{2+} from/to ECM (extracellular matrix), (2) the pumping of Ca^{2+} from cytosol to the ER (endoplasmic reticulum) and a leak from the ER back to cytosol, (3) the release of Ca^{2+} from the ER via inositol (1,4,5)-trisphosphate (IP_3) receptors and the phenomenon called Ca^{2+} induced Ca^{2+} release. Lavrentovich and Hemkin have recently modeled spontaneous Ca^{2+} oscillations in astrocytes in [12]. They used three ordinary differential equations to model the three processes affecting the cytosolic [Ca^{2+}].

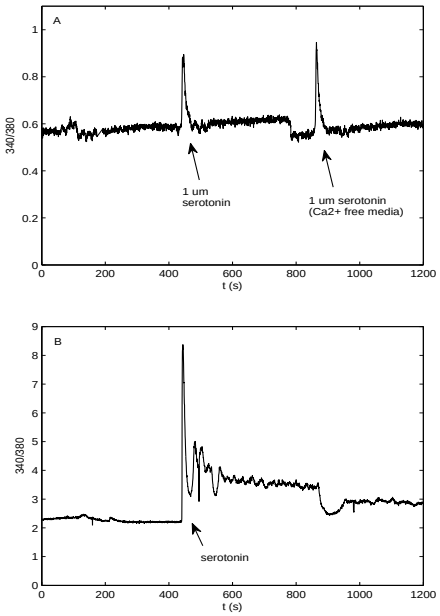


Figure 4. Changes in intracellular Ca^{2+} concentration measured in experiments. **(A)** Transient Ca^{2+} peaks induced by 1 μM serotonin both in Ca^{2+} containing and Ca^{2+} free media. **(B)** The synergistic effect of 200 nM Aβ25-35 and serotonin.

However, the existing models for Ca^{2+} oscillation in astrocytes do not reproduce the observed stochastic behavior, an external stimulus e.g. the effect of neurotransmitters, let alone the effect of Aβ. With some modifications to the model introduced in [12], an external stimulus can be modeled to mimic the effect of neurotransmitters and Aβ. In addition, with stochastic methods also the stochastic behavior of the [Ca^{2+}] responses can be modeled.

3.2. Model constraints

It has been concluded, that small amounts of Aβ25-35 do not cause persistent calcium leak into astrocytes, though calcium leak has been suggested to be one possible cause for neuronal leak in AD [6]. Thus, a low concentration of Aβ42 or Aβ25-35 solely, does not in general have any persistent visible effect on the intracellular Ca^{2+} concentration. On the other hand, the addition of serotonin causes a transient elevation of intracellular [Ca^{2+}], most probably due to a ligand binding to the serotonin receptor [7] and G-protein mediated pathways, modeled, e.g. in [13].

The intracellular [Ca^{2+}] will recover to its original level soon after the addition of serotonin, where a transient peak is seen when adding 1 μM serotonin regardless of the original external [Ca^{2+}], see Fig. 4A. Thus, a Ca^{2+} peak appears even in Ca^{2+} free media. This implies that the Ca^{2+} liberation happens from the intracellular pools, e.g. from ER, mitochondria, and other Ca^{2+} stores. At this point Ca^{2+} is not yet transported from the extracellular space, although the membrane potential may be changed.

Aβ25-35 has no effect on the duration of the serotonin

induced [Ca^{2+}] peak of intracellular release. However, the amplitude of the peak is increased substantially when Aβ25-35 is added together with serotonin, see Fig. 4B. Thus, synergistic tendencies between Aβ and serotonin is clearly seen from the experimental data. Further analysis will be made on the later component of calcium influx through ion channels, also seen in Fig. 4B.

4. CONCLUSION

It is here presented by experimental measurements how even small amounts of Aβ fragments in the brain tissue can, together with e.g. neurotransmitters such as serotonin, induce a meaningful change in the intracellular Ca^{2+} concentration. The pathway from separate effects of Aβ and neurotransmitters to their synergistic effect must be known before the effects and devastating consequences of Aβ aggregates can be hindered.

The phenomenon to be described with a computational model can be sensitive to simplifications. A simplified computational model for Ca^{2+} oscillations, as in [12], has been taken here as an elementary model to be verified and expanded. In addition to basic Ca^{2+} oscillations, to mimic the synergistic effects of Aβ25-35 and serotonin an external stimulus has to be included in the model. However, the model needs to be kept relatively simple due to computational constraints. Therefore, some players in the phenomenon to be modeled can be mimicked without being explicitly modeled with distinct parameters.

Future work will explore in more detail the complex mechanisms leading to Aβ and neurotransmitters induced Ca^{2+} oscillations in astrocytes. Then, the complete computational model can be used to study pathological phenomena associated with AD. This may help to clarify the means to alter the advancement of AD via learning to prevent the formation and thus the devastating symptoms of Aβ aggregations and plaque formations in AD patient's brain.

5. ACKNOWLEDGMENTS

This work was supported in part by Tampere University of Technology Graduate School, Tampere Graduate School in Information Science and Engineering (TISE), the Academy of Finland, No. 213462 (Finnish Programme for Centres of Excellence in Research 2006-2011), 106030, 107694, and 124615, and Emil Aaltonen Foundation.

6. REFERENCES

[1] M. Mattson, "Pathways towards and away from Alzheimer's disease," *Nature*, vol. 430, pp. 631 – 639, 2004.

[2] C. Branden and J. Tooze, *Introduction to Protein Structure, 2nd ed.*, Garland Publishing, Inc., New York, 1999.

[3] A. D'Ursi, M. Armenante, R. Guerrini, S. Sarvadori, G. Sorrentino, and D. Picone, "Solution structure of Amyloid β-peptide (25-35) in different media," *J. Med. Chem.*, vol. 47, pp. 4231 – 4238, 2004.

[4] O. Crescenzi, S. Tomaselli, R. Guerrini, S. Salvadori, A. D'Ursi, P. Temussi, and D. Picone, "Solution structure of the Alzheimer amyloid β-peptide (1-42) in an apolar microenvironment - Similarity with a virus fusion domain," *FEBS Letters*, vol. 269, pp. 5642 – 5648, 2002.

[5] N. Guex and M. Peitsch, "SWISS-MODEL and the Swiss-PdbViewer: An environment for comparative protein modeling," *Electrophoresis*, vol. 18, pp. 2714 – 2723, 1997.

[6] T. Jalonen, C. Charniga, and D. Wielt, "β-Amyloid peptide-induced morphological changes coincide with increased K^+ and Cl^- channel activity in rat cortical astrocytes," *Brain. Res.*, vol. 746, pp. 85 – 97, 1997.

[7] H. Kimelberg, Z. Cai, P. Rastogi, C. Charniga, S. Goderie, V. Dave, and T. Jalonen, "Transmitter-induced calcium responses differ in astrocytes acutely isolated from rat brain and in culture," *J. Neurochem.*, vol. 68, pp. 1088 – 1098, 1997.

[8] I. Smith, K. Green, and F. LaFerla, "Calcium dysregulation in Alzheimer's disease: Recent advances gained from genetically modified animals," *Cell Calcium*, vol. 38, pp. 427 – 437, 2005.

[9] L. Bojarski, J. Herms, and J. Kuznicki, "Calcium dysregulation in Alzheimer's disease," *Neurochem. Int.*, vol. 52, pp. 621 – 633, 2008.

[10] T. O. Jalonen, *Astrocytes and the Regulation of Brain Function: In Vitro Studies on Ion Channels and Intracellular Calcium Changes in Rat Cortical Astrocytes*, Ph.D. Thesis, University of Tampere Medical School, ISBN 952-90-9085-4, Cityoffset Oy. (Finland), 1997.

[11] P. Hof, B. Trapp, J. de Vellis, L. Claudio, and D. Colman, *From molecules to networks - An introduction to cellular and molecular neuroscience*, chapter Cellular components of nervous tissue, pp. 1 – 29, Elsevier Science (USA), 2004.

[12] M. Lavrentovich and S. Hemkin, "A mathematical model of spontaneous calcium(II) oscillations in astrocytes," *J. Theor. Biol.*, vol. 251, pp. 553 – 560, 2008.

[13] E. Mäkiraatikka, A. Saarinen, and M.-L. Linne, "Modeling G-protein induced protein kinase C activation cascade: deterministic and stochastic simulations," in *Proc. of the 5th IEEE International Workshop on Genomic Signal Processing and Statistics (GENSIPS'07)*, Tuusula, Finland, 2007.

EVALUATION OF MEMORY EFFECTS OF KEY METABOLITES IN A FERMENTATIVE H_2-PRODUCTION BIOPROCESS

Nikhil[1,2], Perttu E.P. Koskinen[2], Ari Visa[1], Jaakko A. Puhakka[2], Olli Yli-Harja[1]

[1]Department of Signal Processing, Tampere University of Technology,
[2]Department of Chemistry and Bioengineering, Tampere University of Technology,
P.O. Box 553, FI-33101 Tampere, Finland,
nikhil@tut.fi

ABSTRACT

In this study, an experimental dataset from a suspended-cell bioreactor for H_2 dark fermentation was analyzed using Pearson's product-moment correlations and multiple linear regressions. The aim of the study was to evaluate the effect of previous values of the key metabolites (ethanol, acetate and butyrate) on H_2 production rates. The results show that none of the metabolites have very strong correlation with H_2 production rate. H_2 production rate has a maximum and only positive correlation of 0.69 with butyrate. The inclusion of the previous values of the metabolites in regression models doesn't improve the model performance. The global multiple linear regression models are not efficient in modeling the H_2 production rate as a function of operational parameters (hydraulic retention time, pH), volatile fatty acids (acetate, butyrate, propionate, valerate) and alcohol (ethanol) concentrations.

INTRODUCTION

The dependence on fossil fuels as an energy source results in global warming, air pollution and environmental and health problems. H_2 produced from renewable energy sources offers a clean alternative for the fossil fuels [1]. H_2 is an ideal energy carrier as it is transportable, storable and globally available, and has an efficient and emission free conversion to electricity. H_2 also has multiple uses in industrial applications including hydrogenation processes (saturation of compounds, cracking of hydrocarbons, removal of sulphur and nitrogen compounds), O_2 scavenger for corrosion and oxidation prevention, and coolant in electrical generators [2]. Today, H_2 is mainly produced from fossil fuels by energy intensive processes, such as steam reforming or coal gasification [2]. Several techniques for H_2 production from renewable sources exist including microbiological fermentation processes [3].

Microbiological dark fermentation can be used to produce H_2 from biomass or organic waste materials [4], [5]. H_2 production through dark fermentation is an intermediary step in the anaerobic degradation of organic material. H_2 is produced in order to maintain the electron balance in the anaerobic system. In dark fermentative H_2 production, gases (H_2 and CO_2) and organic acids and alcohols (e.g. ethanol, acetate, butyrate, propionate and valerate) are the end products of the bioprocess. H_2-fermenting mixed cultures are characterized by complex behavior and interaction between H_2-producing, H_2-consuming and neither H_2 -consuming nor -producing organisms [6]. There is a need to understand the relationship that exists between several end products, and utilize this information to better comprehend the complex dynamic behavior of the system [7], [8]. Modeling of H_2 fermentation processes may offer a means to reveal and describe better these complex interactions and to provide information for the optimization of H_2 production bioprocesses.

In this study, H_2 fermentation was operated and monitored in a suspended-cell bioprocess for over 5 months. The Pearson's product-moment correlation was used to analyze the dependencies of H_2 production rate with other end products. The multiple linear regressions were used to model the H_2 production rate as function of HRT, ph, ethanol, acetate, propionate, butyrate; and previous values of these variables. The aim of this study was to evaluate the memory effect of key metabolites (ethanol, acetate and butyrate) on the H_2 production rates.

MATERIALS AND METHODS

Experimental Design

The H_2-fermenting microbial community was enriched from an anaerobic digester treating municipal wastewater sludge as described in [6]. The bacteria closely affiliated with *Clostridium butyricum* and *Escherichia coli* dominated the microbial community [6]. The bioreactor was inoculated with 150 ml of the enrichment culture.

A completely mixed bioreactor (total volume 0.8 l, height to diameter ratio 7.7) with a gas extraction module was used for H_2 production at 35°C (Figure 1). The bioreactor was operated under anaerobic conditions. Gas extraction module was installed in the recycle line and consisted of 3.7 m of gas permeable silicone tubing inside a vacuumed chamber. Bioreactor was operated continuously for 156 days and reactor performance was determined by measuring gaseous and soluble end prod-

ucts, glucose consumption and biomass concentrations. On day 62, the gas extraction module was uninstalled for 30 days to study its effect on the bioreactor performance. The constituents of bioreactor feed were as earlier described [6], except that glucose concentration was kept constant (5 g l-1). Gas production was measured by wet gas meters (Ritter Apparatebau, Bochum, Germany). H_2 production rate was determined by adding together production rates from top of the reactor, and from the gas extraction.

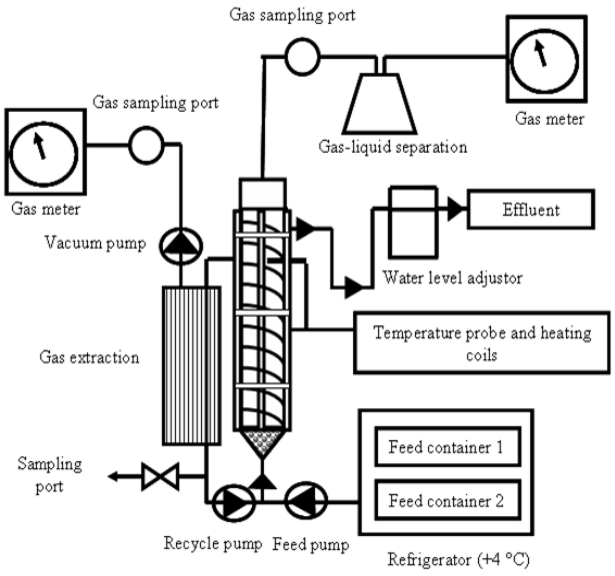


Figure 1. Bioreactor system configuration. Feed containers 1 and 2 contained glucose and buffered nutrients solutions, respectively.

Chemical Analyses

The gaseous end products (H_2, CO_2) were analyzed using a HP 5890II gas chromatograph equipped with a 6 ft Porapak N column (80 / 100 mesh), and a thermal conductivity detector. Oven, injector and detector temperatures were 50, 80 and 80 °C, respectively. N_2 was used as carrier gas. The formation of organic acids and alcohols was measured using a HP 5890II gas chromatograph with a 30 m DB-FFAP capillary column (Agilent Industries Inc, Palo Alto, CA, U.S), and a flame ionization detector. Residual glucose concentrations were analyzed colorimetrically (Shimadzu UV-1601) by anthrone-method [9]. Biomass as volatile suspended solids (VSS) was analysed according to APHA Standard Methods [10].

Dataset

The dataset used in this study consisted of 7 variables: operational parameters (HRT and pH), volatile fatty acid concentrations (butyrate: HBu, acetate: HAc, propionate: HPr, valerate: HVa) and alcohol (ethanol: EtOH), and hydrogen production rate (H_2PR) and carbon dioxide production rate (CO_2PR). There were 151 measurements spread over 156 days. The minimum, mean and maximum values of measured variables are listed in Table 1.

Table 1. Details of the experimental dataset used in the study. Min: minimum value observed, Max: maximum value observed

	Min	Mean	Max
HRT(h)	1.5604	1.9378	2.6956
H_2PR (mmol/l/h)	0	2.6391	18.7879
CO_2PR (mmol/l/h)	0	11.478	30.3008
pH	5.059	5.9715	7.213
EtOH (mol/l)	0.0008	0.0042	0.0104
HAc (mol/l)	0.0014	0.0113	0.0255
HPr (mol/l)	0	0.0008	0.0064
HBu (mol/l)	0.0001	0.0037	0.0099
HVa (mol/l)	0	0.0002	0.0008

Multiple Linear Regression

Multiple linear regression is a statistical technique used for technical and fundament analyses of multivariable datasets [11]-[14]. It models the relationship between two or more explanatory variables and a response variable by fitting a linear equation to an observed data. The model for multiple linear regression, given n observations is:

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \ldots + \beta_k x_{ik} + \varepsilon_i \quad (1)$$

Where $i = 1,2,\ldots n.$, y is the dependent variable, x is the set of k explanatory variables, and ε is the residual term.

Pearson Product-Moment Correlation

In statistics, the Pearson product-moment correlation coefficient is the most common measure of the correlation between two variables X and Y [15], [16]. It is represented by rho (ρ), when measured in a population. It is represented by (r), when computed in a sample. The Pearson coefficient is obtained as equations 2 and 3.

$$\rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X \sigma_Y} \quad (2)$$

Where $cov(X,Y)$ is the covariance between X and Y, and σ_X and σ_Y are the standard deviations of X and Y.

$$r_{xy} = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n\sum x_i^2 - (\sum x_i)^2}\sqrt{n\sum y_i^2 - (\sum y_i)^2}} \quad (3)$$

Where $i = 1, 2\ldots n$, and n is the sample size.

Correlation is a bivariate measure of association between two variables. It varies from 0 (random relationship) to 1 (perfect linear relationship) or -1 (perfect negative linear relationship). It is symmetrical in nature and doesn't provide the direction of causation.

RESULTS AND DISCUSSION

The Pearson's product-moment correlations were used for analyzing the relationship between H_2 production rate and pH ethanol, acetate, propionate, butyrate and valerate. The H_2 production rate was also correlated with one (t-1) and two (t-2) previous time-step values of pH, ethanol, acetate, propionate, butyrate and valerate. Table 2 lists the correlation coefficients obtained. It can be seen that H_2 production rate does not correlate very strongly with any of the variables (also from Figure 2). It has a moderate positive correlation with butyrate. For all the other variables, H_2 production rate shows a moderate negative correlation. The valerate seems to have neutral effect on the H_2 production rates. The effects of these variables were further analyzed by modeling the H_2 production rate as a function of these variables and also the previous values of the variables.

Table 2. Pearson's correlation coefficients for hydrogen production rate against other variables. Pcorr: partial correlations of Hydrogen production rates with other variables at time't', 't': the values are from the same time points, 't-1': the values are from one previous time step, 't-2': the values are from two previous time steps.

	Pcorr	t	t-1	t-2
pH	-0.09	-0.31	-0.36	-0.35
EtOH	0.25	-0.24	-0.22	-0.19
HAc	-0.35	-0.41	-0.38	-0.36
HPr	0.09	-0.28	-0.26	-0.26
HBu	0.61	0.69	0.57	0.63
HVa	-0.35	-0.06	0.02	0.06

In Table 3, the mean square error (MSE) values obtained for different multiple regression models formulated are listed. The best observed model (MSE=2.5934) is the model 2, which models H_2 production rate as a function of HRT, pH, ethanol, acetate, propionate, butyrate, valerate and one time-step previous values of pH, ethanol, acetate, propionate, butyrate and valerate. In comparison with other model, results obtained in this study, model 2 was not significantly better than others. In the model 1, H_2 production rate was modeled as the function of variables without including any previous values. All the other remaining models included previous values of some of the observed variables. It can be seen from Table 3, even the inclusion of the previous values didn't improve the model performances. The reason could be due to few very high H_2 production rate values which were not predicted close enough by any of the models. The model was however able to capture the general trend of the experimental measurements. Figure 3 shows the results for model 2. Also, the measurements were not done in equal intervals of time, and hence the estimation of regression parameters wouldn't have been efficient when previous values were included.

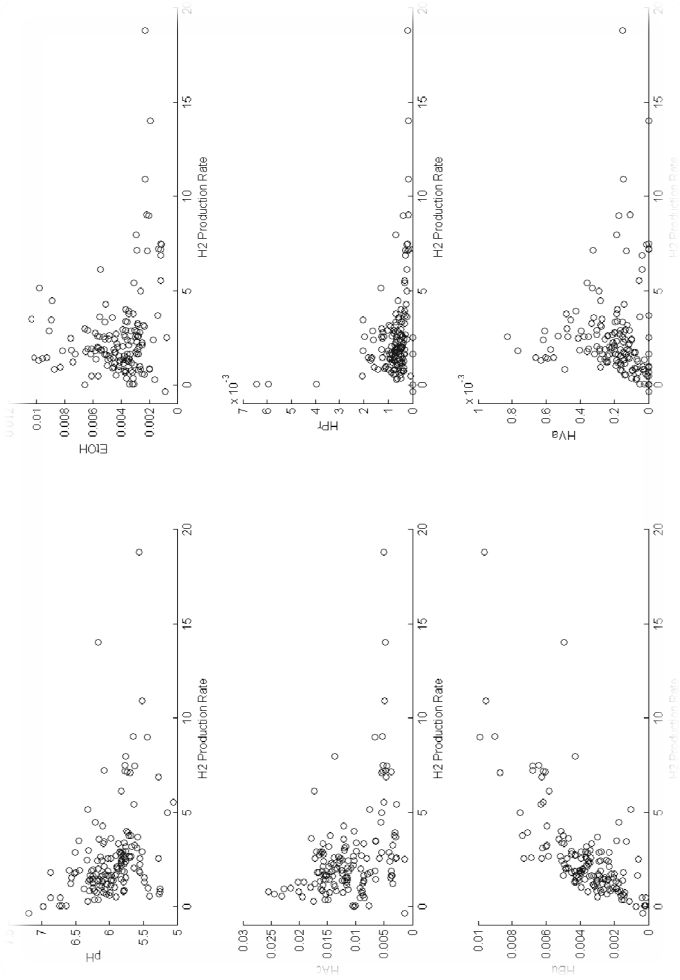


Figure 2. Scatter plots of Hydrogen production rate with other variables.

The same dataset was used in our previous study, where H_2 production rate was modeled using clustering hybrid regression (CHR) approach [17]. In CHR approach, none of the previous values of the variables were included. The MSE values observed were 0.55 [17]. Based on MSE values, the models in this study perform nearly six times poorer than those reported in [17] for same dataset.

The global multiple linear regression, as in this study, are not the best statistical tools to model the H_2 production rate as a function of other end products. However, local multiple regression models as developed using CHR approach are efficient in modeling the H_2 production rates [17].

CONCLUSION

The glucose based H_2 dark fermentation bioprocess was analyzed using Pearson's product-moment correlations and multiple regression models. The memory effect of the key metabolites (ethanol, acetate and butyrate) on H_2

production rates was analyzed. The correlation analyses and model results suggest that the previous values of pH, ethanol, acetate, propionate, butyrate and valerate have no significant impact on the H_2 production rates.

Table 3. Mean square error (MSE) values for different regression models. Model 1: function (HRT, pH, EtOH, HAc, HPr, HBu and HVa). Model 2: function (HRT, pH, EtOH, HAc, HPr, HBu, HVa and one time-step previous values of (pH, EtOH, HAc, HPr, HBu, HVa)). Model 3: function (HRT, pH, EtOH, HAc, HPr, HBu, HVa, one time-step previous values of (pH, EtOH, HAc, HPr, HBu, HVa) and two time-step previous values of (pH, EtOH, HAc, HPr, HBu and HVa)). Model 4: function (HRT, pH, EtOH, HAc, HPr, HBu, HVa and one time-step previous values of (pH, EtOH, HAc, HPr and HBu)). Model 5: function (HRT, pH, EtOH, HAc, HBu and one time-step previous values of (pH, EtOH, HAc and HBu))

	MSE
Model 1	2.6531
Model 2	2.5934
Model 3	3.2641
Model 4	2.8854
Model 5	2.9484

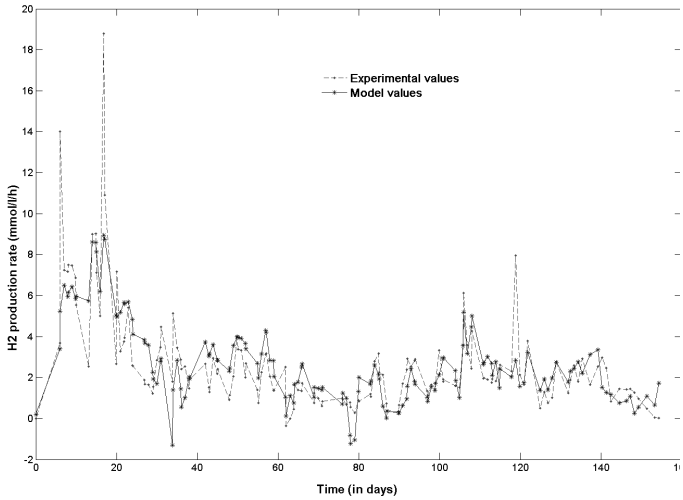


Figure 3. Hydrogen production rate: Experimental and model values. The multiple regression results for Model 2 (mentioned in Table 3). MSE = 2.5934

ACKNOWLEDGMENTS

This research was funded by the Academy of Finland (HYDROGENE project, no. 107425), Nordic Energy Research (BIOHYDROGEN project, no. 28-02) and Tampere University of Technology Graduate School (P. E. P. Koskinen). The work was also supported by the Academy of Finland (application number 213462, Finnish Programme for Centers of Excellence in Research 2006-2011).

REFERENCES

[1] J'O.M. Bockris, "The origin of ideas on a hydrogen economy and its solution to the decay of the environment," *Int J Hydrogen Energy,* vol. 27, pp. 731-740, 2002.

[2] D. Das, and T.N. Veziroğlu, "Hydrogen production by biological processes: a survey of literature," *Int J Hydrogen Energy,* vol. 26, pp. 13-28, 2001.

[3] J. Benemann, "Hydrogen biotechnology: Progress and prospects," *Nat Biotechnol,* vol. 14, pp. 1101-1103, 1996.

[4] I.K. Kapdan, and F. Kargi, "Bio-hydrogen production from waste materials," *Enzyme Microb Tech,* vol. 38, pp. 569-582, 2006.

[5] C. Li, and H.H.P. Fang, "Fermentative hydrogen production and wastewater and solid wastes by mixed cultures," *Crit Rev Env Sci Technol*, vol. 37, pp. 1-39, 2007.

[6] P.E.P. Koskinen, A.H. Kaksonen and J.A. Puhakka, "The relationship between instability of H_2 production and compositions of bacterial communities within a dark fermentation fluidized-bed bioreactor," *Biotechnol Bioeng*, vol. 97(4), pp. 742-758, 2007.

[7] C.-Y. Lin, and R.C. Chang, "Fermentative hydrogen production at ambient temperature," *Int J Hydrogen Energy*, vol. 29, pp. 715-720, 2004.

[8] J. Rodriguez, R. Kleerebezem, J.M. Lema, and M.C. van Loosdrecht, "Modeling product formation in anaerobic mixed culture fermentations," *Biotechnol Bioeng*, vol. 93, pp. 592-606, 2006.

[9] J. Hansen, and I. Møller, "Percolation of starch and soluble carbohydrates from plant tissue for quantitative determination with anthrone," *Anal Biochem*, vol. 68, pp. 87-94, 1975.

[10] APHA. 1995. *Standard methods for the examination of water and wastewater*. 19th edn, American Public Health Association, Washington DC, U.S.

[11] V. E. McGee, and W. T. Carleton, "Piecewise regression," *J. Am. Stat. Assoc.*, vol. 65, pp. 1109–1124, 1970.

[12] M. N. Karim, D. Hodge, and L. Simon, "Data-based modeling and analysis of bioprocesses. Some real experiences," *Biotechnol. Prog.*, vol. 19, pp. 1591–1605, 2003.

[13] W. S. Cleveland, E. H. Grosse, and W. M. Shyu, *Local regression models*. London: Chapman and Hall, J. M. Chambers, and T. J. Hastie, Eds., 1992, pp. 309–376.

[14] Y. Chen, G. Dong, J. Han, B. W. Wah, and J. Wang, "Multi-dimensional regression analysis of time-series data streams," *Proc. 28th Int. Conf. Very Large Data Bases*, Hongkong, China, pp. 323–334, 2002.

[15] J. Cohen, P. Cohen, S.G. West, and L.S. Aiken, *Applied multiple regression/correlation analysis for the behavioral sciences.* (3rd ed.) Hillsdale, NJ: Lawrence Erlbaum Associates, 2003.

[16] Edwards, A. L. "The Correlation Coefficient." Ch. 4 in *An Introduction to Linear Regression and Correlation.* San Francisco, CA: W. H. Freeman, pp. 33-46, 1976.

[17] Nikhil, P. E. P. Koskinen, A. Visa, A. H. Kaksonen, J. A. Puhakka, and O. Yli-Harja, "Clustering hybrid regression (CHR): a novel computational approach to study and model biohydrogen production through dark fermentation," *Bioproc Biosyst Eng*, 2008, DOI: 10.1007/s00449-008-0213-9.

DECOMPOSING GENE EXPRESSION INTO REGULATORY AND DIFFERENTIAL PARTS WITH BAYESIAN DATA FUSION

Janne Nikkilä[1]*, Timo Erkkilä*[2] *and Harri Lähdesmäki*[2]

[1]Adaptive Informatics Research Centre & Helsinki Institute for Information Technology
Department of Computer and Information Science
Helsinki University of Technology
P.O. Box 5400, FI-02015 TKK, Finland
janne.nikkila@tkk.fi
[2]Department of Signal Processing
Tampere University of Technology
P.O. Box 553, FI-33101 Tampere, Finland
{timo.p.erkkila,harri.lahdesmaki}@tut.fi

ABSTRACT

The two main interests in gene expression data, differential expression and transcriptional regulatory effects, are usually difficult to separate from each other. We propose a method for decomposing observed gene expression data into i) a part explainable directly by transcription factor (TF) mRNA level, and ii) a part attributable to other effects induced by experimental setting. The method fits a Bayesian hierarchical linear model to the expression data given prior information about transcriptional regulatory mechanisms. Our primary source of prior information are TF binding probabilities, derived from a probabilistic model for TF binding to gene regulatory sequences. The proposed method can be easily extended to include additional and other types of prior information (ChIP-chip, other gene expression data), and the same modeling framework can be used to make inference regarding a large variety of questions. Simulation results show that, relative to standard approaches, the proposed method can better detect regulatory relations and that it is also able to distinguish general differential expression from the effects of direct regulatory mechanisms.

1. INTRODUCTION

Detection of differential gene expression induced by the chosen experimental setting is the primary focus in most microarray gene expression studies. In addition to pure differential expression, key interests are the gene regulation effects taking place during the experiment. While gene regulation can happen in various stages, one of the most important of these stages is direct transcriptional regulation by transcription factor (TF) proteins. Approximate protein levels of TFs can in principle be monitored through the expression levels of the genes coding the factor proteins. However, estimating these regulatory effects from gene expression data is not a trivial task. The most important challenges include problems in estimation due to small number of samples, regulatory relations is confounded with co-expression of genes and TFs, and differential expression of a gene might be induced by other TFs than the known regulating TFs.

Discovering regulatory relations from high-throughput gene expression data has been in focus since the emergence of microarrays. The earliest and most common attempts for finding relations between genes were based on detecting genes with similar behaviour in the experiments [1]. Naturally, this results in a set of genes consisting both the regulators, the target genes, and the co-expressed genes with no means to distinguish between them. More focused approaches have been presented as well which rely on prior knowledge about the potential regulators. For example, Bayesian networks have also been applied to estimate regulatory relations between a set of known TFs and the rest of the genes as groups (see [2]). However, practically all these approaches by-pass the actual interest of a single microarray gene expression experiment: differential expression in the given situation.

Statistical models for differential expression range from early fold-change based approaches to classical ANOVA based models, and their Bayesian variants (see [3, 4]). But, analogously to regulation models, differential expression models largely disregard the estimation of the regulatory effects.

We propose here a model that integrates gene expression data with transcriptional regulatory knowledge, such as transcription factor binding site location information. Using the binding probabilities from a probabilistic model for transcription factor binding to gene promoter region as a prior, the proposed model can in principle distinguish which genes are transcriptionally regulated by known TFs in any given experiment, based on the observed expression data. In particular, the model is capable of distinguishing whether some known, differentially expressed TF is causing its target genes to be differentially expressed in the

given experiment. In addition, the model is able to discover differential expression of target genes due to other reasons than the known TF, in case the regulatory TF is not differentially expressed. The model is formulated in Bayesian framework enabling a natural way to handle the uncertainty in the data and small sample size problem.

The results show that the proposed model can detect regulatory relations taking place during the experiment more efficiently than mere co-expression based approaches, and at the same time detect differential expression induced by the experiment.

2. METHODS

Our approach is closely related to a Bayesian hierarchical model for gene expression allowing heterogenous errors (HEM) [4]. HEM separates the technical noise from the biological noise, and is shown the perform favorably in the analysis of gene expression data. We extend the standard HEM by incorporating an additional regression term that allows explicit modeling of direct transcriptional regulation. Further, the regression term allows to incorporate prior knowledge about transcriptional regulation into the hierarchical error model. The prior information can come from a variety of different sources, such as sequence-based TF binding predictions [5], ChIP-chip data [6], other gene expression data.

We propose to model the observed expression $y_{i,j,k}$ for ith gene, jth condition, and kth replicate with a linear model as

$$y_{i,j,k} = \mu + g_i + d_j + r_{i,j} + a_i \cdot z_i \cdot x_{TF,j} + \epsilon_{i,j,k}, \quad (1)$$

where μ is the general mean, g_i is the effect of ith gene, d_j is the effect of the jth condition, $r_{i,j}$ is their joint effect, $z_i \cdot a_i \cdot x_{TF,j}$ describes the regulatory effect of the given TF with TF's expression level $x_{TF,j} = \mu + g_{TF} + d_j + r_{TF,j}$, regulation strength a_i and a binary indicator z_i of whether the TF regulates the ith gene. The residual variance is described with $\epsilon_{i,j,k}$. For the expression measurements of the TF we use the same model but without the regression term, i.e. $y_{TF,j,k} = x_{TF,j} + \epsilon_{i,j,k}$.

We assume the following prior distributions with fixed parameters:

$$\begin{aligned}
\mu &\sim N(\mu_\mu, \sigma_\mu^2) && (2)\\
g_i, g_{TF} &\sim N(\mu_g, \sigma_g^2) && (3)\\
d_j &\sim N(\mu_d, \sigma_d^2) && (4)\\
r_{i,j}, r_{TF,j} &\sim N(\mu_r, \sigma_r^2) && (5)\\
a_i &\sim N(\mu_a, \sigma_a^2) && (6)\\
z_i &\sim Bernoulli(\theta_i) && (7)\\
\epsilon_{i,j,k} &\sim N(0, \tau_{i,j}^2) && (8)\\
\tau_{i,j}^{-2}, \tau_{TF,j}^{-2} &\sim Gamma(\alpha, \beta). && (9)
\end{aligned}$$

Note that the error variance is allowed to be heterogeneous, i.e. different for each gene and condition. Prior information about transcriptional regulation can easily be incorporated via θ_i parameters. The proposed model is able to analyze only one TF at a time. The graphical model is presented in Figure 1.

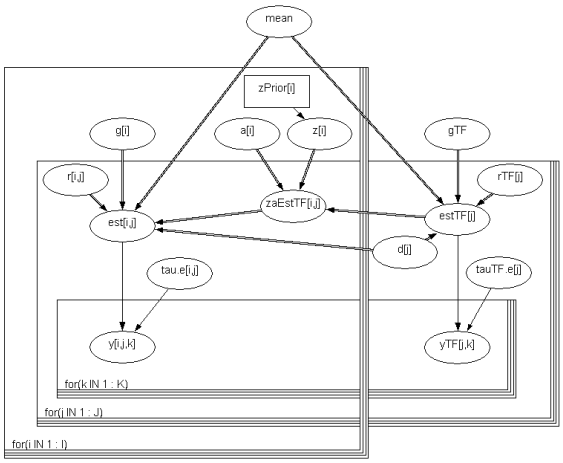


Figure 1. A graphical representation of the proposed model. The boxes indicate loops over samples i, conditions j, and replicates k.

The unknown model parameters are estimated with Gibbs sampling in WinBUGS [7]. Convergence to the posterior is assessed using the potential scale reduction factor method of Gelman et al. [8]. Posterior mean is used as the final estimate for each parameter.

3. RESULTS

In simulations, our primary aim is to demonstrate that the proposed model can detect the regulatory relations based on prior information. The proposed model is also compared to a simplified HEM [4], where the hierarchy distinguishing technical noise from the biological noise has been dropped away. Note that the error hierarchy could be added analogously to both models.

The proposed model is tested with simulated data generated from a model similar to the proposed model. We consider a model that consists of 100 genes (i), 2 conditions (j) and a single TF. The Gibbs sampling is initialized with sample mean values and is run for 1000 burn-in steps after which a sample of size 2000 is collected. The potential scale reduction factor convergence diagnostic indicates that this is typically sufficient for the Gibbs sampling to converge. Parameters are set as follows: $\mu_g = \mu_d = \mu_r = \mu_a = 0$, $\sigma_\mu^2 = 100$, $\sigma_g^2 = 1$, $\sigma_d^2 = 1$, $\sigma_r^2 = 1$, $\sigma_a^2 = 1$, $\alpha = 1$, $\beta = 0.5$. For the μ_μ parameter we use the empirical sample mean of all the measurements. Data is generated from the above model (priors) except that $\mu = 0$ and $a_i = \pm 1.5$, the additive noise ϵ is sampled from the standard normal, and 10% of z_i terms are uniformly randomly set to 1, others are 0. In the first simulation we assume to have three replicates (k) and vary $\theta_i \in \{0.5, 0.55, 0.6, 0.65\}$ for those i that corresponds to the underlying regulatory mechanisms (i.e., true $z_i = 1$) and $\theta_i \in \{0.5, 0.45, 0.4, 0.35\}$ for the others (i.e., true $z_i = 0$). In the second simulation we set $\theta_i = 0.5$ for all i and vary the number of replicates $k \in \{2, 3, 5, 10\}$. Both

simulations are repeated 50 times and average results are reported. Each individual simulation with 100 genes and varying number of replicates takes only about (on the order of) minutes to run in WinBUGS and, thus, the method should be fast enough to analyze thousands of genes.

Figure 2 (a) shows how the proposed model can detect the true regulatory relations from the simulated data with varying degrees of prior information. For the receiver operating characteristic (ROC) curves the potential target genes for the TF are estimated by ranking the genes based on the absolute magnitude of $a_i \cdot z_i$ term (averaged over posterior samples). The same figure additionally compares the performance of the model to a naive approach where the regulatory relations are estimated by computing the correlations between the estimated TF expression and all the other genes' expression measurements and picking the most strongly correlating genes as potential targets for TF regulation. While the comparison is slightly artificial, it serves as demonstration about the potential of principled data fusion approaches.

Figure 2 (b) demonstrates that as the number of replicates increases the model performance increases as well. This provides further evidence about the correct functioning of the model, but on the other hand also reveals in part that it is somewhat prone to small sample sizes. Figure 2 (b) also suggests that prior information can be more valuable than having more replicates of expression measurements.

The second important difference of the proposed model to the simplified HEM is its ability to detect differential expression that is confounded by a strongly regulating TF. In the proposed model the term r_{ij} captures the changes in gene expression due other reasons than the potential regulating TF. Since the comparison model, the simplified HEM, does not take into account any direct regulation, its estimates of r_{ij} should be erroneus in the cases where there some confounding TF regulator is present. Figure 3 presents the difference between the estimates of r_{ij}s from the proposed model and the comparison model in such cases.

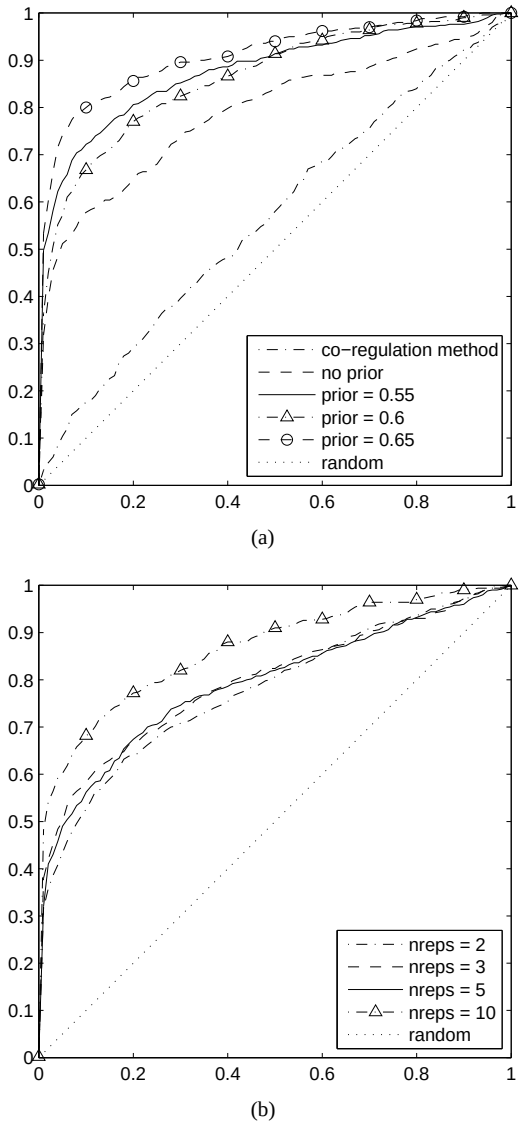


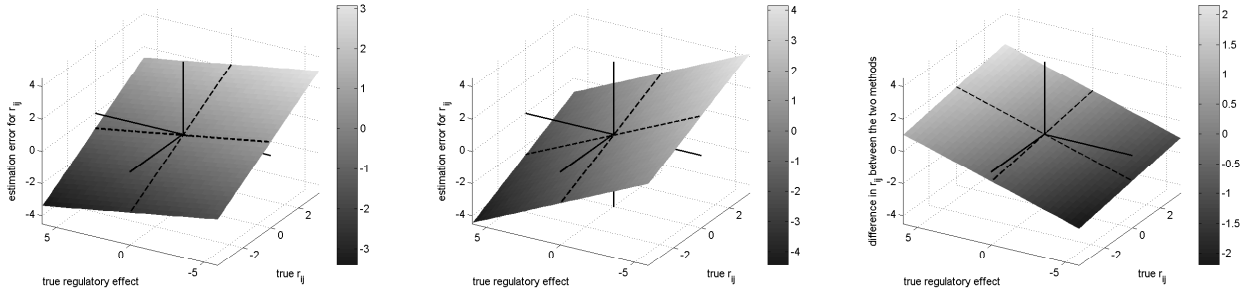
Figure 2. ROCs presenting the effect of (a) the prior strength and (b) the sample size to the model's ability to detect the true regulatory relations, in comparison to a co-expression based model.

4. DISCUSSION

We have proposed a statistical model for gene expression that can estimate separately the expression changes due to TF regulation, and the expression changes due to other reasons (unknown regulators etc.) The model is formulated in Bayesian framework and integrates the knowledge of about the potential regulators as prior data. We showed with simulated data that the model i) detects the true regulatory relations better than simple alternatives, and ii) is able to estimate the differential expression better than the comparison models in the presence of the confounding TF regulation. When there is no TF regulation present the proposed model performs equivalently to the comparison model.

The key aspect of the model is its ability to integrate prior data, such as one that describes binding probabilities of the TF proteins to the promoter regions of the genes. Since the mechanism of integration of prior information is desigend to be as simple as possible, model is versatile enough to incorporate many kinds of binding information, including for example ChIP-chip data and sequence based computationally derived binding probabilities. In particular, in the next stage, the proposed model is going to be extended to utilize a novel probabilistic model providing binding probabilities based directly on the TF motifs and promoter sequence. Since the promoter sequence is known practically for every gene for which expression can be measured, this will enable the discovery of regulatory relations for any TFs whose motifs are known, in the given experiment.

The priors we have used here represent sensible but arbitrary choices. It is clear that they have strong effects on the estimates, especially regarding the discovery of regu-



(a) The proposed model (b) The comparison model (c) Difference

Figure 3. The better ability of the proposed model to estimate differential expression in the presence of a confounding TF regulation, in comparison to the simplified HEM. Subfigures (a) and (b) show the errors in estimates from the proposed model and the comparison model for varying levels of regulation and expression changes due to other reasons (the true r_{ij}). The figures reveal the errors in r_{ij} are smaller for the proposed model than for the comparison model. Both models also make some errors in the high absolute values of true r_{ij}, which is due to selected prior centered around value zero. Subfigure (c) emphasizes how the difference between the models is largest when there is either a large positive or negative TF regulation present by showing directly the difference between the estimates of the models. Note also that the difference between the models' estimates is zero when the true regulatory effect is zero. The estimates are computed from simulated data sets including five replicates, by averaging over posterior samples and by fitting a plan for visualization purposes.

latory relations, but also with respect to other parameters. In the next stage the model will be validated more thoroughly for suitable prior distributions.

While this work focused on studying the functionality of the new model as such, the next stage will be applying the model to real gene expression data with real prior information about the binding probabilities of TFs to gene promoter regions.

5. ACKNOWLEDGMENTS

This work was supported by the Academy of Finland, (application number 213462, Finnish Programme for Centres of Excellence in Research 2006-2011).

6. REFERENCES

[1] M. B. Eisen, P. T. Spellman, P. O. Brown, and D. Botstein, "Cluster analysis and display of genome-wide expression patterns," *Proceedings of the National Academy of Sciences of the USA*, vol. 95, pp. 14863–14868, 1998.

[2] E. Segal, M. Shapira, A. Regev, D. Pe'er, D. Botstein, D. Koller, and N. Friedman, "Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data," *Nature Genetics*, vol. 34, pp. 166–176, 2003.

[3] M. Kerr, M. Martin, and G. A. Churchill, "Analysis of variance for gene expression microarray data," *Journal of Computational Biology*, vol. 7, pp. 819–837, 2000.

[4] H. Cho and J. K. Lee, "Bayesian hierarchical error for analysis of gene expression data," *Bioinformatics*, vol. 20, no. 13, pp. 2016–2025, 2004.

[5] H. Lähdesmäki, A. G. Rust, and I. Shmulevich, "Probabilistic inference of transcription factor binding from multiple data sources," *PLoS ONE*, vol. 3, no. 3, e1820.

[6] B. Ren, F. Robert, J. J. Wyrick, O. Aparicio, E. G. Jennings, I. Simon et al., "Genome-wide location and function of DNA binding proteins," *Science*, vol. 290, no. 5500, pp. 2306–2309, 2000.

[7] D. J. Lunn, A. Thomas, N. Best and D. Spiegelhalter, "WinBUGS — a Bayesian modelling framework: concepts, structure, and extensibility," *Statistics and Computing*, vol. 10, pp. 325–337, 2000.

[8] A. Gelman, J. B. Carlin, H. S. Stern and D. B. Rubin, *Bayesian Data Analysis*, 2nd edition, Chapman & Hall/CRC, 2003.

ON THE IMPACT OF ENTROPY ESTIMATOR IN TRANSCRIPTIONAL REGULATORY NETWORK INFERENCE

Catharina Olsen[1]*, Patrick E. Meyer*[1] *and Gianluca Bontempi*[1]

[1]Machine Learning Group, Université Libre de Bruxelles,
CP212, 1050 Brussels, Belgium
colsen@ulb.ac.be, pmeyer@ulb.ac.be,gbonte@ulb.ac.be

ABSTRACT

The reverse engineering of transcription regulatory networks from expression data is gaining much interest in the bioinformatics community. This work studies the impact of entropy estimation on network inference based on mutual information. The study involves the recently proposed open source R-package MINET for transcription regulatory networks. Five entropy estimators, namely the empirical, the Miller-Madow corrected, a shrinkage estimate, the Schürmann-Grassberger Dirichlet and the Gaussian are compared on a network inference task using synthetically generated microarray datasets. The extensive simulation setting allows us to study the effect of the number of samples and of the presence of missing values.

1. INTRODUCTION

The problem of reverse engineering of transcription regulatory networks from expression data is far from being trivial because of the large number of genes and the poor informational content of expression data [1]. The information-theoretic approaches typically rely on the estimation of mutual information from expression data in order to measure the statistical dependence between the genes [2].
This work studies the influence of noise, missing values, different discretization and estimation methods on the MRNET method for inference of networks using synthetically generated datasets. The study was carried out using the R-package MINET [2] which allows the choice of different inference methods, estimators and discretization methods.
The outline of the paper is as follows. Section 2 introduces the entropy estimators and the discretization methods used in this paper. In Section 3 the R-package MINET is presented and in Section 4 the performed simulation is described. Finally Section 5 concludes the paper.

2. ENTROPY ESTIMATION

This section introduces the entropy estimators used in this paper. Entropy estimation is important in network inference because it allows the computation of the mutual information matrix.
If X is a continuous random variable taking values comprised between a and b, the interval $[a, b]$ can be discretized by dividing this interval into $|\mathcal{X}|$ subintervals, the so-called bins. In the following $\mathcal{X}$ is an index vector. Further $nb(x_k)$ is the number of data points in bin k and $m = \sum_{k\in\mathcal{X}} nb(x_k)$ the number of all data points. If X is a random vector each element X_i can be discretized seperately into $|\mathcal{X}_i|$ bins with index vector $\mathcal{X}_i$.
Let X be a random vector and p a probability measure. The (i, j)th element of the mutual information matrix (MIM) is defined by

$$\begin{aligned} MIM_{ij} &= H(X_i) + H(X_j) - H(X_i, X_j) \\ &= I(X_i; X_j) \\ &= \sum_{k_i\in\mathcal{X}_i}\sum_{k_j\in\mathcal{X}_j} p(x_{k_i}, x_{k_j}) \log\left(\frac{p(x_{k_i}, x_{k_j})}{p(x_{k_i})p(x_{k_j})}\right) \end{aligned}$$

where the entropy of a random variable X is defined as

$$H(X) = -\sum_{k\in\mathcal{X}} p(x_k)\log p(x_k) \tag{1}$$

and $I(X_i; X_j)$ is the mutual information between the random variables X_i and X_j.

2.1. Estimators

This section presents the employed entropy estimators.

2.1.1. Empirical

The empirical estimator, often called maximum likelihood estimator, is the entropy of the empirical distribution

$$\hat{H}^{emp} = -\sum_{k\in\mathcal{X}} \frac{nb(x_k)}{m}\log\frac{nb(x_k)}{m}. \tag{2}$$

It has been shown in [3] that the asymptotic bias of the empirical estimator is

$$bias(\hat{H}^{emp}) = -\frac{|\mathcal{X}| - 1}{2m}. \tag{3}$$

2.1.2. Miller-Madow

The Miller-Madow estimator, [3], takes the asymptotic bias (3) into account and subtracts it from the empirical estimator

$$\hat{H}^{mm} = \hat{H}^{emp} + \frac{|\mathcal{X}| - 1}{2m}. \tag{4}$$

This estimator reduces the bias without decreasing the variance.

2.1.3. Shrink

The shrink estimator, [4], proposes to combine two different estimators for the probability of an event with a weighting factor $\lambda \in [0,1]$

$$\hat{p}_\lambda(x_k) = \lambda \frac{1}{|\mathcal{X}|} + (1-\lambda)\frac{nb(x_k)}{m}. \quad (5)$$

Let λ^* be the value minimizing the mean square function, see [4],

$$\lambda^* = \arg\min_{\lambda\in[0,1]} E\left[\sum_{k\in\mathcal{X}} (\hat{p}_\lambda(x_k) - p(x_k))^2\right]. \quad (6)$$

It has been shown in [5] that the optimal λ is given by

$$\lambda^* = \frac{|\mathcal{X}|(m^2 - \sum_{k\in\mathcal{X}} nb(x_k)^2)}{(m-1)(|\mathcal{X}|\sum_{k\in\mathcal{X}} nb(x_k)^2 - m^2)}. \quad (7)$$

2.1.4. Dirichlet

The Dirichlet estimator is an example of the Bayesian entropy estimation. The prior follows a Dirichlet distribution with parameter vector β

$$f(X;\beta) = \frac{\prod_{k\in\mathcal{X}}\Gamma(\beta_k)}{\Gamma(\sum_{k\in\mathcal{X}}\beta_k)} \prod_{k\in\mathcal{X}} x_k^{\beta_k - 1}. \quad (8)$$

There are two possible approaches. For the first, the probabilities are estimated and then plugged into the entropy formula. If $\beta_i = N\ \forall i$, this estimator, [4], is equivalent to adding N "pseudo-counts" to each bin i

$$\hat{H}^{Bayes} = \sum_{k\in\mathcal{X}} \frac{nb(x_k)+N}{m+|\mathcal{X}|N} \log \frac{nb(x_k)+N}{m+|\mathcal{X}|N}. \quad (9)$$

The second approach consists of a direct computation of the entropy via

$$\begin{aligned} \hat{H}^{Dir} &= \frac{1}{m+|\mathcal{X}|N}\sum_{k\in\mathcal{X}}(nb(x_k)+N) \quad (10)\\ &\quad (\psi(m+|\mathcal{X}|N+1) - \psi(nb(x_k)+N+1), \end{aligned}$$

where $\psi(z) = \frac{d\ln\Gamma(z)}{dz}$ the digamma function, see [6].
Various values for N have been proposed [4] including $N = \frac{1}{|\mathcal{X}|}$, known as the Schürmann-Grassberger-Dirichlet (SG-Dirichlet) estimator which is used in this paper.

2.1.5. Gaussian

Let now X be a multivariate Gaussian, whose probability density function with mean μ and covariance matrix C is defined as

$$f(X) = \frac{1}{\sqrt{(2\pi)^n |C|}} \exp^{(-\frac{1}{2}(x-\mu)^T C^{-1}(x-\mu))}. \quad (11)$$

The entropy of this distribution is given by

$$H(X) = \frac{1}{2}\ln\{(2\pi e)^n |C|\}, \quad (12)$$

where $|C|$ is the determinant of the covariance matrix, see [7].
The mutual information between two variables X_i and X_j is then given by

$$\begin{aligned} I(X_i, X_j) &= \frac{1}{2}\log\left(\frac{\sigma_{ii}\sigma_{jj}}{|C|}\right) \quad (13)\\ &= -\frac{1}{2}\log(1-\rho^2). \quad (14) \end{aligned}$$

where ρ is the Pearson's correlation.

2.2. Discretization Methods

In order to use the described discrete estimators if the random variable X is continuous, the space $\mathcal{X}$ has to be discretized. The two most used methods for discretization are the equal width and the equal frequency methods [8].
If X takes values in the interval $[a,b]$, the equal width discretization method divides this interval into $|\mathcal{X}|$ subintervals of the same size.
The equal frequency methods also divides the interval $[a,b]$ into $|\mathcal{X}|$ subintervals. Each of these subintervals contains the same number of data points. Therefore the subinterval sizes are likely to be different.

3. R-PACKAGE: MINET

The introduced R-package MINET[1] allows the use of three different inference methods, namely ARACNE, introduced in [9], CLR in [10] and MRNET described in [2].
Furthermore different entropy estimators can be employed for calculating the mutual information. The empirical, the Miller-Madow, the shrink, the SG-Dirichlet and the Gaussian estimator.
The first four estimators require discrete data. Two different discretization methods are implemented in the package, either equal frequency or equal width discretization with default size $\sqrt{|\mathcal{X}|}$.
The inference proceeds in two steps. In the first step, the mutual information matrix is calculated. In the second step, the chosen algorithm is applied to the mutual information matrix in order to compute a score that is used as the weight between the network nodes.
The CLR algorithm computes the mutual information (MI) for all pairs and derives a score related to the empirical distribution of these values. The ARACNE method calculates the MI pairwise for three genes. Eventually the weakest edge of each triplet is removed.

3.1. The MRNET method

The MRNET method, [2], is based on the maximum relevance/ minimum redundancy technique. Among the least redundant variables the one having the highest mutual information with the target is chosen.
The method ranks the set V of inputs according to a score which is the difference between the mutual information with the output variable Y (maximum relevance) and the average mutual information with the previously ranked

[1] http://cran.r-project.org/web/packages/minet

No.	Dataset	source net	n	m
1	ecoli_300_300	E.coli	300	300
2	ecoli_300_200	E.coli	300	200
3	ecoli_300_100	E.coli	300	100
4	ecoli_300_50	E.coli	300	50
5	ecoli_200_300	E.coli	200	300
6	ecoli_200_200	E.coli	200	200
7	ecoli_200_100	E.coli	200	100
8	ecoli_200_50	E.coli	200	50
9	ecoli_100_300	E.coli	100	300
10	ecoli_100_200	E.coli	100	200
11	ecoli_100_100	E.coli	100	100
12	ecoli_100_50	E.coli	100	50

Table 1. Generated datasets. Number of genes n, number of samples m.

variables (minimum redundancy). The network is infered by deleting all edges whose score lies below a given threshold.
Direct interactions should be well ranked whereas indirect interactions should be badly ranked.

4. SIMULATION

4.1. Network generation

In order to compare the results, artificial microarray datasets were generated by the SynTReN generator, [11]. It generates the network topology by selecting subnetworks from E.coli and S.cerevisia source networks. Interaction kinetics are modeled by equations based on Michaelis-Menten and Hill kinetics.
The generator was used to generate twelve datasets. The details of each generated dataset can be found in Table 1 with respect to the number m of samples and the number n of genes. The datasets were generated without noise. The noise was later added to analyze its impact on the network inference.

4.2. Introducing missing values

To study the impact of missing values, missing values were inserted into the generated datasets. The number of missing values is distributed according to the $\beta(a, b)$ distribution with parameters $a = 2$ and $b = 5$. The maximal allowed number of missing values was a third of the entire dataset. This distribution was utilized, instead of the uniform distribution, because the latter one could have favoured the empirical estimator.

4.3. Setup

For each experiment twenty repetitions were carried out. Each dataset was analyzed with the MRNET method using the five available estimators: Gaussian, empirical, Miller-Madow, shrink and SG-Dirichlet. Apart from the Gaussian, all estimators were computed applying both available discretization approaches. Furthermore, the computation was carried out with added Gaussian noise, $N(0, 0.1)$, and without noise. Each of these setups was also assessed with introduced missing values.

4.4. Validation

For each pair of nodes the inference algorithm adds an edge or not. If the added edge is present in the underlying true network, it is considered to be a true positive (TP), if it is not present a false positive (FP). On the other hand, an edge the agorithm did not add and which is not present in the underlying network is called true negative (TN). If this edge is present then it is called false negative (FN). To validate the infered network the precision quantity and the recall quantity, respectively,

$$p = \frac{TP}{TP + FP} \tag{15}$$

$$r = \frac{TP}{TP + FN} \tag{16}$$

have been introduced. The former one measures the fraction of real edges among the ones classified as positive and the latter one the fraction of real edges that are correctly infered.
A weighted harmonic average of precision and recall is given by the F-score [12]

$$F = \frac{2pr}{r + p}. \tag{17}$$

To validate the simulation's results, the maximal F-score was computed for each experiment. Using a paired t-test, the maximal F-scores were then compared and statistically validated.
The results of these calculations are displayed in Table 2.

4.5. Results

In Table 2, the maximal F-scores for each setup are listed, In bold face are the best values and those values that are not significantly different from the best value with respect to each of the four categories based on a p-value less than 0.05. At first, it can be noted that in case of noise, missing values, or both, the Gaussian estimator is the best compared to the other employed estimators. The Gaussian estimator is not strongly influenced by pertubed data. It remains on an average level through all experiments.
Since the Gaussian estimator seems to be more robust to missing data and noise compared to the other applied estimator, it should be utilized in case of a setup with few samples.
In Table 4, it can be observed that in small sample regions the Gaussian estimator is the best for every setup apart from the case with no noise and no missing values.
The next notable observation is the absence of a significant difference between the empirical, Miller-Madow, shrink and the SG-Dirichlet estimator, given that the same discretization procedure is applied to all of them. Furthermore, there is a significant difference between the results if the same estimator, computed with the equal frequency approach, is compared to its counterpart using the equal

Estimator		no noise no NA	noise no NA	no noise NA	noise NA
Gaussian		0.2006	**0.1691**	**0.1790**	**0.1611**
EqF	Emp	**0.3420**	0.1551	0.1136	0.0868
EqF	MM	**0.3396**	0.1524	0.11402	0.0923
EqF	Shr	0.3306	0.1506	0.1150	0.0788
EqF	Dir	**0.3389**	0.1478	0.1057	0.0827
EqW	Emp	0.2028	**0.1650**	0.1036	0.0822
EqW	MM	0.1909	0.1592	0.1068	0.0883
EqW	Shr	0.1935	0.1574	0.1090	0.0839
EqW	Dir	0.2099	0.1592	0.0968	0.0808

Table 2. Results using MINET with inference method MRNET; noise $N(0, 0.1)$, number of missing values maximal one third of the dataset; in bold: maximum F-scores and significantly not different values, based on p-value 0.05.

Estimator	no noise no NA	noise no NA	no noise NA	noise NA
Gaussian	0.1920	**0.1502**	**0.1821**	**0.1483**
Emp	**0.2975**	**0.1521**	0.1111	0.0683
MM	**0.2976**	0.1479	0.1051	0.0687

Table 3. Results for number of genes $n = 300$, samplesize $m = 800$, equal frequency discretization approach.

width discretization approach. It can be observed that using the equal frequency approach is generally better than using the equal width discretization approach.
In case of high sample size and well behaved data with no missing values, the empirical estimator should be used with the equal frequency discretization approach, see Table 3.

5. CONCLUSION

An experimental study on the impact of the influence of different estimators, discretization methods, noise and missing values on network inference has been carried out.
It can be concluded that the usage of the equal frequency method led to higher F-scores.
In case of a high sample number, the empirircal estimator was, with a statistically significant p-value, among the estimators that led to an infered network with the highest F-scores. In case of no noise, the F-scores were significantly higher for the empirical and the Miller-Madow estimator compared to the results from the Gaussian estimator.
Without missing values, the empirical estimator and the other ones led to good results. However, if missing values occur, the Gaussian estimator seems to be more robust. Furthermore, in case of few sample sizes, the Gaussian estimator should be preferred.
For all other estimators, a strong influence of missing values and noise could be observed. The F-scores dropped by a significant amount compared to the results from calculations without any missing values or noise.

Estimator	no noise no NA	noise no NA	no noise NA	noise NA
Gaussian	0.1916	**0.1455**	**0.1767**	**0.1461**
Emp	**0.3653**	0.1286	0.1098	0.0654
MM	**0.3592**	0.1305	0.0996	0.0674

Table 4. Results for number of genes $n = 300$, samplesize $m = 100$, equal frequency discretization approach.

6. REFERENCES

[1] E. van Someren, L. Wessels, E. Backer, and M. Reinders, "Genetic network modelling," *Pharmacogenomics*, vol. 3, pp. 507–525, 2002.

[2] P. E. Meyer, K. Kontos, F. Lafitte, and G. Bontempi, "Information-theoretic inference of large transcriptional regulatory networks," *EURASIP Journal on Bioinformatics and Systems Biology*, 2007.

[3] L. Paninski, "Estimation of entropy and mutual information," *Neural Computation*, 2003.

[4] J. Hausser, "Improving entropy estimation and the inference of genetic regulatory networks," August 2006.

[5] J. Schäfer and K. Strimmer, "A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics," *Statist. Appl. Genet. Mol. Biol.*, 2005.

[6] L. Wu, P. Neskovic, E. Reyes, E. Festa, and W. Heindel, "Classifying n-back eeg datta using entropy and mutual information features," in *ESANN*, 2007.

[7] S. Haykin, *Neural Networks: A Comprehensive Foundation*, Prentice Hall International, 1999.

[8] J. Dougherty, R. Kohavi, and M. Sahami, "Supervised and unsupervised discretization of continuous features," *ICML*, 1995.

[9] A. Margolin, I. Nemenman, and K. B. et al, "Aracne: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context," *BMC Bioinformatics*, vol. 7, 2006.

[10] J. Faith, B. Hayete, and J. T. et al, "Large-scale mapping and validation of escherichia coli transcriptional regulation from a compendium of expression profiles," *PLoS Biology*, vol. 5, 2007.

[11] T. V. den Bulcke, K. V. Leemput, B. Naudts, P. van Remortel, H. Ma, B. D. Moor, and K. Marchal, "Syntren: a generator of synthetic gene expression data for design and analysis of structure learning algorithms," *BMC Bioinformatics*, 2006.

[12] M. Sokolova, N. Japkowicz, and S. Szpakowicz., "Beyond accuracy, f-score and roc: a family of discriminant measures for performance evaluation," in *Proceedings of the AAAI'06 workshop on Evaluation Methods for Machine Learning*, 2006.

A RETENTION-TIME ALIGNMENT ALGORITHM FOR LC/MS DATA

Katharina Podwojski[1,2*], *Arno Fritsch*[1,2*], *Daniel Chamrad*[3], *Wolfgang Paul*[4,2], *Petra Mutzel*[4,2], *Katja Ickstadt*[1,2], *Jörg Rahnenführer*[1,2]

[1]Fakultät Statistik, Technische Universität Dortmund, 44221 Dortmund, Germany
[2]Zentrum für Angewandte Proteomik (ZAP), Dortmund, Germany
[3]Protagen AG, Otto-Hahn-Str. 15, 44227 Dortmund, Germany
[4]Fakultät für Informatik, Technische Universität Dortmund, 44221 Dortmund, Germany
katharina.podwojski@uni-dortmund.de, fritsch@statistik.uni-dortmund.de

ABSTRACT

Liquid chromatography coupled to mass spectrometry (LC/MS) has advanced to a leading technology for the analysis of complex protein mixtures. Typical quantitative proteomic studies aim at detecting differentially expressed peptides between different proteomes. Thus the combination of several LC/MS maps is a crucial step in a typical analysis workflow. Nonlinear shifts in the retention time between LC/MS maps make this a nontrivial task. We have developed a statistical two-step algorithm for the retention-time alignment of a large number of LC/MS maps. First a clustering procedure detects well-behaved compounds. Afterwards these compounds are used to calculate a non-linear deviation curve for each map. We evaluated our algorithm through a simulation study. After the alignment most compounds are correctly assigned.

1. INTRODUCTION

The method of LC/MS is highly suitable for the analysis of complex protein or peptide mixtures as it allows good separation, high sensitivity and easy automation. Thus LC/MS can be used as a high-throughput method.
For an LC/MS experiment, first the protein mixture is digested into smaller peptides. These peptides are afterwards separated by the LC-column, typically according to their hydrophobicity. The peptides elute from the column according to their chemical properties. The resulting fractions are afterwards inserted in the mass spectrometer where the peptides in each fraction are further separated by their mass to charge (m/z). The data can then be presented in a 2-D or 3-D map, where the elution or retention time is represented on the first axis, the m/z value on the second axis, and the abundance is either depicted by color or on the third axis.
One possible problem that may be tackled with LC/MS experiments is the detection of differentially expressed proteins or peptides in different samples. This is of special interest in clinical studies aiming at detecting differences in the proteome of healthy and diseased patients. The hope is to find biomarkers that may then be used either for diagnostics or as potential drug targets.
Typically a series of pre-processing steps is needed before the differential analysis can take place [1, 2]. These steps can be calibration, noise reduction, normalization, peak detection and quantification. Sometimes single peaks are further combined according to the isotopic distributions of peptides or different charge states. Pre-processing generates a so-called feature or compound map for each LC/MS map.
After pre-processing of single LC/MS maps the combination of several LC/MS maps is necessary for performing differential studies. Due to variation both in retention time and in m/z values it is hard to compare different LC/MS maps. Aligning corresponding compounds is already difficult when the same sample is measured multiple times. The deviations between LC/MS maps are even larger when different samples are measured. While differences in m/z values, which depend on the type of mass spectrometer, are usually small and can be handled easily, differences in retention time between two LC/MS runs can be quite large, from several seconds up to several minutes. Even worse, the type of deviation across the retention-time axis often is nonlinear.
Hence, a lot of effort was spent on the construction of algorithms that correct for the differences in retention time. There are several approaches that may be applied at different steps during pre-processing. Early methods presented in the literature only work on total ion chromatograms (TIC) that measure the total ion count in each fraction of the LC [3, 4]. Other alignment algorithms are applied to the complete raw 2-D images derived from LC/MS maps, see for example [5]. Recently, also algorithms for alignment of compound maps have been proposed [6, 7]. For a

* Both authors contributed equally to this work.

recent overview of alignment methods see [8].
The paper is organized as follows. In section 2.1 the proposed new alignment procedure is briefly explained. Then, a simulation study for evaluating the performance of the algorithm is introduced. In section 3 the results of the alignment algorithm are shown and discussed. Finally, a discussion and outlook on future work is presented.

2. METHODS

Often the goal in high-throughput LC/MS experiments is to detect clinically relevant differences in the proteome. A suitable retention-time alignment algorithm thus needs to be able to cope with a large number of LC/MS maps corresponding to a sufficiently large number of patients. Hence, an alignment algorithm based on the complex raw data is typically not feasible. Aside from the size of the datasets the noise still present in the raw LC/MS maps makes it hard to align a large number of maps. On the other hand, alignment algorithms based on TIC ignore too much of the information that becomes visible only in the 2-D LC/MS maps. Thus the retention-time algorithm we propose works on compound maps that still comprise much of the 2-D information but are less complex and less noisy than the raw maps.

2.1. Alignment

Briefly, our new retention-time alignment algorithm works as follows. First, a simple alignment based mainly on mass (m/z values) is performed and groups of compounds are identified that can be well aligned. Second, these groups are then used for estimating a nonlinear retention-time deviation curve for each sample.
Usually, in complex protein mixtures from the same source, for example samples obtained from serum or urine, there are a couple of highly abundant peptides that can be identified in every LC/MS map. These compounds can be easily aligned. State-of-the-art algorithms fit linear transformations in retention time [6]. This is in disagreement with our experience that often much more complicated deviations between LC/MS maps can be observed. These deviations can not be fitted with linear or even quadratic functions. Furthermore, as our algorithm is intended to be used in clinical settings, there may be large time intervals between the measurements of different samples. In such cases deviations in retention times will tend to be even larger and become more complex.
The input of the alignment algorithm are n compound maps. The simple alignment in the first step of our algorithm is based on a hierarchical average-linkage cluster analysis applied to the compounds from all n combined maps with the highest intensities in a fixed mass-window. Knowledge about the mass precision of the spectrometer is used to determine coherent groups. From this preliminary alignment 'well-behaved' groups are picked. Such a group must contain compounds from a predefined minimum number of different of the n runs and at the same time is not allowed to contain more than one compound from the same run. The mass window is iteratively shifted and the procedure is repeated. A slight overlap between windows is used in order to avoid the splitting of potentially well-behaved groups. The result of this step are groups of aligned compounds ranging across the whole mass domain. In analogy to [9], for each run, the retention-time deviation for a single compound is calculated as the difference between the retention time of this compound and the median of the corresponding retention times for all runs in the same well-behaved group. This procedure generates for every run a two-dimensional plot with retention on the x-axis and deviation from the median retention time on the y-axis. A two-dimensional scatterplot smoother can then be used to estimate a smooth function of retention-time shift along the time axis. For this purpose we fit a locally weighted regression (loess) curve to the retention-time deviations for each LC/MS run.
An advantage of the algorithm is that it can be applied to the combined dataset of all compounds from all LC-MS runs, since no master dataset has to be chosen to which all other maps have to be aligned. The fitted curves are then used to correct the shifts in retention time by simply subtracting these estimated differences over time.
The retention-time alignment algorithm is implemented in the statistical programming language R [10].

2.2. Simulation Study

To evaluate the performance of our alignment algorithm we use a simulated dataset. This dataset is based on a real LC/MS experiment of *E. Coli* samples. We have combined five LC/MS runs of this *E. Coli* experiment to obtain a complex compound map (with more than 35.000 compounds). We then assume that this map corresponds to one true experiment, containing all compounds present in the sample. Based on this map 20 LC/MS runs are simulated in the following way.
First, a subset of all compounds is chosen for each LC/MS run. This step mimics the problem, that in reality not all compounds of a sample are found in every LC/MS run. Each of the runs finally contains about 10.000 compounds and thus can be seen as a highly complex sample as observed in a real LC/MS experiment in which a sample of serum of urine is used. Then, each compound is varied randomly in retention time, in mass to charge (m/z) and in intensity. The retention-time variations consist of a time offset, a typical retention-time curve and an additive typical random error. The typical retention-time curves and the random errors are estimated from a basic experiment where one single peptide (phosphorylase b) is digested and measured with LC/MS. Due to the small amount of compounds in the single LC/MS runs for this basic experiment, the different compounds can be matched visually across runs and thus the retention-time deviation curves can be estimated with high precision.
Our algorithm is applied to the simulated datasets. This controlled scenario allows the objective evaluation of the ability to identify corresponding proteins and to correctly estimate shifts in retention time.

Table 1. Distribution of standard deviations of retention time of compounds across runs, in minutes.

	Minimum	1st Quartile	Median	Mean	3rd Quartile	Maximum
before alignment	0.0000704	0.3608	0.4815	0.4730	0.5775	1.9339
after alignment	0.0000185	0.0251	0.0356	0.0568	0.0500	1.1460

3. RESULTS

We present the results of the application of our new alignment algorithm to the 20 simulated LC/MS maps. Originally, retention-time deviations between different runs are as large as up to 3.5 minutes. The variations in m/z are maximally of size $\pm$ 50 parts-per-million (ppm). This is quite high compared to mass accuracies achieved by modern mass spectrometers. Thus we can check if the algorithm even works when the mass accuracy is low.

The alignment algorithm is able to find a large number of 'well-behaved' groups throughout the retention-time range. Thus an alignment curve can be fitted to every LC/MS run. Two typical curves are shown in Figure 1. The alignment curves for the other simulated LC/MS runs look similar.

We evaluated the performance of the alignment by comparing the deviations between retention times of corresponding compounds in different maps. In the simulation study, only a subset of the original full compound list is chosen for each simulated LC/MS run. Therefore most compounds are only present in a subset of the 20 simulated LC/MS runs. We thus only use those compounds for evaluation that are present in at least two simulated maps. As only about 4.000 of the over 35.000 compounds are present in only one run, enough compounds are left for the evaluation. For the compounds present in at least two maps the standard deviation in retention time across all runs is calculated, once before and once after the alignment.

The results are shown in Table 1 and in Figures 2 and 3. In Figure 2 logarithms of standard deviations are plotted for better visibility. It can be seen that the alignment very well corrects the original retention-time deviations. Only few outliers still exhibit a large deviation across different runs, mostly due to high random errors that were added to the data during the simulation. The corresponding compounds can not be aligned correctly any more as the values only represent noise in the data.

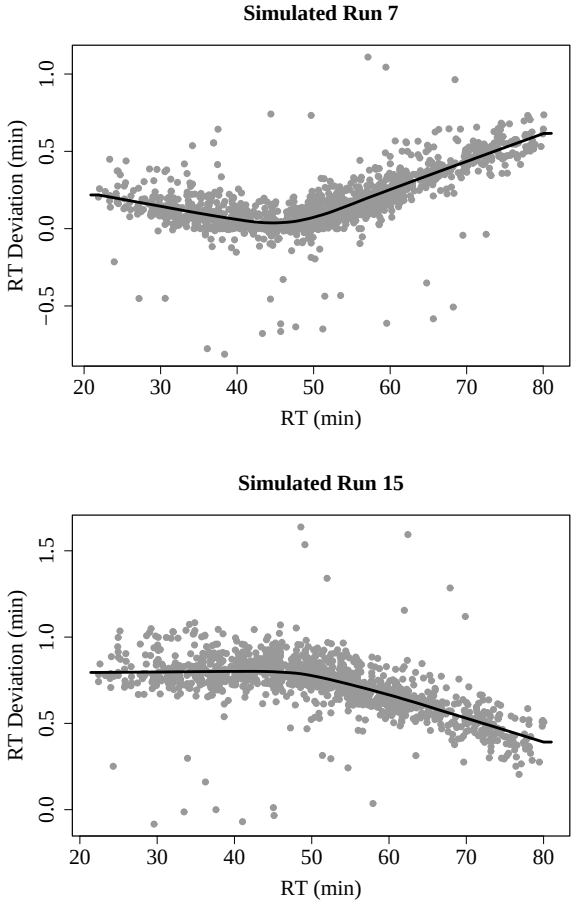


Figure 1. Retention time (RT) alignment curve for one simulated LC/MS run. Each point depicts a compound in a 'well-behaved' group with its retention time and deviation from the median retention time in minutes in the corresponding compound group.

4. CONCLUSION AND OUTLOOK

We have proposed a retention-time alignment algorithm for LC/MS data that works on compound maps. We have evaluated our algorithm on simulated data. The algorithm is able to correct highly nonlinear retention-time deviations as often seen in real LC/MS experiments.

The algorithm is currently also being evaluated on real LC/MS data. First applications both on spike-in LC/MS data and on a differential study with stimulated cells have shown promising results. In these applications it is necessary to bin together compounds from different runs that represent the same peptide in order to then perform differential analyses. With this additional binning step one can cope with some of the remaining variations in retention time after alignment. Thus a perfect alignment that maps corresponding compounds exactly onto the same retention time is not needed. An evaluation of a combined alignment and binning approach is in progress.

Our results can be validated with the help of MS/MS experiments that can be used for peptide identification. This will enable an evaluation that checks if the right compounds have been binned together after the alignment. In comparison with other state-of-the art alignment algorithms the algorithm shows a competitive behavior.

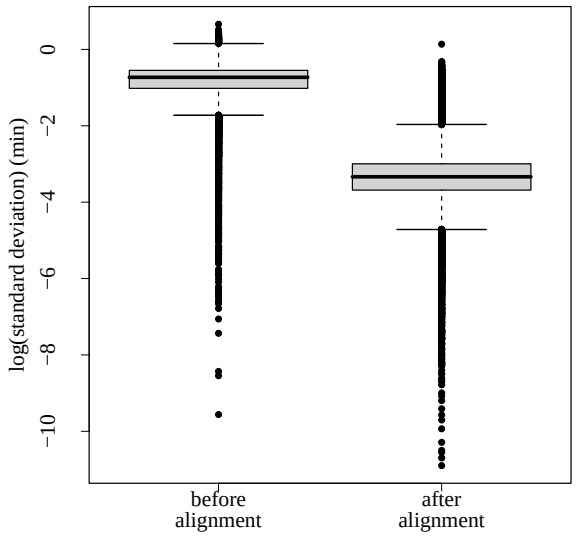


Figure 2. Boxplot of log(standard deviations) of retention time of compounds across runs, in minutes.

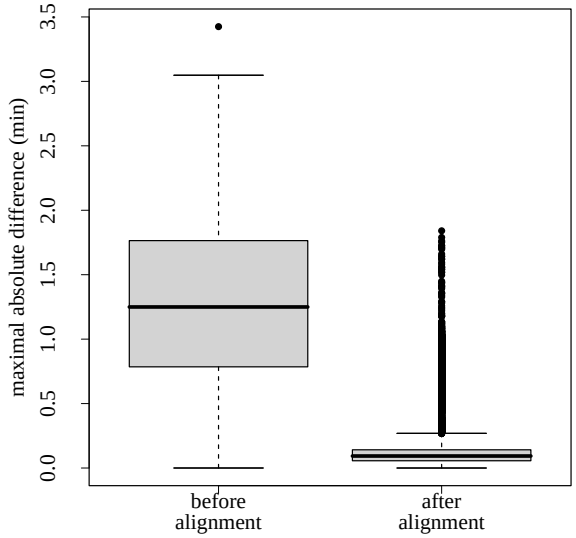


Figure 3. Boxplot of maximum absolute difference of retention time of compounds across runs, in minutes.

5. ACKNOWLEDGMENTS

We thank Barbara Sitek from the Center of Applied Proteomics (ZAP) and Carsten Bäßmann and his group from Bruker Daltonics for providing the real LC/MS datasets used in the simulation study. We thank Christian Stephan from the Medical Proteome Center in Bochum for helpful and stimulating discussions.
KP, AF, WP, PM, KI, and JR kindly acknowledge funding by the European Union and the state of North Rhine-Westphalia.

6. REFERENCES

[1] D. Radulovic, S. Jelveh, S. Ryu, T. Hamilton, E. Foss, Y. Mao, and A. Emili, "Informatics platform for global proteomic profiling and biomarker discovery using liquid chromatography-tandem mass spectrometry," *Molecular and Cellular Proteomics*, vol. 3, pp. 984 – 997, 2004.

[2] J. Listgarten and A. Emili, "Statistical and computational methods for comparative proteomic profiling using liquid chromatography-tandem mass spectrometry," *Molecular and Cellular Proteomics*, vol. 4, pp. 419 – 434, 2005.

[3] G. Tomasi, F. van den Berg, and C. Andersson, "Correlation optimized warping and dynamic time warping as pre-processing methods for chromatographic data," *Journal of Chemometrics*, vol. 18, pp. 231 – 241, 2004.

[4] A. van Nederkassel, M. Daszykowski, P. Eilers, and Y. V. Heyden, "A comparison of three algorithms for chromatographic alignment," *Journal of Chromatography A*, vol. 1118, pp. 199 – 210, 2006.

[5] J. Listgarten, R. Neal, S. Roweis, P. Wong, and A. Emili, "Difference detection in lc-ms data for protein biomarker discovery," *Bioinformatics*, vol. 23, pp. e198 – e204, 2006.

[6] E. Lange, C. Gröpl, O. Schulz-Trieglaff, A. Leinenbach, C. Huber, and K. Reinert, "A geometric approach for the alignment of liquid chromatography-mass spectrometry data," *Bioinformatics*, vol. 23, pp. i273 – i281, 2007.

[7] P. Wang, H. Tang, M. Fitzgibbon, M. McIntosh, M. Coram, H. Zhang, E. Yi, and R. Aebersold, "A statistical method for chromatographic alignment of lc-ms data," *Biostatistics*, vol. 8, pp. 357 – 367, 2007.

[8] M. Vandenbogaert, S. Li-Thiao-T, H.-M. Kaltenbach, R. Zhang, T. Aittokallio, and B. Schwikowski, "Alignment of lc-ms images, with applications to biomarker discovery and protein identification," *Proteomics*, vol. 8, pp. 650 – 672, 2008.

[9] C. Smith, W. Want, G. O'Maille, R. Abagyan, and G. Siuzdak, "Xcms: Processing mass spectrometry data for metabolite profiling using nonlinear peak alignment, matching, and identification," *Analytical Chemistry*, vol. 78, pp. 779 – 787, 2006.

[10] R Development Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2006, ISBN 3-900051-07-0.

RATE VARIATIONS, PHYLOGENETICS, AND PARTIAL ORDERS

Sonja J. Prohaska[1,2], *Guido Fritzsch*[3,4], *and Peter F. Stadler*[5,1,2,4,6]

[1]Santa Fe Institute, 1399 Hyde Park Rd., Santa Fe NM 87501, USA
[2]Department of Theoretical Chemistry, University of Vienna,
Währingerstraße 17, A-1090 Wien, Austria;
[3]Institute of Biology II: Zoologie, Molekulare Evolution und Systematik der Tiere,
University of Leipzig, Talstrasse 33, D-04103 Leipzig, Germany
[4]Interdisciplinary Center for Bioinformatics, and
[5]Bioinformatics Group, Department of Computer Science
University of Leipzig, Härtelstraße 16-18, D-04107 Leipzig, Germany
[6]RNomics Group, Fraunhofer Institut for Cell Therapy and Immunology (IZI),
Deutscher Platz 5e, D-04103 Leipzig, Germany
sonja@santafe.edu, gfritz@rz.uni-leipzig.de, studla@bioinf.uni-leipzig.de

ABSTRACT

The systematic assessment of rate variations across large datasets requires a systematic approach for summarizing results from individual tests. Often, this is performed by coarse-graining the phylogeny to consider rate variations at the level of sub-claded. In a phylo-geographic setting, however, one is often more interested in other partitions of the data, and in an exploratory mode a pre-specified subdivision of the data is often undesirable. We propose here to arrange rate variation data as the partially ordered set defined by the significant test results.

1. INTRODUCTION

Rate variations are an important source of information in evolutionary biology. Typically, one devises so-called relative-rate tests (RRTs) for statistically significant rate variations between two species [1, 2, 3, 4] or between subgroups of species [5, 6]. Group tests, however, require an initial hypothesis about which species to summarize. In particularly in an exploratory phase this is typically undesirable, since rate variations can be associated with many very different mechanisms, for clade-specific changes in mutation rates to differences in population structure.

In this contribution we therefore introduce an explorative approach to summarizing the results of many pairwise RRTs. The basic idea is to arrange the individual statistically significant pair-wise test results in a partially ordered set. Inspection of the Hasse diagram of this graph can then be used to identify systematic rate variations. In particular, this approach has the potential to highlight systematic rate variations even if they do not conform to a phylogenetic tree but correlate with other variables, such as migratory history.

2. RELATIVE RATE PO-SET

2.1. Po-Sets

Recall that a partially ordered set, *po-set* for short, is a set X together with a relation $\preceq$ satisfying
(P0) $x \preceq x$.
(P1) $x \preceq y$ and $y \preceq x$ implies $x = y$.
(P2) $x \preceq y$ and $y \preceq z$ implies $x \preceq z$.
A finite po-set $(X, \preceq)$ can be respresented as directed acyclic graph G (by drawing an arc $x \leftarrow y$ whenever $x \preceq y$ and $x \neq y$). The Hasse diagram of G is the subgraph H of G with the same vertex set X, and an arc $x \to y$ if $x \to y$ is an arc in G and there is no $z \neq x, y$ such that z lies on a directed path from x to y in G.

2.2. Substitution Rates

Let $\mathcal{X}$ be a set a taxa, which we represent here by their (aligned) nucleic acid or peptide sequences of length n. Furthermore, let $\mathfrak{T}$ be the underlying phylogenetic tree. Each interior vertex w of the tree can be specified as the *last common ancestor* $w = \mathrm{lca}(A, B)$ of two of the descents A and B of w so that the path connecting A and B runs through w.

The Hamming distance $d_{AB} = |\{i|A_i \neq B_i\}|$ counts the positions i in which the characters of the sequences differ. Now consider a triple (A, B, C) of sequences. The quantities

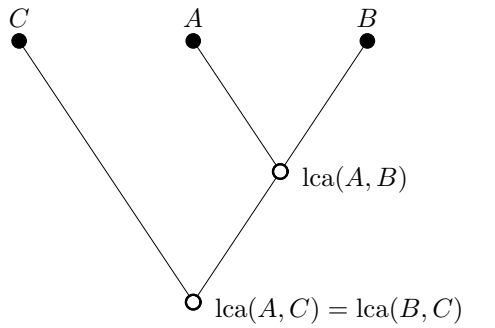
$$\begin{aligned} a_{ABC} &= |\{i|A_i = B_i = C_i\}|, \\ m_{AB|C} &= |\{i|A_i = B_i \neq C_i\}|, \\ m_{AC|B} &= |\{i|A_i = C_i \neq B_i\}|, \\ m_{BC|A} &= |\{i|B_i = C_i \neq A_i\}|, \\ w_{ABC} &= |\{i|A_i \neq B_i \neq C_i \neq A_i\}| \end{aligned} \tag{1}$$

distiguish five classes of alignment positions: (i) constant positions, (ii) positions in which all three sequence differ and (iii) three classes of positions in which two sequences are the same and the third one ins different.

The Hamming distance d_{AB} can be decomposed into three different components w.r.t. to a third sequence C. These correspond to the sequence position where C agrees with B (but not with A), the positions where C agrees with A (but not with B), and those where all three sequences differ:

$$d_{AB} = m_{BC|A} + m_{AC|B} + w_{ABC} \tag{2}$$

Now consider a subtree of $\mathfrak{T}$ consisting of three taxa A, B, C so that C is an outgroup to A and B:



(3)

Let us denote by a and b the lengths of branches between A, B and $\mathrm{lca}(A, B)$, respectively. We have

$$\begin{aligned} 2a = d_{AC} + d_{AB} - d_{BC} &= 2m_{BC|A} + w_{ABC} \\ 2b = d_{BC} + d_{AB} - d_{AC} &= 2m_{AC|B} + w_{ABC} \end{aligned} \tag{4}$$

and hence

$$a - b = m_{BC|A} - m_{AC|B}\,. \tag{5}$$

Note that $m_{BC|A}$ and $m_{AC|B}$ count independent sequence positions, while the Hamming distances are dependent via the common term w_{ABC}. Equ.(5) is the basis of Tajima's relative rate test [2], while the older Wu & Li test [3] uses the difference $d_{AC} - d_{BC}$. Alternatively, one might want to employ a suitable maximum likelihood test to assess the significance of branch length differences [1, 4].

We can estimate the relative rate of evolution along the branches a and b for those comparisons that are statistically signficant according to the relative rate test of choice. In the following, it will be more convenient to use the following logarithmic measure

$$\eta_{AB} = \begin{cases} \ln \frac{a}{b} & \text{if } a - b \text{ is statistically significant} \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

Next we show that for ideal data we do not have to fear contradictory results of relative rate tests involving different triples of taxa selected from the tree $\mathfrak{T}$. Recall that the distances d_{AB} of leafs A and B in a additive metric tree $\mathfrak{T}$ are defined as the sum of the lengths of the edges along the unique path that connects A and B in $\mathfrak{T}$.

More precise, we have the following

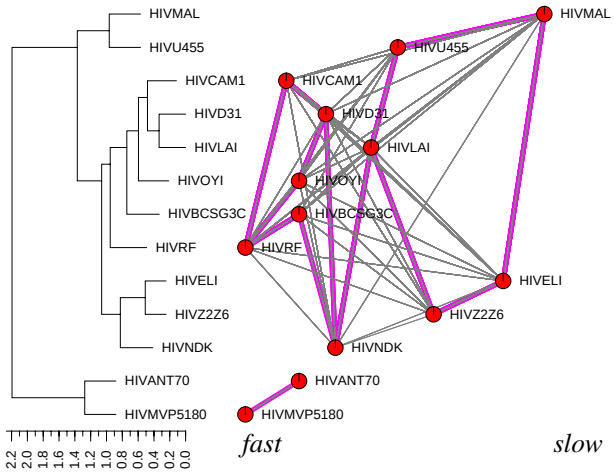
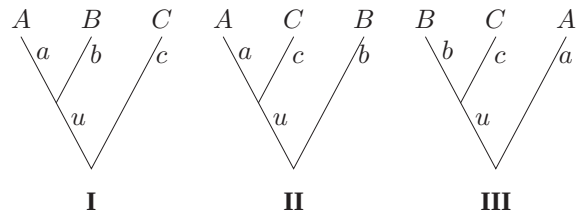


Figure 1. Example of a relative rate poset. Data are 5'UTRs of HIV-1. Thin lines in the r.h.s. panel indicate significant Tajima tests, the thick lines represent the associated Hasse diagram of the partially ordered set.

Theorem 1. *The directed graph associated with $\boldsymbol{\eta}$ is acyclic provided d is an additive tree metric on $\mathcal{X}$.*

Proof. First, we observe that $\boldsymbol{\eta}$ is antisymmetric by construction, $\eta_{AB} = -\eta_{BA}$. Thus there are no cycles of length 2. Next assume $\eta_{AB} > 0$ and $\eta_{BC} > 0$. We have to consider the following three cases



Translating the assumption in inequalities of branch lengths in each of the three cases yields:

- **(I)** $a > b$ and $b + u > c$ implies $a + u > c$, i.e., $\eta_{AC} \geq 0$.
- **(II)** $a + u > b$ and $b > c + u$ implies $a > c$, i.e., $\eta_{AC} \geq 0$.
- **(III)** $a > b + u$ and $b > c$ implies $a > c + u$, i.e., $\eta_{AC} \geq 0$.

These three inequalities for η_{AC} assume that the underlying statistical test is "sane" in the sense that it never returns a significantly larger rate for the short branch. Thus $\eta_{AB} > 0$ and $\eta_{BC} > 0$ always implies $\eta_{AC} \geq 0$. Now consider a chain of taxa $\{A^j | 1 \leq j \leq m\}$ such that $\eta_{A^{j-1}A^j} > 0$ for $2 \leq j \leq m$. By repeated application of the this result we conclude $\eta_{A^k,A^l} \geq 0$ for any $l > k$, i.e., the $\{A^j\}$ cannot be part of a directed cycle. Since there is an edge from node i to node j iff $\eta_{i,j} > 0$, we conclude that the corresponding graph is a DAG, and hence the matrix $\boldsymbol{\eta}$ is acyclic. □

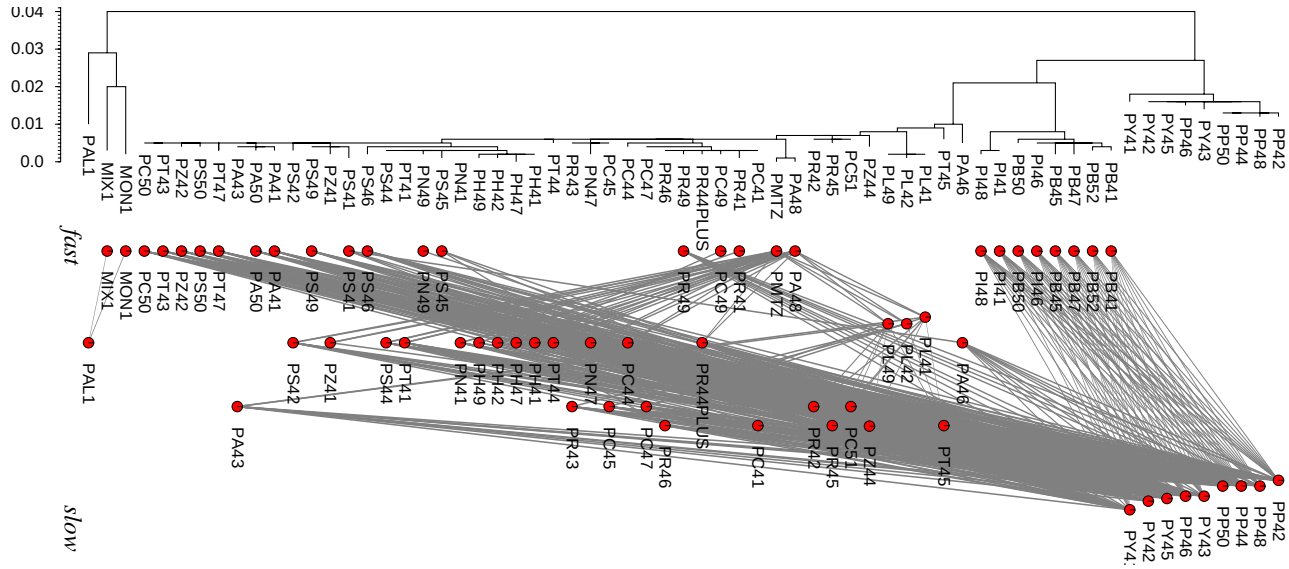


Figure 2. Phylogenetic tree (neighbor joining) and Hasse diagram of the relative-rate poset of mtND1 nucleotide sequence data of wolf spiders of the *Pardosa saltuaria* group [7]. Significance level for Tajima tests $p \leq 0.1$ ($\chi^2 = 2.706$), test results of all subtrees included. Labels refer to geographic locations: North/South Scandinavia PN, PS; Eastern/Western Riesengebirge PC, PR; Tatra Mountains PT; Alps PA, PL, PZ; Eastern/Western Pyrenees PP, PY; Balkans PB, PI; Bohemia PH; Lago di Garda area PMTZ. Outgroup: *P. palustris* PAL, *P. monticola* MON1, *P. mixta* MIX.

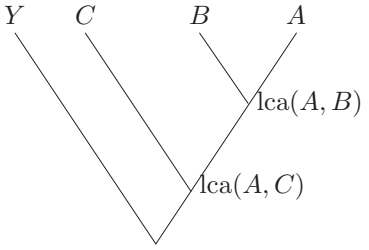
In order to work with real data, we have to relax the assumption that d is an additive tree metric. The estimates for a and b will then depend explicitly on the outgroup C. Note, however, that these variations are small as long as the data are at least approximately tree-like. We can therefore estimate η_{AB} as an *average* over all those triples (A, B, C) for which the Tajima test demonstrates a significant rate difference. The χ^2 value obtained from the Tajima test can be used as weight of the individual estimates. Numerically, we observe that $\boldsymbol{\eta}$ is indeed acyclic even when small χ^2 significance thresholds for the Tajima test are used.

The construction of the matrix $\boldsymbol{\eta}$ starting from a sequence alignment using Tajima's relative rate test has been implemented in a software prototype. It either uses a phylogenetic tree $\mathfrak{T}$ as additional input, or tests for all triples (A, B, C) with outgroup C if $d_{AC}, d_{BC} > d_{AB}$. In order to facilitate the interpretation of the data, it produces a graphical out that compares the phylogenetic tree with the Hasse diagram of the po-set derived from $\boldsymbol{\eta}$, Fig. 1. Points are positioned so that differences along the rate-axis are approximately proportional to differences in $\boldsymbol{\eta}$-values.

2.3. Loss of Phylogenetic Footprints

Relative rate tests can also be designed for more complex settings than substitution rates in homologous sequences. For example, the quantitative analysis of dynamical aspects of footprint loss and acquisition is complicated by the fact that individual regulatory DNA regions cannot be observed independently of sequence conservation. The reason is that phylogenetic footprinting [8, 9, 10, 11] always detects regulatory elements in (at least) pairs of sequences. As a consequence, even very simplistic models of footprint loss lead to rather sophisticated inference.

In the approach proposed in [12], *two* outgroups are required to first identify conserved sequence positions, before one tests for differential loss rates among two ingroup species. More precisely, consider a sub-tree of the following form:



(7)

Restricting the sequences to those positions for which $Y_i = C_i$ holds, we define

$$\begin{aligned} c_{CA} &= |\{i|Y_i = C_i = A_i\}|, \\ c_{CB} &= |\{i|Y_i = C_i = B_i\}|, \\ c_{CAB} &= |\{i|Y_i = C_i = A_i = B_i\}|\,. \end{aligned} \tag{8}$$

Note that $c_{CA} \geq c_{CAB}$ and $c_{CB} \geq c_{CAB}$ always holds. The number of conserved positions exclusively lost along the edge $A, \mathrm{lca}(A, B)$ is $m'_A = c_{CB} - c_{CAB}$ and similarly, for $B, \mathrm{lca}(A, B)$ we have $m'_B = c_{CA} - c_{CAB}$. One now tests whether m'_A and m'_B are significantly different. The corresponding matrix $\boldsymbol{\eta}$ has entries $\eta_{AB} = \ln(m'_A/m'_B)$ provided the difference is statistically signficant, and $\eta_{AB} = 0$, otherwise. For a fixed combination of outgroups Y, C, we immediately check that $m'_A - m'_{A'} > 0$ and $m'_{A'} - m'_{A''} > 0$ implies $m'_A - m'_{A''} > 0$. We therefore expect $\boldsymbol{\eta}$ to be acyclic. Since the choice of a different outgroup pair may lead to the selection of different conserved position, we cannot logically rule out contradictory

test results in this case, however. The implementation of this test is currently in progress.

3. EXAMPLE

The expansion of a species in a heterogeneous environment can be correlated with relative rates of evolution in geographically separated subpopulations. The rate variation may be due to adaptation to different environmental conditions and due to changes in population size or structure [13]. Slowly evolving populations are typically large and stable, while small unstable populations exhibit higher evolution rates. Multiple waves of migration thus may lead to rate variations that show little correlation with phylogenetic position.

As an example of a real-life data set we consider here a recent comprehensive European-wide phylogeographical study of the arctic-alpine distribution of wolf spiders of the *Pardosa saltuaria* group [7]. The data, mitochondrial ND1 gene sequences, show a complex picture of rate differences, with some clear regularities.

For instance, the substitution rates are increased in almost all lineages relative to the samples from the the Pyrenees. This suggests that the Pyrenees served as glacial refugia. The rate correlation between the sequences of the Pyrenees and the Balkan individuals indicates a secound glacial refugium in the Balkan mountains. However, the data indicate migration out of the Pyrenees refugia only. The data set also reflects one further cold period with refugia in the Alps, Sudeten Mountains, and the Upper Tatra.

4. DISCUSSION

We have introduced here an a convenient way to visualize and summarize information on significant rate differences across larger phylogenetic data sets. The poset-approach seems convenient for the exploratory phase of data analysis. As it stands our tool does not attempt to correct for multiple testing, although a strategy such as Bonferroni's correction could easily be incorporated. We also note that the $\mathcal{O}(N^3)$ RRTs that can be performed within a given tree are of course not independent from each other. It might therefore be desirable to restrict attention to a less redundant set of tests.

5. ACKNOWLEDGMENTS

This work was supported in part by the DFG Bioinformatics Initiative and the 6th Framwork Programme of the European Union as part of the EDEN project (contract no. 043251).

6. REFERENCES

[1] J. Felsenstein, "Phylogenies from molecular sequences: inference and reliability," *Annu. Rev. Genet.*, vol. 22, pp. 521–565, 1988.

[2] F. Tajima, "Simple methods for testing molecular clock hypothesis," *Genetics*, vol. 135, pp. 599–607, 1993.

[3] C.-I. Wu and W.-H. Li, "Evidence for higher rates of nucleotide substitution in rodents than in man," *Proc. Natl. Acad. Sci. USA*, vol. 82, pp. 1741–1745, 1985.

[4] Z. Yang, "Maximum-likelihood models for combined analyses of multiple sequence data," *J. Mol. Evol.*, vol. 42, pp. 587–596, 1996.

[5] P. Li and J. Bousqet, "Relative-rate test for nucleotide substitutions between two lineages," *Mol. Biol. Evol.*, vol. 9, pp. 1185–1189, 1992.

[6] M. Robinson, M. Gouy, C. Gautier, and D. Mouchiroud, "Sensitivity of relative-rate tests to taxonomic sampling," *Mol. Biol. Evol.*, vol. 15, pp. 1091–1098, 1998.

[7] C. Muster and T. U. Berendonk, "Divergence and diversity: lessons from an arctic-alpine distribution (*Pardosa saltuaria* group, lycosidae)," *Mol. Ecol.*, vol. 15, pp. 2921–2933, 2006.

[8] D. A. Tagle, B. F. Koop, M. Goodman, J. L. Slightom, D. L. Hess, and R. T. Jones, "Embryonic epsilon and gamma globin genes of a prosimian primate (galago crassicaudatus). Nucleotide and amino acid sequences, developmental regulation and phylogenetic footprints," *J. Mol. Biol.*, vol. 203, pp. 439–455, 1988.

[9] C. Mayor, M. Brudno, J. R. Schwartz, A. Poliakov, E. M. Rubin, K. A. Frazer, L. S. Pachter, and I. Dubchak, "`VISTA`: visualizing global DNA sequence alignments of arbitrary length," *Bioinformatics*, vol. 16, pp. 1046–1047, 2000.

[10] M. Blanchette and M. Tompa, "Discovery of regulatory elements by a computational method for phylogenetic footprinting," *Genome Research*, vol. 12, pp. 739–748, 2002.

[11] S. Prohaska, C. Fried, C. Flamm, G. Wagner, and P. F. Stadler, "Surveying phylogenetic footprints in large gene clusters: Applications to Hox cluster duplications," *Mol. Phyl. Evol.*, vol. 31, pp. 581–604, 2004.

[12] G. P. Wagner, C. Fried, S. J. Prohaska, and P. F. Stadler, "Divergence of conserved non-coding sequences: Rate estimates and relative rate tests," *Mol. Biol. Evol.*, vol. 21, pp. 2116–2121, 2004.

[13] C. Stringer and R. McKie, *African Exodus: The Origins of Modern Humanity*, J.Macrae/H.Holt, New York, 1996.

SELECTIVE ADVANTAGES OF STOCHASTIC PHENOTYPIC DETERMINATION IN UNPREDICTABLE ENVIRONMENTS.

Andre S. Ribeiro[1], John J. Grefenstette[2], Daniel Cloud[3], Antti Hakkinen[1], Tiina Rajala[1] and Olli Yli-Harja[1]

[1]Computational Systems Biology Research Group, Institute of Signal Processing, Tampere University of Technology, P.O. Box 553, FI-33101 Tampere, Finland.
[2]Dept. of Bioinformatics and Computational Biology, George Mason University, USA.
[3]Dept. of Philosophy, Princeton University, USA.

ABSTRACT

We investigate the fitness of stochastic mechanisms of phenotypic determination, in a model where cells' reproduction is fitness dependent, within a variable environment. Comparing the fitness of cells with stochastic and deterministic phenotypes, we find that only when cells can detect environmental conditions does stochasticity provide better fitness. We analyze the result's dependency on environment variability rate and biases. Next, we evolve the rate of phenotypic state transition via mutations in various conditions, and show correlations between phenotypic variability and environmental variability. The results provide insights on how cells may use stochastic mechanisms to face environmental changes and how maintenance of latent phenotypes may provide higher fitness.

1. INTRODUCTION

Populations of genetically identical individuals have a wide diversity of phenotypes [1]. Stochasticity in gene expression and environmental conditions are two known sources of variability [1]. Physiologic noise is present in most biological processes such as gene expression [3], interactions between proteins, cells, tissues, and between organisms and the environment [2]. Complex phenotypic traits, such as some behaviors, depend on the interaction between gene expression and environment [5].

Some phenotypes are only expressed in certain conditions. The phenotypic variability of a population can be produced by genetic changes, but also without these changes (e.g., via methylation) [4]. *B. subtilis* has probabilistic and transient differentiation, dependent on the environment [7]. Noise in ComK expression, the protein that regulates competence for DNA uptake, causes cells to transit to the competent state. Experimental reduction of this noise decreased the number of competent cells [12] suggesting that noise-driven mechanisms can evolve [7]. Importantly, in *B. subtilis*, initiation of competence is not affected by memory of previous events.

Reversible differentiation between the states im+ to im- was observed in *E. coli*, lysogenic for λCI8B7 [11]. Differentiation is both spontaneous and responds to an external factor (temperature), and is a transmissible ability of physiological change without genetic change.

Here we investigate the fitness value of inherent phenotypic variability and ability to switch between distinct phenotypes to cope with environment changes.

Real cell types are confined patterns of gene activity, hence, were identified as attractors [6]. However, real gene regulatory networks (GRN) are subject to molecular noise; hence the precise closure of a state cycle is problematic [8]. To address this problem, the concept of ergodic set was introduced, i.e., a set of states from which, once reached, the system cannot leave even due to noise [8]. If cell types are ergodic sets, and multi-cellular organisms typically have multiple cell types, then GRN models must have more than one ergodic set.

Here, we model cells with one ergodic set with two "noisy attractors". Noise can induce transitions between attractors. Each attractor is assumed to express a phenotype better fit to one of two possible environmental states. Environmental state transitions are also stochastic. This simple model appears to show the value of stochasticity in phenotype determination and of maintenance of phenotypic diversity in populations.

The cells are given the ability to mutate the rate constants controlling state transitions between the noisy attractors. We note that here we do not focus on the underlying GRN required to express the phenotypes. Thus, no evolution is possible on how well a phenotype is adapted to the world. Only the choice of available phenotypes is possible without genetic change (as in [11]).

2. MODELS AND SIMULATION

The models' dynamics follow the Stochastic Simulation algorithm [9] and are implemented in *SGNSim*[10]. Each cell's dynamics is independent of the rest. In each generation cells' fitness are measured. The best 50% are duplicated and their daughters simulated in the next generation. The others are eliminated.

In Model 1, we model to two cell types (reactions 1 to 8). Generation 1 (G1) consists of N identical cells, each with a 50% chance of being of type 1 or 2. Daughters inherit the type from the mothers. One cell type is unable

to flip between its two possible states. The phenotypic variability of its population is created at G1. The other cell type can stochastically flip between its two phenotypes. Cells are unable to regulate its internal state as a function of environment state.

$$W \xrightarrow{init} W_A,\ W \xrightarrow{init} W_B \quad (1)$$

$$W_A \xrightarrow{CW_A} W_B,\ W_B \xrightarrow{CW_B} W_A \quad (2)$$

$$S_{init} \xrightarrow{init} Si_a,\ S_{init} \xrightarrow{init} Si_b \quad (3)$$

$$Si_a \xleftrightarrow{flip(i)} Si_b \quad (4)$$

$$*Si_a + *W_A \xrightarrow{r} fit_i \quad (5)$$

$$*Si_b + *W_B \xrightarrow{r} fit_i \quad (6)$$

$$*Si_b + *W_A + fit_i(\min:1) \xrightarrow{p} \varnothing \quad (7)$$

$$*Si_a + *W_B + fit_i(\min:1) \xrightarrow{p} \varnothing \quad (8)$$

Reactions (1) initialize the environment state (W_A or W_B). Only one can occur, once, in the beginning of the simulation (since W(t=0) = 1). Reactions (2) model environment state transitions, i.e., switching between states W_A and W_B. Reactions (3) initialize the state of cell's of G0 only. Index i indicates the cell type (1 or 2). Reactions (4) allow a cell to change its phenotype between S_a and S_b (a fixed phenotype is attained setting these reactions rate constants to zero). On average, at G1, 50% of the cells are type 1 and the rest type 2.

A cell's fitness equals its number of fit_i units at the end of a simulation, computed by reactions (5) to (8). Reactions (5) and (6) allow gaining fit units, when cell and environment states match. If environment and cell are in opposite states, fitness is lost via (7) or (8).

The number of fit units does not affect the penalization for being in an "incorrect" state, since the propensity of reactions 7 and 8 doesn't depend on the number of fit_i. The X(min:1) notation means that this quantity all contributes to propensity with value 0 if not existent and 1 if existent, independently of its quantity.

When a cell divides, all substances are duplicated and equally shared by the daughter cells, including fit units. The weak dependence of cell fitness on initial conditions is meant to mimic the fact that mother cells pass its molecules, to its daughters, thus a fitter mother ought to provide a small initial "advantage" to its daughters. However, cells' lifetime is set so that the initial state is not the main determinant of a cell's survival, which depends mostly on the cell's ability to cope with the environment during its lifetime.

In Model 2 cell's state transitions depend on the environment state. Signaling molecules (sig_i) are generated via reactions (9), and mimic cell sensors informing about the environment state. These decay via reactions (10). Signaling molecules carrying information contrary to the environment state are further degraded by (11):

$$*W_A \xrightarrow{sig} sig_A,\ *W_B \xrightarrow{sig} sig_B \quad (9)$$

$$sig_A \xrightarrow{dsig} \varnothing,\ sig_B \xrightarrow{dsig} \varnothing \quad (10)$$

$$*W_B + sig_A \xrightarrow{D} \varnothing\ *W_A + sig_B \xrightarrow{D} \varnothing \quad (11)$$

To allow these signals to regulate state transitions we alter reactions (4), setting distinct rate constants:

$$Si_a + *sig_b \xrightarrow{flip(i)ab} Si_b \quad (4a)$$

$$Si_b + *sig_a \xrightarrow{flip(i)ba} Si_a \quad (4b)$$

In Model 3, the cell can regulate its state transitions. A mutation mechanism affecting the value of the rate constants "flip" is introduced for cell type 1. As the fitter cells are chosen, these rates ought to acquire local optimum values. To mutate the rate constants ($flip_{ab}$ and $flip_{ba}$) at run-time we introduce virtual substances ($Ad_{w,z}$), two for each rate constant such that: if "$Ad_{ab,up}$" quantity increases, $flip_{ab}$ increases linearly. If "$Ad_{ab,down}$" quantity increases, $flip_{ab}$ decreases linearly.

Creation and decay reactions of these substances are introduced, both independent of the substances quantity, i.e., mutation of the rates constants values occur via a Markov process. Let "$Ad_{w,z}$" be such that w is either "ab" or "ba", and "z" is "up" or "down". We add the following reactions to model 2 (setting $k_{up} = k_{down}$):

$$*Ad_{w,z} \xrightarrow{k_{down}} \varnothing \quad (12)$$

$$\xrightarrow{k_{up}} Ad_{w,z} \quad (13)$$

Finally, in model 3, we change the way the propensity P of reactions (4a) and (4b) are calculated to:

$$P^{4a} = \frac{flip(i)_{ab}.[Si_a].[sig_b].[Ad(ab,up)]}{[Ad(ab,down)]} \quad (14)$$

$$P^{4b} = \frac{flip(i)_{ba}.[Si_b].[sig_a].[Ad(ba,up)]}{[Ad(ba,down)]} \quad (15)$$

As the quantities of the virtual substances change in time, via reactions 12 and 13, so will the propensity of reactions 4a and 4b. As the fitter cells are selected at each generation, one can observe which values of $\frac{flip(i)_{ab}.[Ad(ab,up)]}{[Ad(ab,down)]}$ and $\frac{flip(i)_{ba}.[Ad(ba,up)]}{[Ad(ba,down)]}$ locally optimize cell fitness in various environments.

We now present the results of simulating the dynamics of populations based on these three models.

3. RESULTS

Each simulation models 1000 cells per generation. Cells' lifetime is 1000 seconds. Unless stated otherwise, rate constants (units in s^{-1}) are: init = 10^9, CW_a=CW_b=0.01, $flip_{ab}(1)$=$flip_{ba}(1)$=0.01, r=1, p=0.1. We first compare the fitness of types 1 and 2, using model 1, where cells have no information on the environment. Type 2 (fixed phenotype) almost always wins. In biased environments (e.g., 75% of the time as W_A), is even more likely that type 2 wins. The results indicate that the ability to stochastically change phenotype doesn't provide advantages without observing the environment. Although both cell types have, on average, 50% of its cells

adapted to the environment state at any time, any small disadvantage (due to stochastic fluctuations of phenotypic expression in type 1) is sufficient to unbalance the unstable equilibrium, since one is simulating finite populations. As type 2 population grows, its victory over type 1 becomes ever more likely.

Fig. 1 shows one time series of model 2 in an unbiased bistable environment. Stochastic state transition only provides selective advantage because phenotypic transitions follow real time observations of the environment (as is the case in λCI8B7 [11]).

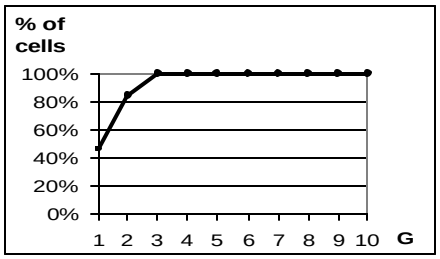


Fig 1. Simulation of model 2. Percentage of the population 1. Cell type 2 has fixed phenotype. Cell type 1 flip rate constant is $0.01s^{-1}$.

We now analyze the population dynamics in biased environments. A biased environment is obtained by setting different values for the rate constants of reactions 2, CW_a and CW_b. The ratio between them determines the expected time in each state. Fig 2 shows the outcome in various biased environments. Cell type 1 (fixed state) wins for highly biased environments since in these the environment is mostly in one state. Only rarely and for short time durations does the other state occur.

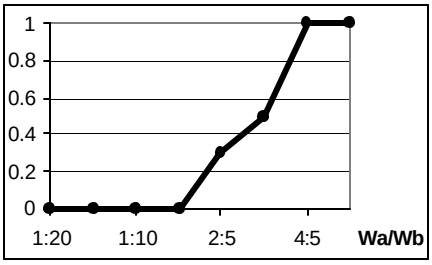


Fig 2. Fraction of wins of type 1 for various environment states biases (100 simulations per data point).

The rate at which the environment state changes also affects the outcome. We simulated model 2 setting $CW_a = CW_b$, with various values (Fig. 3). The transition value of CW above which cell type 1 wins is 0.001, i.e., on average only 1 environment change occurs per cell lifetime. Below this value, on average, a cell will face the same environmental conditions during its entire lifetime, making unnecessary for survival the ability to flip, i.e., a cell with fixed phenotype, starting its life well adapted, will most likely remain so through its life.

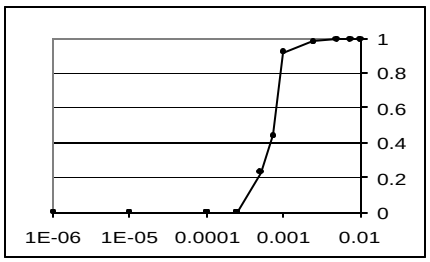


Fig 3. Fraction of wins of cell type 1 for various state transition rates (100 sim. per data point).

Using model 3, where cells are able to mutate the rate constants of the reactions responsible for phenotypic state transitions (reactions 4a and 4b), we now observe cell type 1 evolution when facing a biased environment. We compare two cell populations (simulated separately). One is able to mutate the rate constants controlling phenotypic transition (adaptive), while the other cannot (non-adaptive). In all cells of G1, propensity of reactions 4a and 4b are equal: $P^{4a}(t=0) = P^{4b}(t=0)$.

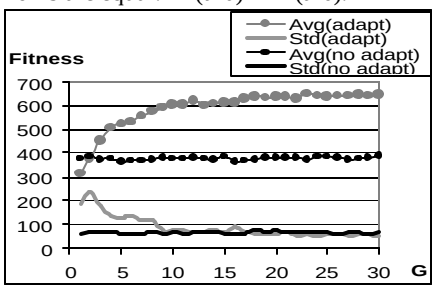


Fig 4. Evolution of avg. and std. of fitness of populations of cells able (grey) and unable to mutate (black).

Because $CW_b = 5. CW_a$, on average, the environment is in state A 85% of the time. Interestingly, in the population of cells able to mutate, the fitness variability of the population decreased, while the cells unable to mutate maintained their fitness and variability constant.

The fitness increase is due to fluctuations in the propensity of the reactions controlling cells' state transition (Fig. 4). The propensity to go from state S_b to S_a became by selection, much higher than the opposite, allowing the cells to remain in S_a most of the time (Fig. 5).

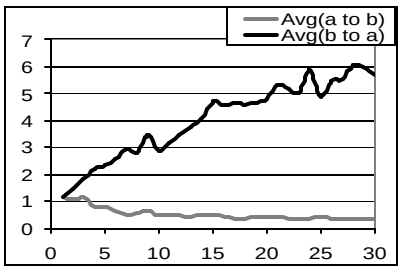


Fig 5. Evolution in 30 generations of the mutation parameters. These, multiplied by the flip rate constant, equal the propensity of reactions (4a) and (4b).

Fig 6 shows the average fraction of time that the adaptable cell population spent in state Sa. Interestingly, it stabilizes at ~95%, while the environment state is W_a only ~85% of the time. The solution adopted by this cell population appears to be to almost ignore transient environment changes. However, a degree of stochasticity in phenotypic determination is maintained, rather than being nullified. Thus the cells can, in sudden changes in the environment state bias for example, quickly shift the population phenotypic composition, selecting from the few remaining cells that express the less common state, those that mutate to adapt to the new situation.

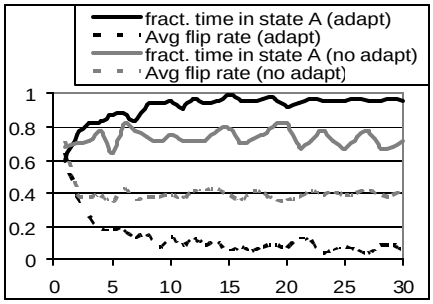


Fig 6. Evolution of the average fraction of time a cell spends in state A and average state flip rate (black lines: cells able to mutate; gray lines: cells unable to mutate).

4. CONCLUSIONS

We simulated a simple model to study selective advantages of stochastic phenotypic determination in a stochastic environment. Although using a simplistic model, where gene networks determining the phenotype were not explicitly modeled, our results agree with experimental observations. Probabilistic phenotypic determination provides selective advantages if transitions between latent phenotypes can be biased by environmental conditions. This is the case in *B. subtilis* differentiation [7] (a memoryless process), or lactose usage in *E. coli* [4] (where transitions depend on previous states).

The model cells can adapt to environmental changes occurring sparsely in time, due to its inherent probabilistic nature of choice of existing phenotypes. In highly biased worlds, adaptive cells were able to become non-reactive to transient external changes while maintaining a degree of phenotypic diversity capable of coping with sudden change in environment state biases.

We note that cells fitness improvement is far from trivial in this model. E.g., even if phenotype toggling and environment flip rates match, that doesn't imply that cell and environment state will match. Nevertheless, the results show that adaptation is possible by mutation. Importantly, stochasticity of phenotypic determination provides robustness to environmental changes.

While our simple model of cells could easily evolve to adapt to environmental biases, we note that adaptation to changes in the flip rate of an unbiased environment is harder. Further studies are required. Perhaps the difficulty is related to the need of varying simultaneously the two parameters controlling state transitions rate. Otherwise state transitions become biased which diminishes fitness. Associated with the uncertainty of the selection of even the fitter cells, evolution by mutation becomes even less likely to occur in this scenario. We finally note that, in general, population size affects the degree of randomness of the outcome suggesting that, e.g., studies of survival ability of small populations ought not to be done using deterministic models.

In the future, we aim to increase the complexity of phenotypic diversity, environmental conditions and implement more realistic mechanisms of phenotype determination, namely, the gene networks expressing them.

5. ACKNOWLEDGEMENTS

We thank the support of Acad. Finland, proj. No. 213462 (Finnish Centre of Excellence prog. 2006-11).

6. REFERENCES

[1] M. Samoilov, G. Price, and A.P. Arkin, "From Fluctuations to Phenotypes", *Science*, re17, 2006.

[2] R. Kellermayer, "Physiologic noise obscures genotype-phenotype correlations", *Amer. J. Med. Genet*. 143A, pp. 1306–7, 2007.

[3] J. Yu, J. Xiao, X. Ren, K. Lao and X. Xie, "Probing Gene Expression in Live Cells, One Protein Molecule at a Time", *Science* 311, pp. 1600-3, 2006.

[4] W. Smits, O. Kuipers, and J. Veening, "Phenotypic variation in bacteria: the role of feedback regulation". *Nature* 4, pp. 259–71, 2006.

[5] F. Biddle, B. Eales, "The degree of lateralization of paw usage (handedness) in the mouse is defined by three major phenotypes". *Beh. Genet.* 26, pp 391–406, 1999.

[6] S.A. Kauffman, *"The Origins of Order"*, Oxford Univ. Press, 1993.

[7] G. Suel, J. Garcia-Ojalvo, L. Liberman, and M. Elowitz, "An excitable gene regulatory circuit induces transient cellular differentiation", *Nature*, 440(23), 2006.

[8] A.S. Ribeiro and S.A. Kauffman, "Noisy Attractors and Ergodic Sets in Models of Genetic Regulatory Networks", *J. Theo. Bio*. 247(4), pp. 743-55, 2007.

[9] D.T. Gillespie, "Exact stochastic simulation of coupled chemical reactions" *J. Phy. Chem.* 81: pp.2340–61, 1977.

[10] A.S. Ribeiro, J. Lloyd-Price, "SGNSim, Stochastic Gene Networks Simulator", *Bioinf.* 23(6), pp.777-9, 2007.

[11] Z. Neubauer and E. Calef, "Immunity phase shift in defective lysogens: non-mutational hereditary change in early regulation of λ phage, *J. Mol. Biol.* 51, 1-13, 1970.

[12] H. Maamar, A.Raj, D.Dubnau, "Noise in gene expression determines cell fate in B. subtilis", *Science* 317, 2007.

WHEN CORRELATIONS MATTER - RESPONSE OF DYNAMICAL NETWORKS TO SMALL PERTURBATIONS

Thimo Rohlf[1,2]*, Natali Gulbahce*[3,4] *and Christof Teuscher*[5]

[1]Santa Fe Institute, 1399 Hyde Park Road, Santa Fe, NM 87501, USA
[2]Max-Planck-Institute for Mathematics in the Sciences, Inselstrasse 22, D-04103 Leipzig, Germany
[3] Center for Complex Network Research, Northeastern University, Boston, MA 02115, USA
[4] Center for Cancer Systems Biology, Dana Farber Cancer Institute, Boston, MA, 02215, USA
[5] Los Alamos National Laboratory, CCS-3, MS B256, Los Alamos, NM 87545, USA
rohlf@santafe.edu

ABSTRACT

We systematically study and compare damage spreading for random Boolean and threshold networks under small external perturbations (damage), a problem which is relevant to many biological networks. We identify a new characteristic connectivity K_s, at which the average number of damaged nodes after a large number of dynamical updates is independent of the total number of nodes N. We estimate the critical connectivity for finite N and show that it systematically deviates from the annealed approximation. Extending the approach followed in a previous study [1], we present new results indicating that internal dynamical correlations tend to increase not only the probability for small, but also for very large damage events, leading to a broad, fat-tailed distribution of damage sizes. These findings indicate that the descriptive and predictive value of averaged order parameters for finite size networks - even for biologically highly relevant sizes up to several thousand nodes - is limited.

1. INTRODUCTION

Random Boolean networks (RBN) were originally introduced as simplified models of gene regulation [2, 3]. In the limit of large system sizes, they exhibit a dynamical order-disorder transition at a critical wiring density K_c [4]; similar observations were made for sparsely connected random threshold (neural) networks (RTN) [5, 6]. For a finite system size N, the dynamics of both systems converge to periodic attractors after a finite number of updates. At K_c, the phase space structure in terms of attractor periods [7], the number of different attractors [8] and the distribution of basins of attraction [9] is complex. Furthermore, critical networks exhibit many properties reminiscent of biological networks, leading to the idea K_c might be an "attractor of evolution" [3].

To ensure proper function, regulatory networks in living cells have to be robust (insensitive) against external perturbations. In terms of RBN/RTN dynamics, perturbations can disrupt the generic dynamical state (fixed point or periodic attractor) of the network, and hence are referred to as "damage"; this type of study has been applied, for example, to the perturbation of gene expression patterns in a cell due to mutations [10].

Mean-field techniques as, for example, the *annealed approximation* (AA) introduced by Derrida and Pomeau [4], allow for an analytical treatment of damage spreading and exact determination of the critical connectivity K_c under various constraints [11]. It has been shown that local rewiring rules coupled to mean-field-like order parameters of the dynamics can drive both RBN and RTN to self-organized criticality [12, 13].

Studies of RBN/RTN dynamics based on the AA usually implicitly assume that, at least for large N, *principal* properties of damage spreading should not depend on the initial perturbation size. For example, the determination of K_c using a one-bit initial perturbation (sparse percolation limit), or an initial perturbation size increasing with N should yield the same value for large N, since it is assumed that correlations can be neglected in this limit by averaging over a large number of different random network realizations. In this paper, we extend results of a previous study [1] and present the following findings that are, at least in part, in clear contradiction to these assumptions:

- In section 3.1, we identify a new characteristic point $K_s < K_c$, where the expectation value of the number of damaged nodes after large number of dynamical updates is independent of N.
- By the definition of marginal damage spreading, we estimate the critical connectivity $K_c(N)$ for finite N, and present evidence that, even in the large N limit, for small initial perturbations K_c systematically deviates from the predictions of the AA (section 3.2).
- In section 3.3, we present new results proving that, slightly below K_c, starting from random initial conditions, the AA holds only for small times t, indicating that after passing transient dynamics inherent correlations considerably affect damage propagation.

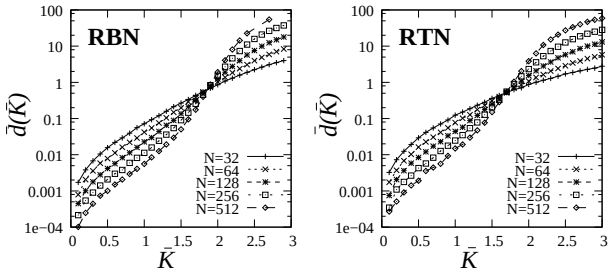


Figure 1. Average Hamming distance (damage) $\bar{d}$ after 200 system updates, averaged over 10000 randomly generated networks for each value of $\bar{K}$, with 100 different random initial conditions and one-bit perturbed neighbor configurations for each network. For both RBN and RTN, all curves for different N approximately intersect in a characteristic point K_s.

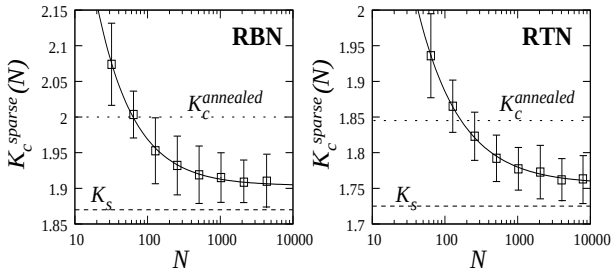


Figure 2. The critical connectivity $K_c^{sparse}(N)$ in the SP limit as a function of N. Curves are power-law fits according to Eq. (9), straight dashed lines mark $K_c^{annealed}$ and K_s for RBN and RTN, respectively.

- Last, we show that vanishing, as well as large damage events are overrepresented in damage size statistics, leading to highly skewed distributions, which are poorly characterized by averages (section 3.4).

2. DYNAMICS

2.1. Random Boolean Networks

A RBN is a discrete dynamical system composed of N automata. Each automaton is a Boolean variable with two possible states: $\{0,1\}$, and the dynamics is such that

$$\mathbf{F} : \{0,1\}^N \mapsto \{0,1\}^N, \tag{1}$$

where $\mathbf{F} = (f_1, ..., f_i, ..., f_N)$, and each f_i is represented by a look-up table of K_i inputs randomly chosen from the set of N automata. Initially, K_i neighbors and a look-table are assigned to each automaton at random.

An automaton state $\sigma_i^t \in \{0,1\}$ is updated using its corresponding Boolean function:

$$\sigma_i^{t+1} = f_i(\sigma_{i_1}^t, \sigma_{i_2}^t, ..., \sigma_{i_{K_i}}^t). \tag{2}$$

We randomly initialize the states of the automata (initial condition of the RBN). The automata are updated synchronously using their corresponding Boolean functions.

2.2. Random Threshold Networks

An RTN consists of N randomly interconnected binary sites (spins) with states $\sigma_i = \pm 1$. For each site i, its state at time $t+1$ is a function of the inputs it receives from other spins at time t:

$$\sigma_i(t+1) = \text{sgn}\left(\sum_{j=1}^{N} c_{ij}\sigma_j(t) + h.\right) \tag{3}$$

The N network sites are updated synchronously. In the following discussion the threshold parameter h is set to zero. The interaction weights c_{ij} take discrete values $c_{ij} = +1$ or -1 with equal probability. If i does not receive signals from j, one has $c_{ij} = 0$.

3. RESULTS

3.1. Scaling

We first study the expectation value $\bar{d}$ of damage, quantified by the Hamming distance of two different system configurations, after a large number T of system updates. Fig. 1 shows $\bar{d}$ as a function of the average connectivity $\bar{K}$ for different network sizes N by using a random ensemble for statistics. For both RBN and RTN, the observed functional behavior strongly suggests that the curves approximately intersect at a common point (K_s, d_s), where the observed Hamming distance for large t is independent of the system size N.

We verified this finding quantitatively by using finite-size-scaling methods [1]. In particular, one can show that $\bar{d}$ as a function of N and $\bar{K}$ obeys the following scaling ansatz:

$$\bar{d}(\bar{K}, N) = a(\bar{K}) \cdot N^{\gamma(\bar{K})} + d_0(\bar{K}),\ -1 \leq \gamma \leq 1. \tag{4}$$

It is straight-forward to show that $\gamma \to -1$ for small $\bar{K} \to 0$, and that $\gamma \to 1$ for densely connected networks above the percolation transition ($\bar{K} > K_c$). Evidently, this implies that at some characteristic connectivity K_s, there has to be a transition from negative to positive γ values, with $\gamma(K_s) \approx 0$. It is a very interesting question whether K_s coincides with K_c, or if it is different from K_c for large N. For a precise numerical determination of K_s, one can make use of the fact that $\bar{d}$ exhibits an exponential dependence near K_c:

$$\bar{d}(\bar{K}, N) \approx c_1(N)\ \exp\left[c_2(N)\, N^{\alpha}\, \bar{K}\right] \tag{5}$$

with $\alpha \approx 0.42$. High-accuracy fits of this dependence (with c_1 and c_2 as adjustable parameters) in the interval $1.6 \leq \bar{K} \leq 2.1$ yield

$$(K_s^{RBN}, d_s^{RBN}) = (1.875 \pm 0.05, 0.62 \pm 0.05) \tag{6}$$

for RBN and, correspondingly,

$$(K_s^{RTN}, d_s^{RTN}) = (1.729 \pm 0.045, 0.51 \pm 0.04) \tag{7}$$

for RTN. We verified these findings up to $N = 16384$, waiting $T = 5000$ updates for the dynamics to relax; for even larger N, simulations become intractable due to

exponentially increasing relaxation times. Evidently, we tend to miss large damage events since they need the most time to develop. Facing this unavoidable *biased undersampling* of large avalanches, one can argue that the *true* values of K_s are probably even lower than our measured values. From this evidence, and also from more refined scaling arguments [1], we conclude that K_s is distinct from K_c in the limit of large N.

3.2. Deviations of K_c from the annealed approximation

Interestingly, K_s is close to, but distinct from the critical connectivities $K_c^{RBN} = 2$ and $K_c^{RTN} = 1.845$, as predicted by the AA. Since in this study we consider the limit of very weak initial perturbations which is usually not covered in theoretical studies of RBN/RTN dynamics, we now have to consider the possibility that K_c itself may deviate from the prediction of the AA. An intuitive definition of criticality for finite N can be formulated in terms of *marginal damage spreading*. If at time t one bit is flipped, one requires at time $t+1$ [11, 6]

$$\bar{d}(t+1) = \langle p_s \rangle(K_c) K_c = 1, \tag{8}$$

where $\langle p_s \rangle(\bar{K})$ is the average damage propagation probability. Fig. 2 shows $K_c^{sparse}(N)$, using the values $c_1(N)$ and $c_2(N)$ obtained from numerical fits of Eq. (5) for both RBN and RTN. We find that both systems, in a very good approximation, obey the scaling relationship

$$K_c^{sparse}(N) \approx b \cdot N^{-\delta} + K_c^{\infty} \tag{9}$$

with $b = 3.27 \pm 0.79$, $\delta = 0.85 \pm 0.07$ and $K_c^{\infty} = 1.9082 \pm 0.008$ for RBN and $b = 3.853 \pm 0.76$, $\delta = 0.736 \pm 0.05$ and $K_c^{\infty} = 1.7595 \pm 0.008$ for RTN. Hence, in the limit $N \to \infty$, we can extrapolate

$$K_c^{\infty,RBN} = 1.9082 \pm 0.008 \tag{10}$$

for RBN, and for RTN

$$K_c^{\infty,RTN} = 1.7595 \pm 0.008. \tag{11}$$

Thus, for both RBN and RTN in the sparse percolation limit, we make the surprising observation that K_c^{sparse} systematically deviates from $K_c^{annealed}$. While we find $K_c^{sparse}(N) > K_c^{annealed}$ for small $N < 128$, for larger N we observe a monotonic decay that approaches an asymptotic value considerably below $K_c^{annealed}$, suggesting that the observed deviations from the AA also hold in the large N limit. In the following two subsections, we will extend this analysis and discuss possible causes for these deviations.

3.3. Time dependence of $\bar{d}$

Since we found systematic deviations from the AA for large t, it is interesting to ask whether the AA still holds for small t, starting from random initial states. In particular, one can derive the following recursive map for damage propagation at $t > 0$ [6]:

$$\bar{d}(t) = N \cdot \langle p_s \rangle(\bar{K}) \cdot \left(1 - e^{-\bar{K} \cdot \bar{d}(t-1)/N}\right), \tag{12}$$

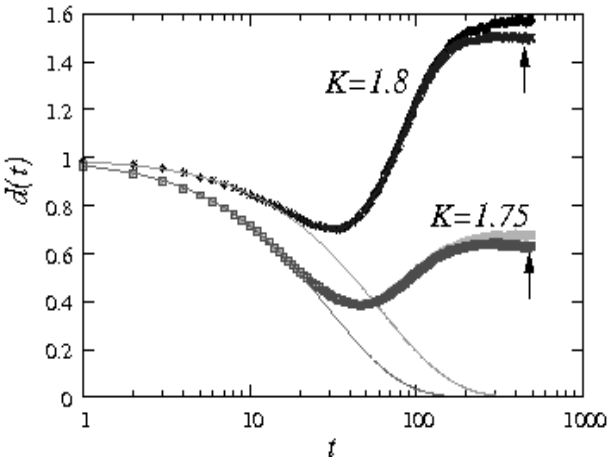


Figure 3. Time-dependence of (average) damage propagation in RTN of size $N = 4096$ just below K_c; damage $\bar{d}$ at time t was averaged over 10^5 network realizations and 100 different initial conditions (and the corresponding neighbor states with one bit perturbed at random) at $t = 0$ for each data point . Lined curves are the corresponding solutions of the AA (Eq. (12)). For $t \geq 20$, pronounced deviations of simulation results from the AA are found, in particular for $\bar{K} = 1.8$. Arrows indicate results with "corrected" statistics, i.e. without "pseudo-damage" due to attractor phase lags.

where $\langle p_s \rangle(\bar{K})$ is the average probability that a link propagates damage. Let us now test this relationship in the interesting range $K_s \leq \bar{K} \leq K_c^{annealed}$ for ensembles of randomly generated networks (RTN with Poissonian degree-distribution), with one-bit perturbations of randomly chosen initial conditions. Figure 3 shows that, for small t, the dependence for $\bar{d}(t)$ found in numerical simulations obeys this prediction very well. However, after an initial decrease of $\bar{d}(t)$, an *increase* above the initial damage size (i.e. supercritical behavior) is found, in clear contradiction to the AA. This indicates that, after the system has passed transient dynamics, inherent dynamical correlations considerably modify damage propagation (fractal structure of attraction basins [9]). One can also show that "pseudo-damage" events , i.e. cases where networks run on the same attractor, but with a phase lag captured in a non-zero Hamming distance, do *not* substantially contribute to this effect (arrows in Fig. 3). This proves that our results are very robust against changes in the way statistics is taken.

3.4. Distribution of damage sizes

Let us now go beyond averaged (mean-field) quantities and investigate detailed statistics of damage sizes. For this purpose, for different $\bar{K}$ and N ensembles of Z_e random network realizations were created; for each network realization, Z_i random initial conditions $\vec{\sigma}$ (plus a neighbor state with one bit perturbed at random) were tested, and statistics of damage sizes was taken after 1000 dynamical updates. Notice that we do *not* average damage sizes for a given network realization, since this would again represent a kind of mean-field approximation. Figure 4 shows that the resulting statistical distributions near K_c

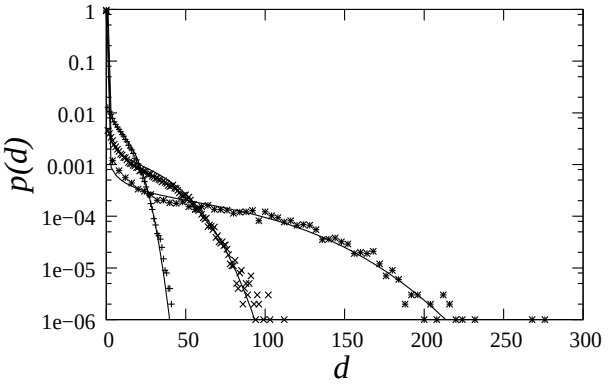


Figure 4. Statistical distribution $p(d)$ of damage sizes for three different system sizes: $N = 64$ (+), $N = 256$ (x) and $N = 1024$ (*). Lined curves are solutions of Eq. (13).

are highly skewed, with more than 90% events of vanishing damage size, and a flat tail of large damage events which becomes more and more pronounced for increasing N. Similar problems have been studied by Samuelsson and Socolar [14] for the number of *undamaged* nodes u in the limit of *exhaustive* percolation. From symmetry considerations, it follows that the probability distribution $p(d)$ of the number d of *damaged* nodes in the limit of sparse percolation obeys a similar dependence as u in the case of exhaustive percolation, and hence

$$p(d) \approx a(N) \cdot \frac{\exp\left[-\frac{1}{2}(d \cdot N^{-2/3})^3\right]}{\sqrt{d \cdot N^{-2/3}}}, \qquad (13)$$

where $a(N)$ is a free parameter. One finds that the results of numerical simulations agree very well with this estimate even for considerably small N (Fig. 4). From the shape of these distributions, one recognizes that vanishing, as well as large damage events are much more probable than expected from mean-field considerations. In part, this explains the deviations from the annealed approximation found for $\bar{d}$ near criticality (Fig. 3), and it also questions in how far averaged quantities deliver an informative description of RBN/RTN dynamics for finite size N.

4. DISCUSSION

We showed that, for very weak (one-bit) perturbations of the initial states of RBN and RTN dynamics, the resulting damage at later times exhibits a non-trivial scaling with network size N, and, near the critical order-disorder transition - the so-called the 'edge of chaos' - considerable deviations from the annealed approximation. These deviations have escaped earlier studies, since usually the *rescaled* damage $\bar{d}/N$ (or the overlap $1 - \bar{d}/N$, respectively) was studied, and the thermodynamic limit of large N was considered. Our study indicates that there is a strong need for more refined studies of damage propagation in RBN/RTN, that explicitly take into account dynamical correlations and the fractal structure of attraction basins [9]. One may expect that the situation is even more complex for networks with more realistic topologies. Even for simple random graphs, as applied in this study, damage size distributions are highly skewed, questioning the descriptive and predictive value of simple, averaged order parameters for this class of complex systems.

5. REFERENCES

[1] T. Rohlf, N. Gulbahce, and C. Teuscher, "Damage spreading and criticality in finite dynamical networks," *Phys. Rev. Lett.*, vol. 99, pp. 248701, 2007.

[2] S. Kauffman, "Metabolic stability and epigenesis in randomly constructed genetic nets," *J. Theor. Biol.*, vol. 22, pp. 437–467, 1969.

[3] S. Kauffman, *The Origins of Order: Self-Organization and Selection in Evolution*, Oxford University Press, 1993.

[4] B. Derrida and Y. Pomeau, "Random networks of automata: a simple annealed approximation," *Europhys. Lett.*, vol. 1, pp. 45–49, 1986.

[5] K. Kürten, "Correspondence between neural threshold networks and kauffman boolean cellular automata," *J. Phys. A*, vol. 21, pp. L615–L619, 1988b.

[6] T. Rohlf and S. Bornholdt, "Criticality in random threshold networks: Annealed approximation and beyond," *Physica A*, vol. 310, pp. 245–259, 2002.

[7] R. Albert and A.-L. Barabasi, "Dynamics of complex systems: Scaling laws for the period of boolean networks," *Physical Review Letters*, vol. 84, pp. 5660–5663, 2000.

[8] B. Samuelsson and C. Troein, "Superpolynomial growth in the number of attractors in kauffman networks," *Phys. Rev. Lett*, vol. 90, pp. 098701, 2003.

[9] U. Bastolla and G. Parisi, "Relevant elements, magnetization and dynamical properties in kauffman networks: A numerical study," *Physica D*, vol. 115, pp. 203–218, 1998.

[10] P. Ramö, J. Kesseli, and O. Yli-Harja, "Perturbation avalanches and criticality in gene regulatory networks," *J. Theor. Biol.*, vol. 242, pp. 164–170, 2006.

[11] R. Solé and B. Luque, "Phase transitions and antichaos in generalized kauffman networks," *Phys. Lett. A*, vol. 196, pp. 331–334, 1995.

[12] S. Bornholdt and T. Rohlf, "Topological evolution of dynamical networks: Global criticality from local dynamics," *Phys. Rev. Lett.*, vol. 84, pp. 6114–6117, 2000.

[13] M. Liu and K. E. Bassler, "Emergent criticality from coevolution in random boolean networks," *Phys. Rev. E*, vol. 74, pp. 041910, 2006.

[14] B. Samuelsson and J. Socolar, "Exhaustive percolation on random networks," *Phys. Rev. E*, vol. 74, pp. 036113, 2006.

SIMILARITY MEASURES BETWEEN EXPERIMENTS AND A MODEL FOR STOCHASTIC NEURONAL FIRING

Antti Saarinen, Olli Yli-Harja and Marja-Leena Linne

Department of Signal Processing, Tampere University of Technology,
P.O. Box 553, FI-33101 Tampere, Finland
antti.saarinen@cs.tut.fi

ABSTRACT

When developing parameter optimization methods for stochastic models it is imperative to be able to compare the model output with the learning data. Due to stochasticity, it is not enough to study the norm of the difference between the model output and the learning data. This is the case also with models for cerebellar granule cell which exhibits stochastic behavior and the responses to repeated current stimulation vary slightly. In model-level this means that we should get slightly different outputs with the same set of parameters. In this work, new ways of measuring similarity between the model output and the learning data are introduced. We conclude that we are able to produce responses with matching characteristics to experimental data.

1. INTRODUCTION

When developing automated parameter optimization methods for neuronal models it is essential to be able to compare the output of the model with the experimental learning data. This can be done, for example, by studying the norm of the difference between the model output data and the learning data. The parameter values of the model are considered optimized when this norm is minimized. It has been shown, however, that this approach is not sufficient for models describing the electroresponsiveness of neurons [1]. For example, the difference between two traces of firing data can result in a large value of the norm when there is only a phase-shift between the traces. Usually, this difference is so large that we would get a smaller value of the norm if we would compare a silent trace to the experimental data.

Cerebellar granule cell which is used as an example in our studies exhibits stochastic behavior and the responses to repeated current stimulation vary slightly [2, 3]. In the model-level this means that we should get slightly different outputs with the same set of parameters. Therefore, it is crucial to consider alternative ways of measuring similarity between the experimental data and the model output. In this work, we use the mean firing rate, mean interspike interval, standard deviation of the interspike intervals, and the coefficient of variation to describe the characteristics of experimental and simulated traces of firing data, and call the two traces similar if these measures match. In the future, we can, based on these similarity measures, construct automatic parameter optimization methods for the stochastic Hodgkin-Huxley type of neuron model presented in [4].

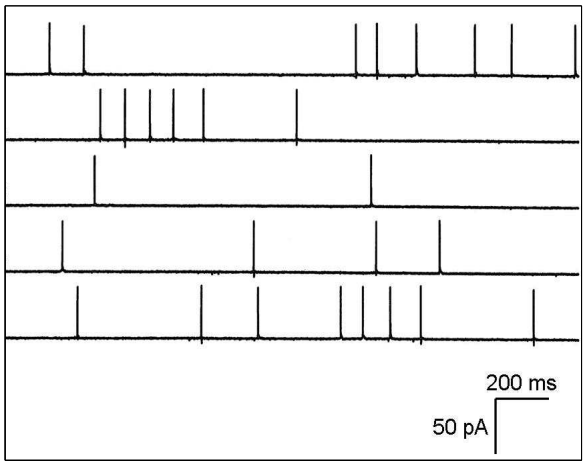


Figure 1. An example of the spontaneous intrinsic activity of a cerebellar granule cell in culture recorded for 10 seconds. The activity is recorded in the cell-attached configuration of the patch-clamp technique to obtain the timing of action potentials in an undisturbed way.

2. METHODS

2.1. Stochastic neuron model

In the model, $V_m(t)$ is the membrane potential, variables $x_{\mathrm{Na_F,a}}(t, V_m(t))$ and $x_{\mathrm{Na_F,i}}(t, V_m(t))$ are the time- and voltage-dependent gating variables for the activation and inactivation processes of the $\mathrm{Na_F}$ channels respectively (other gating variables similarly). Furthermore, $W_i = \{W_i(t), t \geq 0\}$ is Brownian motion, that is a Gaussian process with independent increments. This means that all finite-dimensional distributions of Brownian motion are Gaussian, $W_i(0) = 0$ almost surely, $\mathrm{E}(W_i(t)) = 0$ for all $t \geq 0$, and $\mathrm{Var}(W_i(t) - W_i(s)) = t - s$ for all $t \geq s \geq 0$. In addition, $\mathrm{d}W_i$ stands for the infinitesimal increment of Brownian motion. See Equation (1) and Tables 1 and 2 for the details of the model. In the following we assume

$$
\begin{cases}
\mathrm{d}V_m &= \frac{\mathrm{d}t}{C_m}(I_{\mathrm{app}} - G_{\mathrm{Na_F}} x_{\mathrm{Na_F,a}}^{p_{\mathrm{Na_F}}} x_{\mathrm{Na_F,i}}^{q_{\mathrm{Na_F}}}(V_m - E_{\mathrm{Na_F}}) - G_{\mathrm{K_{Dr}}} x_{\mathrm{K_{Dr},a}}^{p_{\mathrm{K_{Dr}}}}(V_m - E_{\mathrm{K_{Dr}}}) \\
&\quad -G_{\mathrm{K_A}} x_{\mathrm{K_A,a}}^{p_{\mathrm{K_A}}} x_{\mathrm{K_A,i}}^{q_{\mathrm{K_A}}}(V_m - E_{\mathrm{K_A}}) - G_{\mathrm{K_{ir}}} x_{\mathrm{K_{ir},a}}^{p_{\mathrm{K_{ir},a}}}(V_m - E_{\mathrm{K_{ir}}}) \\
&\quad -G_{\mathrm{Ca_{HVA}}} x_{\mathrm{Ca_{HVA},a}}^{p_{\mathrm{Ca_{HVA}}}} x_{\mathrm{Ca_{HVA},i}}^{q_{\mathrm{Ca_{HVA}}}}(V_m - E_{\mathrm{Ca_{HVA}}}) - G_{\mathrm{BK_{Ca}}} x_{\mathrm{BK_{Ca},a}}^{p_{\mathrm{BK_{Ca}}}}(V_m - E_{\mathrm{BK_{Ca}}}) \\
&\quad -\tfrac{1}{R_m}(V_m - E_m)) \\
\mathrm{d}x_{\mathrm{Na_F,a}} &= (\alpha_{\mathrm{Na_F,a}}(V_m)(1 - x_{\mathrm{Na_F,a}}) - \beta_{\mathrm{Na_F,a}}(V_m) x_{\mathrm{Na_F,a}})\mathrm{d}t + \sigma_1 \mathrm{d}W_1 \\
\mathrm{d}x_{\mathrm{Na_F,i}} &= (\alpha_{\mathrm{Na_F,i}}(V_m)(1 - x_{\mathrm{Na_F,i}}) - \beta_{\mathrm{Na_F,i}}(V_m) x_{\mathrm{Na_F,i}})\mathrm{d}t + \sigma_2 \mathrm{d}W_2 \\
\mathrm{d}x_{\mathrm{K_{Dr},a}} &= (\alpha_{\mathrm{K_{Dr},a}}(V_m)(1 - x_{\mathrm{K_{Dr},a}}) - \beta_{\mathrm{K_{Dr},a}}(V_m) x_{\mathrm{K_{Dr},a}})\mathrm{d}t + \sigma_3 \mathrm{d}W_3 \\
\mathrm{d}x_{\mathrm{K_A,a}} &= (\alpha_{\mathrm{K_A,a}}(V_m)(1 - x_{\mathrm{K_A,a}}) - \beta_{\mathrm{K_A,a}}(V_m) x_{\mathrm{K_A,a}})\mathrm{d}t + \sigma_4 \mathrm{d}W_4 \\
\mathrm{d}x_{\mathrm{K_A,i}} &= (\alpha_{\mathrm{K_A,i}}(V_m)(1 - x_{\mathrm{K_A,i}}) - \beta_{\mathrm{K_A,i}}(V_m) x_{\mathrm{K_A,i}})\mathrm{d}t + \sigma_5 \mathrm{d}W_5 \\
\mathrm{d}x_{\mathrm{K_{ir},a}} &= (\alpha_{\mathrm{K_{ir},a}}(V_m)(1 - x_{\mathrm{K_{ir},a}}) - \beta_{\mathrm{K_{ir},a}}(V_m) x_{\mathrm{K_{ir},a}})\mathrm{d}t + \sigma_6 \mathrm{d}W_6 \\
\mathrm{d}x_{\mathrm{Ca_{HVA},a}} &= (\alpha_{\mathrm{Ca_{HVA},a}}(V_m)(1 - x_{\mathrm{Ca_{HVA},a}}) - \beta_{\mathrm{Ca_{HVA},a}}(V_m) x_{\mathrm{Ca_{HVA},a}})\mathrm{d}t + \sigma_7 \mathrm{d}W_7 \\
\mathrm{d}x_{\mathrm{Ca_{HVA},i}} &= (\alpha_{\mathrm{Ca_{HVA},i}}(V_m)(1 - x_{\mathrm{Ca_{HVA},i}}) - \beta_{\mathrm{Ca_{HVA},i}}(V_m) x_{\mathrm{Ca_{HVA},i}})\mathrm{d}t + \sigma_8 \mathrm{d}W_8 \\
\mathrm{d}x_{\mathrm{BK_{Ca},a}} &= (\alpha_{\mathrm{BK_{Ca},a}}(V_m, [\mathrm{Ca}^{2+}])(1 - x_{\mathrm{BK_{Ca},a}}) - \beta_{\mathrm{BK_{Ca},a}}(V_m, [\mathrm{Ca}^{2+}]) x_{\mathrm{BK_{Ca},a}})\mathrm{d}t + \sigma_9 \mathrm{d}W_9 \\
\mathrm{d}[\mathrm{Ca}^{2+}] &= \left(\frac{B G_{\mathrm{Ca_{HVA}}} x_{\mathrm{Ca_{HVA},a}}^{p_{\mathrm{Ca_{HVA}}}} x_{\mathrm{Ca_{HVA},i}}^{q_{\mathrm{Ca_{HVA}}}} (V_m - E_{\mathrm{Ca_{HVA}}})}{\pi \cdot \mathrm{dcell}^2 \cdot \mathrm{dshell}} - \frac{[\mathrm{Ca}^{2+}] - [\mathrm{Ca}^{2+}]_{\mathrm{rest}}}{\tau_{Ca}} \right) \mathrm{d}t.
\end{cases}
\tag{1}
$$

Constant	Value	Description
R_m	0.57 Ωm^2	membrane resistance
C_m	0.03 F/m^2	membrane capacitance
E_m	-0.025 V	equilibrium membrane potential
$E_{\mathrm{Na_F}}$	+0.07 V	equilibrium potential for Na^+
$E_{\mathrm{K_{Dr}}} = E_{\mathrm{K_A}} = E_{\mathrm{K_{ir}}}$	-0.075 V	equilibrium potential for K^+
$E_{\mathrm{Ca_{HVA}}}$	+0.14 V	equilibrium potential for Ca^{2+}
$E_{\mathrm{BK_{Ca}}}$	-0.085 V	equilibrium potential for BK_{Ca}
B	$5.2 \cdot 10^{-6}$ mol/C	constant for Ca^{2+} transfer into the cell
$[\mathrm{Ca}^{2+}]_{\mathrm{rest}}$	$100 \cdot 10^{-6}$ mol/m^3	$[Ca^{2+}]$ at rest
τ_{Ca}	$1 \cdot 10^{-3}$ s	time constant for the decay of intracellular free calcium
dcell	$6 \cdot 10^{-6}$ m	diameter of the granule cell
dshell	$1 \cdot 10^{-7}$ m	diameter of the shell defining the volume in which calcium ions are processed
$G_{\mathrm{Na_F}}$	400 S/m^2	maximal conductance for Na_F
$G_{\mathrm{K_{Dr}}}$	120 S/m^2	maximal conductance for K_{Dr}
$G_{\mathrm{K_A}}$	10 S/m^2	maximal conductance for K_A
$G_{\mathrm{K_{ir}}}$	28 S/m^2	maximal conductance for K_{ir}
$G_{\mathrm{Ca_{HVA}}}$	4.6 S/m^2	maximal conductance for Ca_{HVA}
$G_{\mathrm{BK_{Ca}}}$	30 S/m^2	maximal conductance for BK_{Ca}
$p_{\mathrm{Na_F}}$	3	exponential for Na_F activation
$q_{\mathrm{Na_F}}$	1	exponential for Na_F inactivation
$p_{\mathrm{K_{Dr}}}$	4	exponential for K_{Dr} activation
$p_{\mathrm{K_A}}$	3	exponential for K_A activation
$q_{\mathrm{K_A}}$	1	exponential for K_A inactivation
$p_{\mathrm{K_{ir}}}$	1	exponential for K_{ir} activation
$p_{\mathrm{Ca_{HVA}}}$	2	exponential for Ca_{HVA} activation
$q_{\mathrm{Ca_{HVA}}}$	1	exponential for Ca_{HVA} inactivation
$p_{\mathrm{BK_{Ca}}}$	1	exponential for BK_{Ca} activation

Table 1. Parameter values used in the simulations (See Equation (1) and Table 2).

that $\sigma_i = \sigma$ for $i = 1, \ldots, 9$.

2.2. Learning data

The learning data was obtained from primary cultures of cerebellar granule cells using the cell-attached configuration of the patch-clamp technique (see [5]). The method allows the recording of spontaneous action potential firing without breaking the cell membrane of the small granule cell. The spontaneous activity in this study was recorded from the cell soma without any externally applied stimuli. The cell cultures were prepared from the cerebellum of 7-day-old Wistar rats, as described in [5]. The cells were used for recordings at day seven or eight in culture.

2.3. Statistical measures

First we take a look at the mean firing rate of a trace of neuronal firing data. This gives us the average firing frequency of the cell or the model. However, firing can be very irregular, especially with small values of depolarizing current, and the length of interspike interval can vary. To capture this behavior we calculate also the standard deviation of the interspike intervals.

Variability in the firing produced by the stochastic model can also be assessed by examining the histograms of interspike intervals (see Figure 2) with different values of injected depolarizing current and different values of variable σ (see Equation (1)). The histograms reveal that the value of variable σ has a major effect on the firing with current pulses near the threshold of firing. With larger depolarizing current pulses firing becomes more regular and the value of σ does not have as clear an effect. This can be observed from the histograms as a smaller deviation in the interspike intervals.

We use the coefficient of variation (CV) of the interspike intervals which is often used to quantify the regularity or irregularity of neuronal firing data. A completely regular firing has a CV of zero. Our studies show variability in the mean firing rate when changing the value of parameter σ with depolarizing current pulses near the threshold of firing. With depolarizing current pulses above the threshold of firing the increase in the value of parameter σ increases the irregularity of firing measured with the CV. With depolarizing current pulses below the threshold of firing, the increase in the values of parameter σ enhances spontaneous activity, thus making the firing more regular.

3. RESULTS

In this study, we consider one trace of experimental data and, by tuning the parameter σ in the stochastic neuron model, produce statistically similar data with the model. We use the statistical similarity measures discussed above.

By tuning the parameter σ, it is possible to simulate data with similar statistical characteristics to experimental data. At this point, we select a range of test values for the parameter σ and evaluate the similarity measures at each value. For the data presented in Figure 1, we obtain matching statistical characteristics when we set $\sigma = 0.11$. For the data sets, mean firing frequency is 6.7 Hz, mean interspike interval 0.3501 s, standard deviation of interspike intervals 0.3512 s, and the coefficient of variation 1.0031. One realization of the model is shown in Figure 3.

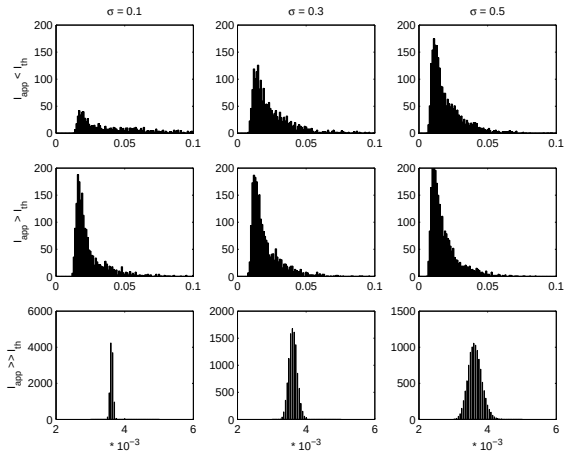


Figure 2. Histograms of interspike intervals. Firing is simulated for 50 seconds with each depolarizing current pulse, I_{app}, and value of parameter σ. Three different values for depolarizing current and for the parameter σ are used. Three upper panels show firing with depolarizing current below the firing threshold ($I_{app} = 11$ pA). Middle panels show firing with depolarizing current pulse just above the firing threshold ($I_{app} = 12$ pA). Lower panels show firing with considerably larger depolarizing current pulses ($I_{app} = 29$ pA). Note the different scales for the last row for illustrative purposes. Reproduced from [4].

4. CONCLUSIONS

We have characterized statistical properties of simulated and experimental traces of neuronal firing data from cultured cerebellar granule cells. We have used mean firing rate, mean interspike interval, standard deviation of interspike intervals, and the coefficient of variation to measure similarity between different traces. Based on these measures we have tuned the stochastic model to produce similar traces to experimental data.

We conclude that we are able to produce responses with matching characteristics to real data. These kinds of similarity measures can also be utilized when developing new automated parameter estimation methods for stochastic neuron models.

Estimation of stochastic neuron models is a challenging problem and has not been extensively studied in the literature. The first order statistics of the firing data presented in this paper may offer one way of fitting the parameters of stochastic models describing the electroresponsiveness of neurons.

5. ACKNOWLEDGMENTS

The work was supported by Tampere Graduate School in Information Science and Engineering (TISE), the Academy

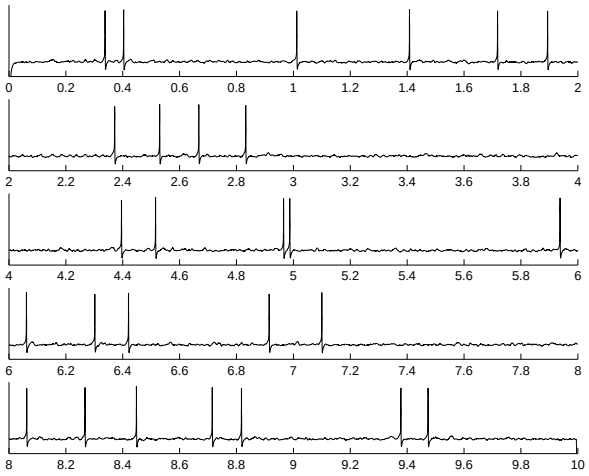


Figure 3. Simulated trace with the same statistical characteristics as those of the experimentally recorded response in Figure 1. As in Figure 1, firing is simulated for 10 seconds.

of Finland (decisions 106030 and 107694, and project number 213462 (Finnish Centre of Excellence program 2006-2011)), and the Emil Aaltonen Foundation.

6. REFERENCES

[1] C. Weaver and S. Wearne, "The role of action potential shape and parameter constraints in optimization of compartment models," *Neurocomputing*, vol. 69, pp. 1053–1057, 2006.

[2] E. D'Angelo, G. De Filippi, P. Rossi, and V. Taglietti, "Ionic mechanism of electroresponsiveness in cerebellar granule cells implicates the action of a persistent sodium currents," *Journal of Neurophysiology*, vol. 80, pp. 493–503, 1998.

[3] E. D'Angelo, T. Nieus, A. Maffei, S. Armano, P. Rossi, V. Taglietti, A. Fontana, and G. Naldi, "Theta-frequency bursting and resonance in cerebellar granule cells: experimental evidence and modeling of a slow k^+-dependent mechanism," *Journal of Neuroscience*, vol. 21, pp. 759–770, 2001.

[4] A. Saarinen, M.-L. Linne, and O. Yli-Harja, "Stochastic differential equation model for cerebellar granule cell excitability," *PLoS Computational Biology*, vol. 4, pp. e1000004, 2008.

[5] M.-L. Linne, S. S. Oja, and T. O. Jalonen, "Simultaneous detection of action potential current waveforms and single ion channel openings in rat cerebellar granule cells," *International Journal of Neural Systems*, vol. 7, pp. 377–384, 1996.

Channel	Process	Forward rate function	Backward rate function
Na_F	activation	$\alpha_{Na_F,a}(V_m) = 3 \cdot 10^3 e^{((V_m-0.01)+39\cdot10^{-3})\cdot0.081\cdot10^3}$	$\beta_{Na_F,a}(V_m) = 3 \cdot 10^3 e^{((V_m-0.01)+39\cdot10^{-3})\cdot-0.066\cdot10^3}$
Na_F	inactivation	$\alpha_{Na_F,i}(V_m) = 0.24 \cdot 10^3 e^{((V_m-0.01)+50\cdot10^{-3})\cdot-0.089\cdot10^3}$	$\beta_{Na_F,i}(V_m) = 0.24 \cdot 10^3 e^{((V_m-0.01)+50\cdot10^{-3})\cdot0.089\cdot10^3}$
K_{Dr}	activation	$\alpha_{K_{Dr},a}(V_m) = 0.34 \cdot 10^3 e^{((V_m-0.01)+38\cdot10^{-3})\cdot0.073\cdot10^3}$	$\beta_{K_{Dr},a}(V_m) = 0.34 \cdot 10^3 e^{((V_m-0.01)+38\cdot10^{-3})\cdot-0.018\cdot10^3}$
K_A	activation	$\alpha_{K_A,a}(V_m) = 2.2 \cdot 10^3 e^{((V_m-0.01)+46.7\cdot10^{-3})\cdot0.04\cdot10^3}$	$\beta_{K_A,a}(V_m) = 2.2 \cdot 10^3 e^{((V_m-0.01)+46.7\cdot10^{-3})\cdot-0.01\cdot10^3}$
K_A	inactivation	$\alpha_{K_A,i}(V_m) = 0.016 \cdot 10^3 e^{((V_m-0.01)+78.8\cdot10^{-3})\cdot-0.075\cdot10^3}$	$\beta_{K_A,i}(V_m) = 0.016 \cdot 10^3 e^{((V_m-0.01)+78.8\cdot10^{-3})\cdot0.055\cdot10^3}$
K_{ir}	activation	$\alpha_{K_{ir},a}(V_m) = 0.133 \cdot 10^3 e^{((V_m-0.01)+83.94\cdot10^{-3})\cdot-0.0411\cdot10^3}$	$\beta_{K_{ir},a}(V_m) = 0.17 \cdot 10^3 e^{((V_m-0.01)+83.94\cdot10^{-3})\cdot0.028\cdot10^3}$
Ca_{HVA}	activation	$\alpha_{Ca_{HVA},a}(V_m) = 0.049 \cdot 10^3 e^{((V_m-0.01)+29.06\cdot10^{-3})\cdot0.063\cdot10^3}$	$\beta_{Ca_{HVA},a}(V_m) = 0.082 \cdot 10^3 e^{((V_m-0.01)+18.66\cdot10^{-3})\cdot-0.039\cdot10^3}$
Ca_{HVA}	inactivation	$\alpha_{Ca_{HVA},i}(V_m) = 0.0013 \cdot 10^3 e^{((V_m-0.01)+48\cdot10^{-3})\cdot-0.055\cdot10^3}$	$\beta_{Ca_{HVA},i}(V_m) = 0.0013 \cdot 10^3 e^{((V_m-0.01)+48\cdot10^{-3})\cdot0.012\cdot10^3}$
BK_{Ca}	activation	$\alpha_{BK_{Ca},a}(V_m, [Ca^{2+}]) = \frac{2.5\cdot10^3}{1+1.5\cdot10^{-3}\cdot e^{-0.085\cdot10^3\cdot(V_m-0.01)/[Ca^{2+}]}}$	$\beta_{BK_{Ca},a}(V_m, [Ca^{2+}]) = \frac{1.5\cdot10^3}{1+[Ca^{2+}]/(150\cdot10^{-6}e^{-0.077\cdot10^3\cdot(V_m-0.01)})}$

Table 2. Forward and backward rate functions for different ion channel types in the stochastic model (see also Equation (1)).

QUANTITATIVE ANALYSIS OF THE RETE PROCESSES FOR THE DIAGNOSIS OF BORDERLINE MALIGNANCIES IN MICROSCOPIC ORAL CANCER IMAGES

Mustafa M. Sami[1], Hisakazu Kikuchi[1], and Takashi Saku[2]

[1]Department of Electrical and Electronic Engineering, Graduate School of Science and Technology,
[2]Division of Oral Pathology, Graduate School of Medical and Dental Sciences, Niigata University, Japan
mustafa@telecom0.eng.niigata-u.ac.jp, kikuchi@eng.niigata-u.ac.jp, tsaku@dent.niigata-u.ac.jp

ABSTRACT

In this work we will present an automated image processing method for quantification of microscopic oral tissue images. The method provides quantitative measurements developed for the better histological characterization in the hematoxylin-eosin stained microscopic images. The rete processes shape analysis is targeted in this study and considered to be a key feature that differentiates different grades in oral borderline malignancies. The method has been successful in the precise extraction of the desired histological feature. The method provided feature, which was of definite value, to compare the drop shaped roughness between the best pair match neighboring rete ridge units and aid to pathologists.

Keywords: Oral cancer, rete processes, shape factors.

1. INTRODUCTION

It is still difficult to make proper pathological diagnosis of oral borderline malignancies only by hematoxylin-eosin stained specimens [1]. Diagnostic tools can be developed for objective analysis of microscopic images of oral tissues. The oral dysplasia and carcinoma in-situ (CIS) are two different grades located in the borderline malignancies of the oral mucosa. Those two grades are very similar to each other and it is difficult to distinguish between them and they often lead to considerable variability [2], [3]. It is important to note that different grades will necessitate different treatments, making such diagnostic decision a difficult responsibility for pathologists.
In this paper we present a new automatic method that shows the distinction of CIS from dysplastic epithelia based on drop shaped variation findings of the rete processes rete ridge units. Our approach can be divided into several well-defined stages, illustrated in Figure 1. After image acquisition using high resolution digital camera, color segmentation in the HSI color space is applied in order to separate the epithelium region from the rest of the image. After segmentation, the main pixels at the border of objects are defined and referred to as edgels. Next, a morphological operations based on dilation followed by thinning morphological operations are applied to connect the pixels. A chain code is designed to trace the boundaries of individual rete processes. The details of the method is described in Section III, while Section II will give some important information about the histological characteristics of normal and malignant tissues used to specify the criteria of malignancy that has been considered. Section IV describes the quantification results and Section V will discuss some specific points about this method. Finally, some concluding remarks are given in Section VI.

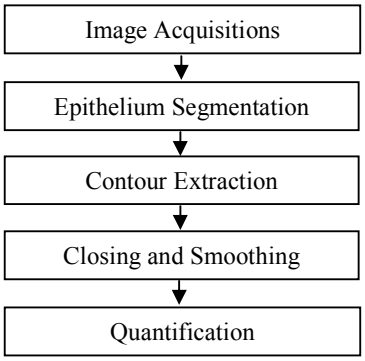
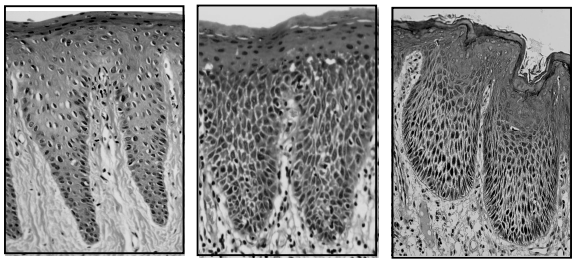


Figure 1. Schematic structure of the method.

2. HISTOLOGICAL FEATURES OF BORDERLINE MALIGNANCIES

This section explains some basic concepts of histopathological diagnosis of the oral mucosa; however, it is not the aim of this paper to expose it in details that can be found in [1]. Here, we are going to highlight to the rete processes and their importance for diagnosis of borderline malignancies of the oral mucosa. The rete processes (also known as basement membrane) is a histological key feature at the tissue level highly considered by oral pathologists for differentiating different grades of the oral mucosa. The rete processes can be defined as the edge segment that separate the epithelium region from the connective tissue and each unit of the rete processes is known as rete ridge. In contrast to normal and malignant oral tissues, normal tissues are characterized by pointed shape rete ridges, while malignant tissues are characterized by drop-shaped rete ridges. Those charac-

terizations were basically indicated by the WHO histopathological classification for epithelial dysplasia and carcinoma in-situ [4]. When normal oral tissue moves towards malignant transformation, its rete processes will start to be enlarged toward the connective tissue performing a drop-shaped like rete ridges. The drop-shaped is, therefore, a strong malignant sign. Figure 2 shows some different drop shaped rete ridges selected from well pre-diagnosed samples of the oral mucosa agreed upon by different pathologists.



(a) (b) (c)

Figure 2. Rete ridge samples of different oral grades. (a) Normal. (b) Dysplasia. (c) CIS.

The drop shaped of the rete ridges in dysplasia and CIS can be obtained in different roughness shapes making such diagnostic decision a hard task to be solved by pathologists' eyes. In this paper, we are going to investigate the drop shaped variation findings of the rete ridges using appropriate shape factors. In practice histology, it is common to observe two or more neighboring rete ridges similar to each other at a glance. Our method is based on shape comparison between the best pair match neighboring rete ridges as illustrated in Figure 3.

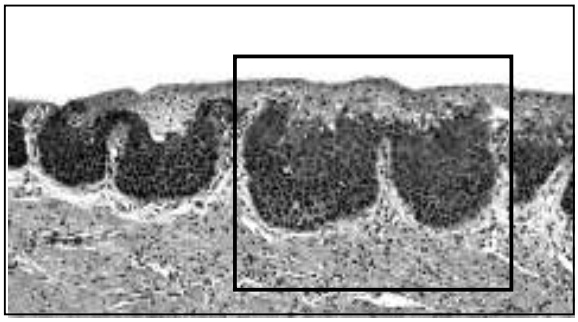


Figure 3. Best pair matches selection method.

Therefore, the precise extraction of the rete processes edge segments is needed in this study and will be explained in the next section followed by a shape quantification method. The obtained values will help pathologists to compare the degree of variation between the neighboring rete ridges to distinguish CIS from dysplastic epithelia.

3. MATERIALS AND METHODS

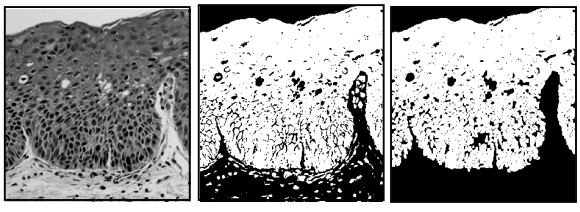
3.1 ORIGIN AND ACQUISITION

In this study, all biopsies were obtained using the same tissue processing. Starting with the fixation procedure that is commonly used to remove and keep the legion part followed by a sectioning procedure using microtome. The microtome is fixed to slice the tissue into 5-micrometer thickness. This very thin tissue is transparent and, therefore, hematoxylin and eosin (H&E) stains are added to emphasize the different parts of the tissue. The H&E staining is a well defined protocol used at different laboratories and can be found in [5]. The technical equipment used for the image acquisition included a high quality optical microscope and a high resolution camera (Nikon DXM1200C). Every image was acquired with the same magnifying factor on the microscope with immersion. Manual interface was needed to adjust the intensity in such a way the digitized image became visually accepted by a specialist. Our quantification method is independent of the minor acquisition and staining artifacts conditions and this will be discussed in Section V.

3.2 SEGMENTATION

In this work, we are interested in the rete processes and its rete ridges shape variations as a malignancy feature. Rete processes is an edge segment that can be obtained from the boundary pixels of the epithelium region, therefore, it is essential to segment the epithelium region from the rest of the image. Segmentation is a crucial step in this study with some challenges. The challenges are mainly from staining artifacts, lighting acquisition conditions, and undesired touching objects.

Epithelium segmentation was targeted in several studies [6], here it was obtained in the HSI color space based on the assumption that the epithelium region has a lower saturation than the connective tissue. A global thresholding based on the Otsu method [7] has been used to determine the saturation value applied to the S component of the HSI color space. HSI color space is good at applying color in terms that are practical for human interpretation. It is believed that the epithelium is the biggest region in the image and it was selected. Figure 4 illustrates the mentioned segmentation process.

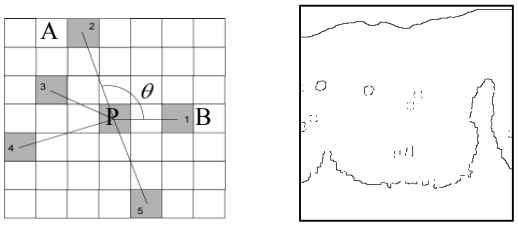


(a) (b) (c)

Figure 4. (a) Original. (b) Saturation component of HSI. (c) Biggest region selection.

3.3 EDGEL PIXEL SELECTION

Seeking for a precise extraction for the contour pixels of the epithelium that can be representing the rete processes, we used a wide angle edgel method that can be found in [8]. Edgels are an edge pixels located at the boundary of object. When someone takes a position at the front edge of an object, he or she will see nothing in the field of view through the line-of-sight distance as it is described in Figure 5 (a).



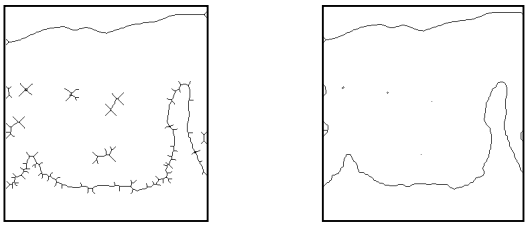
(a) (b)

Figure 5. (a) Wide angle edgel method. (b) The obtained result from the wide angle edgel method.

P is an edgel element with a field of view θ with respect to its two surrounding neighborhood pixels A and B. Inside a window of size 7x7, only those pixels with a field of view of 90 degrees or higher to any of its surrounding pixels were selected as edgel pixel. Figure 5 (b) shows the obtained result.

3.4 MORPHOLOGICAL OPERATIONS

The obtained image from the wide angle edgel method described in the previous section is made by sparse edgel pixels. A morphological operation based on dilation with 4x4 square structural element followed by thinning skeletonization operation has been applied [9]. The spurious arcs were then removed from the image. Figure 6 (b) shows the output image of the morphological operations.



(a) (b)

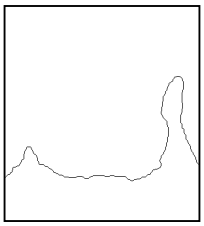
Figure 6. (a) Skeletonized image. (b) Result after spur removal.

3.5 CHAIN CODING

We designed a chain code for our application in order to trace the connected pixels contour that represents the rete processes. The starting point of the chain code is always determined by the pixel located in the bottom most of the first left columns of the image that can be identified by the highest value of the y-axis coordinate. Going forward to the right side end of the image can be obtained by checking all the possibilities after every movement as illustrated in Figure 7 (a).

x7	*x6*	*x5*
x8	*p*	*x4*
x1	*x2*	*x3*

(a)



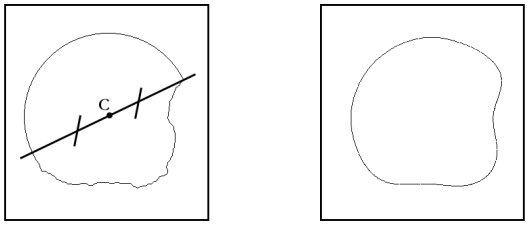
(b)

Figure 7. (a) 3x3 chain code movement priority. (b) The obtained result from chain coding.

In a 3x3 window size, the movement priority of the chain code is always given to the next south side pixel. The obtained rete processes edge segment is shown in Figure 7 (b).

3.6 CLOSING AND SMOOTHING

A half circle arc has been added to the extracted rete ridge curve in order to obtain a closed loop shape representing one rete ridge unit. The highest two peak points located at both side terminals of the rete ridge were identified and from a central point 'C' located at the half distance connect the two peak points a half circle arc has been added and we shall refer to it as a *hat*. The shape analysis factor used in this study is scale variance and, thus, different hat scales added to different rete ridges will have the same effect to the shape quantified results. Figure 8 (a) demonstrates the attached half circle arc.



(a) (b)

Figure 8. (a) Half circle arc attachment. (b) The smoothed rete contour using 2% of the descriptors.

The obtained close loop contour is then smoothed by using 2% of the total descriptors in order to obtain a more regular shape that could represent sufficient levels of reality in the histological image as shown in Figure 8 (b).

4. QUANTIFICATION

As we mentioned in Section III, the aim of this study is to quantify the drop shaped roughness of the rete ridges using appropriate shape factors [10], [11] that can be used to compare between the best pair match neighboring rete ridge units. Shape circularity and roundness were used to measure the shape of the rete ridge units.
Shape circularity and roundness are defined as:

$$Circularity = \frac{4\pi \times Area}{Perimeter^2}$$

$$Roundness = \frac{4 \times Area}{\pi \times MaxDiameter^2}$$

The circularity formula shows its sensitivity to the contour variations while the roundness formula shows its sensitivity to the depth variations that always accompanying the rete ridge growth and is highly considered by pathologists as a malignancy sign.
Table 1 presents the results for the 6 test images. Each tissue sample image contains one best pair match neighboring rete units indicated as left and right. The absolute difference between the left and right units is then computed.
The obtained results showed that difference between the left and right rete ridges in normal oral tissues consisted of small values and the difference increased with dysplasia and becoming highest in CIS.

Table 1. Quantification of the tested images.

Tissue Sample	Circularity			Roundness		
	Left	Difference	Right	Left	Difference	Right
Norm1	0.66	**0.01**	0.65	0.43	**0.01**	0.44
Norm2	0.45	**0**	0.45	0.24	**0.02**	0.26
Dys1	0.86	**0.04**	0.82	0.79	**0.04**	0.83
Dys2	0.75	**0.05**	0.80	0.59	**0.08**	0.67
CIS1	0.80	**0.15**	0.65	0.63	**0.18**	0.45
CIS2	0.64	**0.14**	0.78	0.41	**0.18**	0.59

5. DISCUSSION

The method can be applied for different locations of oral borderline malignancies such as lips, tongue and buccal mucosa, and where the drop shaped rete ridge is a malignancy sign. The method is based on segmentation of the saturation component, which has a good robustness against the staining artifacts and the lighting acquisition conditions. The H&E staining protocol is well defined and only minor changes can appear at different images obtained from different laboratories.
The quantification method is insensitive to the scale variance; however, the tested images were obtained under the same magnification.

6. CONCLUSION

An automated image analysis method has been developed for quantifying the rete processes units in the H&E stained microscopic images of the oral mucosa.
The shape factors, including measures of circularity and roundness were used to examine the drop shaped roughness of the rete ridge units. Those shape factors were used to compare the difference between the neighboring rete units. The obtained results showed that difference in shape factors for the best pair match neighboring rete units in CIS where higher than those in epithelial dysplasia.

7. REFERENCES

[1] Saku T, et al. *Guidelines for Histopathological Diagnosis of Borderline Malignancies of the Oral Mucosa*, Japanese Society of Oral Pathology, Yamazaki Publishing, Niigata, 2005.

[2] A. Andrion, et al. Malignant mesothelioma of the pleura: interobserver variability, *Journal of Clinical. Pathology* 48(1995) 856-860.

[3] S.M. Ismail, A.B. Colclough, J.S. Dinnen, D. Eakins, D.M. Evans, E. Gradwell, J.P. O'Sullivan, J.M. Summerell, R.G. Necombe, Observer variation in histopathological diagnosis and grading of cervical intraepithelial neoplasia, *British Medical Journal* 298(1989) 707-710.

[4] *The WHO Histopathological Typing of Cancer and Precancer of the Oral mucosa.* 2nd edition, 1997.

[5] [www.ihcworld.com/_protocols/special_stains/h&e_elis.com]

[6] DLandini G, Othman IE: Estimation of tissue layer level by sequential morphological reconstruction. *Journal of Microscopy* 2003, 209(2): 118-125.

[7] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Systems, Man, and Cybernetics,* vol. SMC-9, no. 1, pp. 62–66, Jan.1979.

[8] M. Petrou and P. Sevilla, *Image Processing; Dealing with Texture*, John Wiley, 2006.

[9] L. Lam, S.-W. Lee, and C. Y. Suen, "Thinning methodologies—a comprehensive survey" *IEEE Trans. Pattern Analysis and Machine Intell*igence, vol. 14, no.9, pp. 869–885, Sep. 1992.

[10] Rangaraj M. Rangayyan, *Biomedical Image Analysis*, CRC Press, 2005.

[11] Jyrki Selinummi "Automated Quantitative Analysis of Color-Stained Cell Images" Master of Science Thesis, Tampere University of Technology. Finland.

ON THE STATISTICAL ACCURACY OF STOCHASTIC SIMULATION ALGORITHMS IMPLEMENTED IN DIZZY

Werner Sandmann and Christian Maier

University of Bamberg
Department of Information Systems and Applied Computer Science
Feldkirchenstr. 21, D-96045 Bamberg, Germany
werner.sandmann@uni-bamberg.de

ABSTRACT

Stochastic simulation is in widespread use for analyzing biological pathways. Due to the limited efficiency of a straightforward direct implementation such as the Gillespie algorithm, various improvements and approximate algorithms have been developed. For user-friendliness it is important to have efficient implementations available in software tools. Another important issue is the statistical accuracy of simulation results in terms of variances, confidence intervals, or related measures. We address the problem of computing such statistics for Dizzy, a software tool that has been recommended in a recent study of the user-friendliness of software tools. Therefore, a mathematical framework for statistical output analysis of simulation results is provided, the need for statistics as well as the lack of user support in actually obtaining such statistics with Dizzy and other tools is emphasized, and recommendations for future extensions of software tools are given.

1. INTRODUCTION

In the discrete-state stochastic approach to coupled chemical reactions, the system state is defined by the population of all involved molecular species $S_1, \ldots, S_d$. The time evolution is described by a continuous-time Markov chain $(X(t))_{t\geq 0}$ where $X(t) = (X_1(t), \ldots, X_d(t))$ and $X_i(t)$ is the number of molecules of species S_i present at time t. The Kolmogorov differential equations governing the system dynamics are expressed via the chemical master equation (CME), which is a system of ordinary differential equations (ODEs) where the variables are transient (time-dependent) state probabilities. Since the CME is usually difficult to solve for large or stiff models, stochastic simulation is often applied to analyze biological pathways. Rather than directly solving the CME, realizations of Markov chain trajectories (sample paths) are generated. Stochastically exact trajectory generation is often referred to as the Gillespie algorithm in the context of chemical reactions as Gillespie [2, 3] introduced the terminology of the CME and thereby proposed to use stochastic simulation for system analysis. However, direct simulation where each single reaction is explicitly simulated is exceedingly slow. Therefore, various modified implementations as well as accelerated approximate methods for enhanced trajectory generation have been proposed.

A major drawback of stochastic simulation that has not received much attention in systems biology so far is the statistical uncertainty due to the random nature of simulation results. Despite the fact that Gillespie's algorithm is termed exact, a stochastic simulation can never be exact. The exactness of Gillespie's algorithm is only "in the sense that it takes full account of the fluctuations and correlations" [3] of reactions within a single simulation run. It is common sense in stochastic simulation theory that one should never rely on a single simulation run and Gillespie already mentioned that it is "necessary to make several simulation runs from time 0 to the chosen time t, all identical with each other except for the initialization of the random number generator". In fact, the reliability of simulation results strongly depends on a sufficiently large number of simulation runs, where an explanation of the meaning of a sufficiently large number and the determination of that number has to be carefully done in terms of mathematical statistics. Even with approximate methods for accelerated trajectory generation still a large number of trajectories is required in order to obtain reliable and meaningful results with acceptable statistical accuracy. Hence, in either case stochastic simulation is computationally expensive and can only provide statistical estimates. Mathematically, it constitutes a statistical estimation procedure implying that the results are subject to statistical uncertainty.

An important point is tool support such that stochastic simulation algorithms can be applied by practitioners who need not be experts in stochastic simulation. Recently, Mäkiraatikka et al [5] studied the user-friendliness of software tools and among those studied, all of which had a couple of shortcomings, they recommended Dizzy [6], cf. http://magnet.systemsbiology.net/software/Dizzy.

We address the statistical accuracy of stochastic simulations. When we started our study, the initial intention was to figure out how far accelerated generation of single trajectories comes at the prize of an increased number of trajectories necessary to provide a certain statistical accuracy. This obviously requires an appropriate framework within which this accuracy is measured. It turned out that

currently neither Dizzy nor any of the other tools we are aware of do provide any support with regard to statistical output analysis of simulation results. Consequently, the major focus of our work changed towards introducing an appropriate mathematical framework as well as computing relevant statistical measures. While the mathematical framework is completely tool-independent, we discuss the computation and the further processing of necessary information for statistical measures through Dizzy. To come back to our initial intention we obtained several statistics for two test cases but we did not find any essential differences in the statistical accuracy of the stochastic simulation algorithms implemented in Dizzy. However, this is far from being a general result because the lack of support for statistical analysis and the quasi-manually and thus extremely time-consuming computation of statistics prevented more excessive studies and made it even hard to verify the statistical accuracy for relatively small examples. Though we originally aimed at comparing different algorithms, the statistical accuracy of stochastic simulations is an important property for each algorithm in itself. In fact, it is the only mathematical way to investigate the reliability of simulation results. In practice, the number of simulation runs is usually chosen very large but somewhat arbitrarily. Performing many more runs than necessary for a certain desired statistical accuracy means a significant waste of computer time. On the other hand, too less simulation runs render the results meaningless. Hence, it is highly desirable to have some rules giving the required number of simulation runs. In particular, we strongly emphasize the urgent need for integrating statistical output analysis into Dizzy and other tools in a user friendly way and we provide hints and recommendations how this should be done.

The remainder of this paper is organized as follows. In Section 2 we outline how simulation outcomes can be formalized in a unified way such that they yield to statistical analysis. Measures for the statistical analysis are given in Section 3. Then we present our test cases and briefly describe how we computed statistics from the results provided by Dizzy. Finally, we give conclusions and recommendations for future tool extensions.

2. FORMALIZING SIMULATION OUTCOMES

Stochastic simulations are nothing else than statistical estimations using computers. They generate realizations of random variables with the help of random number generators. Similarly as for observations from laboratory experiments, several properties can be derived from the realizations. Thus, from a statistical point of view repeated laboratory experiments and stochastic simulations are equivalent. The only difference is in the way realizations are generated. In a laboratory experiment they are generated within a physical real life environment whereas a stochastic simulation imitates real life environments by using appropriate rules.

In practice, each simulation run is finished at some time and the outcome is a finite sequence of states where state changes are triggered by reactions and several properties can be immediately derived for all species. Such properties can be mathematically described as a function f of the sequence of states. Since outcomes of stochastic simulations are realizations of random variables and functions of random variables are again random variables, the property of interest is also a random variable. We denote it by $Y = f(X(t_0), \ldots, X(t_m))$. Note that although the set of reaction times is countable, yielding a sequence of states, the time differences are in general not equal, i.e. typically $t_{i+1} - t_i \neq t_{j+1} - t_j$ for $i \neq j$. The random variable Y may be the number of molecules of a species at some (not necessarily reaction) time t in which case it is simply the projection to the relevant component of $X(t)$. It may also be the mean number of molecules, the time until a specific number of molecules has been reached or exhausted. In general, Y might be any imageable property that can be determined from a sample path. Each time a realization is generated, it is different in general. Also it will rarely ever exactly coincide with the "true" value Y. Statistical methods are required to assure that no wrong conclusions are drawn from accidentally untypical experiments. More precisely, a statistical estimation procedure must be executed up to some predefined accuracy.

According to classical statistics one builds an *estimator* from several (say N) stochastically independent and identically distributed (iid) random variables, generates N realizations via experiments, and estimates the property of interest by the resulting realization of the estimator. Since an estimator is itself a random variable it follows a probability distribution with mean (expectation), variance, higher moments etc. Hence, it is important to know its fluctuation. The characteristics of the estimator, in particular its variance and measures derived from it, determine the accuracy and the reliability of the estimate.

3. STATISTICAL ACCURACY OF SIMULATIONS

In this section we elaborate on the statistical estimation procedure which is needed and performed in stochastic simulations thereby focusing on the expectation $E[Y]$. We particularly emphasize the large time complexity and the nevertheless remaining inherent uncertainty.

3.1. Point Estimators and Confidence Intervals

Given a sample $Y_1, \ldots, Y_N$, independent and identically distributed as a univariate random variable Y, the natural estimator for $E[Y]$ is the *sample mean*

$$\bar{Y} = \frac{1}{N} \sum_{i=1}^{N} Y_i. \tag{1}$$

It is important to note that the sample mean is an *unbiased* estimator for $E[Y]$, i.e. $E[\bar{Y}] = E[Y]$. Unbiasedness of an estimator is an obviously desirable property, but for more complicated properties than the expectation often not so straightforward to obtain as it might appear. As a simple example note that an unbiased estimator for

the variance $\sigma^2(Y)$ is given by

$$S^2 = \frac{1}{N-1}\sum_{i=1}^{n}\left(Y_i - \bar{Y}\right)^2, \qquad (2)$$

whereas the probably first suggestion to divide the sum by N instead of $N-1$ yields a biased estimator.

As a random variable, an estimator is subject to statistical uncertainty, and the question arising after an estimator has been chosen is that of accuracy or reliability in a statistical sense. Unbiasedness is not a sufficient criterion to assure satisfiable accuracy. In addition the estimator's variance is of major importance. In fact, what is needed to make proper statements on the accuracy, in particular dependent on N, is a *confidence interval*.

A confidence interval is a random (dependent on the random sample) interval that contains the property of interest with some predefined probability $1-\alpha$, where $1-\alpha$ is called the *confidence level*, which is in practice usually chosen as 90%, 95% or 99%. According to the central limit theorem, for sufficiently large N classical statistics gives us the confidence interval

$$C = \left[\bar{Y} - z_{1-\alpha/2}\sqrt{\frac{S^2}{N}}\ ,\ \bar{Y} + z_{1-\alpha/2}\sqrt{\frac{S^2}{N}}\right] \qquad (3)$$

where $z_{1-\alpha/2}$ denotes the $1-\alpha/2$ quantile of the standard normal distribution. An important point is how to interpret confidence intervals. As explained above, experiments generate realizations of all random variables involved in the estimation procedure yielding specific values called estimates. In particular, for a given set of realizations $y_1, \ldots, y_N$ one gets a realization of the confidence interval where endpoints are numerical values and the confidence interval realization either contains $E[Y]$ or not. Thus, there is nothing probabilistic *after* the realizations have been obtained and the endpoints have been accordingly set to numerical values. It is a wrong interpretation that each single confidence interval realization contains $E[Y]$ with probability $1-\alpha$. The correct interpretation is that if one constructs a large number of $100\cdot(1-\alpha)$% confidence interval realizations, each based on N experiments, the proportion (*coverage*) of those that contain (cover) $E[Y]$ is $1-\alpha$. A direct consequence of the correct interpretation of confidence intervals is that one might obtain confidence interval realizations that do not contain $E[Y]$ at all.

3.2. Required Number of Simulation Runs

The width of the confidence interval suggests the amount of variability in the estimated value. As the interval is symmetric meaning that $\bar{Y}$ is the midpoint, it is sufficient to consider the confidence interval half width. In non-simulative computations the relative error is most often more meaningful than the absolute error. Similarly, the relative half width of the confidence interval is an appropriate measure of simulation accuracy.

In iterative numerical computations one proceeds by iterating up to a given accuracy, more specifically up to a maximum relative error. Analogously, a stochastic simulation can be viewed as a kind of iteration where simulation runs must be generated until the accuracy is sufficient which means until the relative confidence interval half width for a given confidence level is less than a given maximum error bound. Obviously, the number of required simulation runs is not fixed in advance since the realizations of the confidence interval depend on the specific outcomes of the simulation runs. As an expression for the number of simulation runs required to meet a predefined maximum relative error of β and a confidence level of $1-\alpha$ expression (3) yields

$$N \geq \frac{z_{1-\alpha/2}^2 S^2}{\beta^2 \bar{Y}^2} = \frac{z_{1-\alpha/2}^2}{\beta^2}\cdot\frac{S^2}{\bar{Y}^2}. \qquad (4)$$

Since S^2 and $\bar{Y}$ are estimators for the variance and the expectation, respectively, the ratio $S^2/\bar{Y}^2$ is an estimator for $c_Y^2 = \sigma^2/E[Y]^2$, the squared *coefficient of variation* of Y which is sometimes also called the (estimated) *relative error of the estimator* $\bar{Y}$.

Now, we can put specific values for the confidence level and the maximum relative error into expression (4). Taking usual values such as a confidence level of 99% and a maximum relative error of 10% we get $z_{1-\alpha/2} \approx 2.58, \beta = 0.1$, and thus $N \geq 664\cdot c_Y^2$. As we can see N is determined by the squared coefficient of variation which is the reason that in some cases simulation can be very proper whereas in other cases it results either in runtime explosion or unsatisfactory inaccuracy. More precisely, if c_Y^2 is close enough to zero, a moderate number of simulation runs suffice but if c_Y^2 is large, the required amount of simulation runs grows enormously. As an extreme example take a situation where a very small probability γ of some event has to be estimated. Such a probability can be estimated via the expectation of the event's indicator function. Then $c_Y^2 = (1-\gamma)/\gamma$ is extremely large for very small γ. To be more specific, with the accuracy requirements stated above the required number of simulation runs in (4) to estimate a probability of 10^{-9} is $N \geq 6.64\cdot 10^{11}$.

Although the latter example might seem unrealistically at a first glance, there are in fact a lot of situations where exactly this problem occurs. Even if we are not concerned with such extreme cases it must be noted that except for cases where the squared coefficient of variation of the property of interest is close to zero, simulation requires a large amount of computer time and at least as seriously there remains a non-negligible probability of getting a wrong estimate.

4. OBTAINING STATISTICS IN DIZZY

The stochastic simulation algorithms available in Dizzy are Gillespie's direct method [2, 3], the so-called Gibson-Bruck algorithm [1] which is an implementation of an equivalent interpretation of the Markov chain dynamics, and two versions of tau-leaping [4], an approximate multistep approach for accelerated trajectory generation.

Dizzy provides a graphical user interface as well as a command-line interface. Unfortunately, neither of these

interfaces provides any support for statistical analysis. The only related option is to compute steady state fluctuations but with regard to potentially infinite time horizons within a trajectory. We computed all previously introduced statistical measures manually for various parameter sets of two test-cases. The first one, the enzymatic reaction set

$$E + S \underset{c_2}{\overset{c_1}{\rightleftharpoons}} ES \xrightarrow{c_3} E + P \tag{5}$$

is one of the small examples that comes with Dizzy. The second one is a part of the bacteriophage λ pathway, the lysis-lysogeny switch whose reaction kinetics are given in Table 1. As mentioned in the introduction, for these reaction sets we did not find any significant differences in the statistical accuracy of the stochastic simulation algorithms implemented in Dizzy. It does not make much sense to present excessive tables in order to illustrate this. So, also due to lack of space we omit it.

Table 1. Lysis-lysogeny switch in bacteriophage λ

$2X$	$\overset{c_1}{\rightleftharpoons}$	X_2	dimerization
$D + X_2$	$\overset{c_2}{\rightleftharpoons}$	DX_2	binding 1)
$D + X_2$	$\overset{c_3}{\rightleftharpoons}$	DX_2^*	binding 2)
$DX_2 + X_2$	$\overset{c_4}{\rightleftharpoons}$	DX_2X_2	binding 3)
$DX_2^* + X_2$	$\overset{c_5}{\rightleftharpoons}$	DX_2X_2	binding 3)
D	$\xrightarrow{c_s}$	$D + X$	slow transcription
X	$\xrightarrow{c_d}$	$\emptyset$	degradation
DX_2	$\xrightarrow{c_f}$	$DX_2 + X$	enhanced transcription

We were restricted to these rather small examples because all statistics had to be essentially computed manually. Though Dizzy offers the opportunity for performing many independent simulation runs specified as the ensemble size, it does not provide all "subresults" for each run. Three output options are available. The plot option yields, as the name suggests, a plot of the numbers of molecules versus time but gives no numerical values. The other options are tables and their storage where the number of intermediate time points can be specified but for each time point only mean values of molecular numbers averaged over the simulation runs are provided. That is, only sample means are computed without variances, etc. Therefore, we obtained the necessary information for each simulation run one after another. More precisely, for each configuration we performed N single simulation runs by invoking the chosen simulation algorithm N times by hand. The reader may imagine the enormous amount of time wasted. In fact, this way the simulation became interactive in that each simulation run had to be started manually. Fortunately, Dizzy uses fresh random number also when single runs are manually performed one after another and not only when many independent runs are performed automatically. Finally, we proceeded by transferring the outcomes of each run to a statistical software package (S-PLUS) which provided us with the desired statistical measures.

5. CONCLUSIONS AND RECOMMENDATIONS

The statistical accuracy of stochastic simulations is an important but so far largely neglected issue in order to measure the reliability of simulation results. A mathematical framework for unified statistical simulation output analysis can be given by appropriately formalizing simulation outcomes and handling the property of interest, formally expressed as a function of random variables which is itself a random variable, by means of classical statistics. User support for statistical analysis is lacking in current software tools for simulating biological pathways. As statistical accuracy is essential for meaningful results, such a user support is highly desirable and strongly recommended. Hence, future extensions of software tools should integrate the methods outlined here. It seems that this should not be too difficult to implement and rather straightforward if the property of interest is related to the number of molecules at one or more specific times. In such cases, all required information is actually computed within a stochastic simulation and it remains to appropriately process it and provide it to the user. Another recommended feature is to offer the user the opportunity to prespecify the desired statistical accuracy, e.g. in terms of relative errors or relative confidence interval half-width, and automatically perform simulation runs until this accuracy is reached. It would be also of interest to provide a more flexible specification of the time horizon for each simulation run. Properties of practical interest are times until the molecules of certain species are exhausted or certain subsets of the state space are reached. Accordingly, users should be allowed to specify such terminating conditions for simulation runs.

6. REFERENCES

[1] M. A. Gibson and J. Bruck. Efficient exact stochastic simulation of chemical systems with many species and many channels. *J. Phys. Chem.*, 104:1876–1889, 2000.

[2] D. T. Gillespie. A general method for numerically simulating the time evolution of coupled chemical reactions. *J. Comp. Phys.*, 22:403–434, 1976.

[3] D. T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.*, 71(25):2340–2361, 1977.

[4] D. T. Gillespie and L. R. Petzold. Improved leap-size selection for accelerated stochastic simulation. *J. Chem. Phys.*, 119:8229–8234, 2003.

[5] E. Mäkiraatikka et al. Stochastic simulation tools for cell signaling: survey, evaluation and quantitative analysis. In *Proc. 2nd Conf. Foundations of Systems Biology in Engineering*, pages 171–176, 2007.

[6] S. Ramsey, D. Orrell, and H. Boulouri. Dizzy: Stochastic simulation of large-scale genetic regulatory networks. *Journal of Bioinformatics and Computational Biology*, 3(2):415–436, 2005.

USING NEIGHBORHOOD GRAPHS FOR THE INVESTIGATION OF *E. COLI* GENE CLUSTERS

Theresa Scharl[1,2] *and Friedrich Leisch*[3]

[1]Department of Statistics and Probability Theory, Vienna University of Technology,
Wiedner Hauptstraße 8-10, A-1040 Vienna, Austria
[2]Department of Biotechnology, University of Natural Resources and Applied Life Sciences,
Muthgasse 18, A-1190 Vienna, Austria
[3]Department of Statistics, University of Munich,
Ludwigstraße 33, D-80539 München, Germany
theresa.scharl@ci.tuwien.ac.at, friedrich.leisch@stat.uni-muenchen.de

ABSTRACT

Clustering is commonly used in the analysis of gene expression data to find groups of co–expressed genes. The definition of gene clusters is not very clear as genetic interactions are extremely complex. For this reason the relationship between clusters is very important as co–expressed genes can end up in different clusters. The neighborhood graph is a useful tool to visualize the cluster structure. In this paper the R package `gcExplorer` is presented which is an interactive toolbox for the exploration of gene clusters. Additional information about the gene clusters like the annotation of genes to functional groups (e.g., GO categories) can easily be investigated. The new visualization toolbox is demonstrated on microarray data from *E. coli*.

1. INTRODUCTION

Clusters of co–expressed genes can help to discover potentially co–regulated genes or association to conditions under investigation. Additionally they might suggest pathways or interactions between genes. Cluster analysis is frequently used for the first investigation of a microarray dataset before actually focussing on particular functional subgroups of interest. Gene interactions are extremely complex and the definition of gene clusters is not clear. Further, gene expression data are very noisy and co–expressed genes can easily end up in different clusters. In this context cluster analysis is used as vector quantization as no clear density clusters exist. The data is divided into artificial subsets where the relationship between clusters plays an important role.

The visualization of the cluster structure is important in order to investigate the relationships between clusters. The display of cluster results is very helpful to make cluster analysis useful for practitioners. The Neighborhood graph [1] can be used to display distances between clusters for centroid–based cluster solutions. Microarray data are high–dimensional and complex datasets yielding a high number of clusters. As the linear projection of the data into two dimensions using for example LDA does not scale well in the number of clusters there is a need for new visualization techniques which can handle this situation [2].

In this paper the R package `gcExplorer` is presented which is an interactive toolbox for the exploration of gene clusters. The layout algorithms implemented in the open source graph visualization software Graphviz are used for non–linear arrangement of the clusters. `gcExplorer` contains several possibilities to investigate gene clusters. Further properties of the clusters are included in the neighborhood graph, e.g., cluster size or cluster tightness. Additionally external knowledge from differential expression analysis or functional grouping can be used to investigate the data. `gcExplorer` is currently available at the homepage of the first author (`http://www.ci.tuwien.ac.at/~scharl/Software/`) and will be released as an R package ([3], `http://www.R-project.org`) soon.

The functionality of `gcExplorer` is demonstrated on time–course gene expression data from NCBI Gene Expression Omnibus (GEO, `http://www.ncbi.nlm.nih.gov/geo/`), the *Escherichia Coli* data set with GEO accession number GSE4357-GSE4380 [4]. *E. coli* cells were sampled at several time points (0, 78, 105, 133, 163, 191, 218, 261, 313, 446, 1440 minutes) as they recover from stationary phase versus the Bonner-Vogel medium OD 0.5. After filtering out incomplete and constant observations over time the data set consists of 1672 genes at 11 time points.

2. NEIGHBORHOOD GRAPHS

Neighborhood graphs [1] can be used to visualize cluster solutions of centroid–based cluster algorithms like K–means and PAM or others where clusters can be represented by centroids (e.g., QT–Clust, [5]). For a

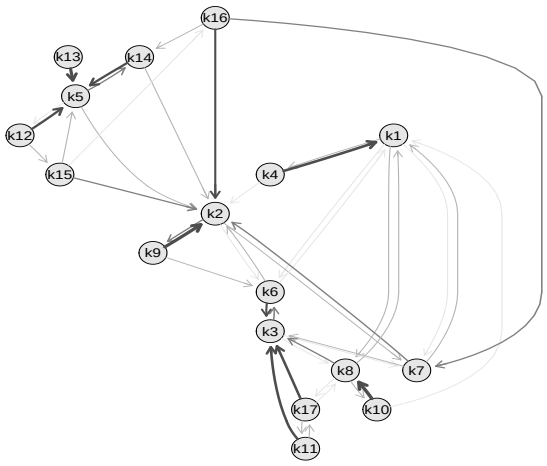


Figure 1. A neighborhood graph of a QT–Clust cluster solution for the *E. coli* data.

given data set $X_N = \{x_1, \ldots, x_N\}$ the distance between points x and y is given by $d(x, y)$, e.g., the Euclidean or absolute distance. $C_K = \{c_1, \ldots, c_K\}$ is a set of centroids and the centroid closest to x is denoted by

$$c(x) = \operatorname*{argmin}_{c \in C_K} d(x, c).$$

Minimizing the average distance between each data point and its closest centroid

$$D(X_n, C_K) = \frac{1}{N} \sum_{n=1}^{N} d(x_n, c(x_n)) \to \min_{C_K}$$

is the task of most cluster algorithms.

Neighborhood graphs use the mean relative distances between points as edge weights in order to measure how separated pairs of clusters are. Hence they display the distance between clusters. In the graph each node corresponds to a cluster centroid and two nodes are connected by an edge if there exists at least one point that has these two as closest and second–closest centroid.

As described above the centroid closest to x is denoted by $c(x)$ and the second closest centroid to x is denoted by

$$\tilde{c}(x) = \operatorname*{argmin}_{c \in C_K \setminus \{c(x)\}} d(x, c).$$

The set of all points where c_k is the closest centroid is given by

$$A_k = \{x_n | c(x_n) = c_k\}.$$

Now the set of all points where c_i is the closest centroid and c_j is second–closest is given by

$$A_{ij} = \{x_n | c(x_n) = c_i, \tilde{c}(x_n) = c_j\}.$$

For each observation x $s(x)$ is defined as

$$s(x) = \frac{2d(x, c(x))}{d(x, c(x)) + d(x, \tilde{c}(x))}.$$

$s(x)$ is small if x is close to its cluster centroid and close to 1 if it is almost equidistant between the two cluster centroids. The average s–value of all points where cluster i is closest and cluster j is second closest can be used as a proximity measure between clusters and as edge weight in the graph.

$$s_{ij} = \begin{cases} |A_i|^{-1} \sum_{x \in |A_{ij}|} s(x), & A_{ij} \neq \emptyset \\ 0, & A_{ij} = \emptyset \end{cases}$$

$|A_i|$ is used in the denominator instead of $|A_{ij}|$ to make sure that a small set A_{ij} consisting only of badly clustered points with large shadow values does not induce large cluster similarity.

3. SOFTWARE

R package `flexclust` [1] is a flexible toolbox for clustering and contains extensible implementations of the K–centroids and QT–Clust algorithm. The plotting method for cluster solutions in `flexclust` is the neighborhood graph using for example LDA for a linear projection of the data into two dimensions. In `gcExplorer` the neighborhood graph is displayed using non-linear arrangement of the nodes (see for example Figure 1). Bioconductor ([6], `http://www.bioconductor.org`) packages `graph` and `Rgraphviz` [7] provide tools for creating, manipulating, and visualizing graphs in R as well as several non–linear layout algorithms.

3.1. Using `gcExplorer`

Now the functionality of the interactive software toolbox `gcExplorer` is demonstrated on publicly available *E. coli* time–course gene expression data. The dataset is clustered using the QT–Clust algorithm by the following R commands

```
> library("gcExplorer")
> library("flexclust")
> data("GSE4363")
> cl1 = qtclust(GSE4363, radius = 3, simple = FALSE)
> gcExplorer(cl1, filt = 0.1)
```

The resulting cluster object consists of 17 clusters and the corresponding neighborhood graph is plotted using function `gcExplorer` (see Figure 1). The graph is simplified by using the argument `filt`. In this case edges between nodes are only drawn if the similarity of a cluster to another cluster is at least 10%. The number of edges pointing from one node to other nodes indicates how distinct the expression profiles are within the corresponding cluster as well as between clusters.

Now there are several possibilities to explore this cluster result. Function `gcExplorer` is an interactive function if `interactive` is set equal to `TRUE` so the clusters can be investigated by clicking on the nodes of the graph. Argument `dev` offers the possibility to choose if each cluster should be opened in a new window or not. The display method for single clusters

is given by the argument `panel.function`. In the case of expression profiles over time function `gcProfile` is used as the plotting function. However, any kind of plotting method can be used instead as well as the display of a cluster in form of an html table with links for each gene to databases like NCBI Entrez Gene (`http://www.ncbi.nlm.nih.gov/sites/entrez?db=gene`). The interactive plotting method can be obtained by the following R commands

```
> gcExplorer(cl1, interactive = TRUE, dev = "many",
+     panel.function = gcProfile)
```

There are several possibilities how to include additional information about the clusters in the representation of nodes. The most simple method is to use color coding, e.g., to color nodes by cluster size or cluster tightness.

3.2. Functional Grouping

The annotation of genes to categories or classes is a very important aspect in the analysis of gene expression data. The genes can for example be mapped to functional groups like Gene Ontology (GO, [8]) classifications or to protein complexes. Gene functions are very complex, therefore genes are usually mapped to multiple classes. In any case the mapping is known a priori and does not depend on the experimental data.

External information about the annotation of genes to functional groups can easily be included in the neighborhood graph, e.g., the accumulation of GO classifications in certain gene clusters can be highlighted in the node representation. In the implementation several functional groupings are included, i.e., GO classifications about Biological Process, Molecular Function and Cellular Component, the GenProtEC ([9], `http://genprotec.mbl.edu/`) classification system for cellular and physiological roles of *E. coli* gene products and some information about operons and regulons from the RegulonDB ([10], `http://regulondb.ccg.unam.mx/`).

The information of interest can be included in the node representation using the corresponding `node.function`. Function `node.go` is used to highlight clusters with accumulation of certain gene functions. The functional group of interest is passed to `node.go` by the argument `node.args`. In this example genes assigned to the GO Biological Process group ("gobp") Metabolism (GO number 8152) are highlighted. This is obtained by the following R commands

```
> gcExplorer(cl1, interactive = TRUE, dev = "many",
+     panel.function = gcProfile,
+     node.function = node.go,
+     node.args = list(gonr = "8152",
+                      source = "gobp"))
```

In Figure 2 a screenshot of an analysis of the *E. coli* data using `gcExplorer` is given. Nodes of clusters containing genes involved in metabolism are highlighted. Clusters 2, 5, 12 and 13 contain a large number of genes related to metabolism. Clusters 1, 3, 9, 14 and 16 contain a few genes related to metabolism. In the top right of the screenshot an html table of cluster 5 is shown containing links to NCBI Entrez Gene. Additionally the expression profiles of several clusters involved in metabolism are shown. The expression profiles are given with the 11 time points on the x–axis and gene expression on the y–axis. A legend containing the corresponding gene symbols is added to each plot.

4. SUMMARY

Cluster analysis is commonly used to find groups of co–regulated genes in a microarray dataset without prior knowledge about the gene functions. However, by clustering expression profiles groups of genes with similar biological function are found. For this reason clustering provides a good initial investigation of the data before actually focussing on groups of genes associated to conditions under investigation. As the definition of gene clusters is not very clear and genetic interactions are extremely complex the relationship between clusters is very important as co–expressed genes can end up in different clusters.

In this paper an interactive toolbox for the investigation of gene clusters was presented. Neighborhood graphs were found useful instruments for the investigation of the underlying cluster structure and for gaining insight into the relationships between clusters. `gcExplorer` is very helpful not only for statisticians but also for practitioners to extract useful information from microarray experiments. It allows not only to visualize the cluster structure, beyond the gene clusters are plotted or shown in html tables with links to databases. Additional properties of the clusters like cluster size or cluster tightness can be highlighted as well as external information like functional grouping. Further extensions of the software are work in progress like the generalization to arbitrary organisms.

5. ACKNOWLEDGMENTS

The project was funded by the Austrian K_{ind}/K_{net} Center of Biopharmaceutical Technology (ACBT).

6. REFERENCES

[1] F. Leisch, "A toolbox for k-centroids cluster analysis," *Computational Statistics and Data Analysis*, vol. 51, pp. 526–544, 2006.

[2] T. Scharl and F. Leisch, "Visualizing gene clusters using neighborhood graphs in R.," *Department of Statistics: Technical Reports*, 2008, http://epub.ub.uni-muenchen.de/2110/.

[3] R Development Core Team, *R: A Language and Environment for Statistical Computing*, R Foun-

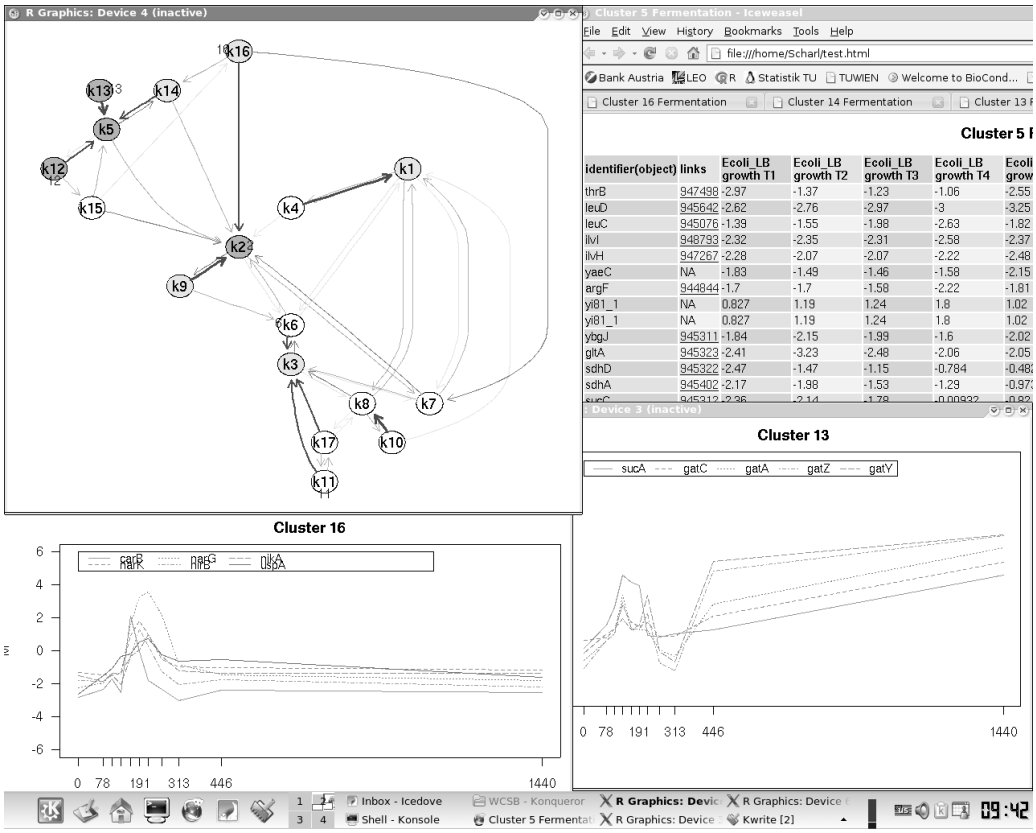


Figure 2. Screenshot of the functionality of `gcExplorer`.

dation for Statistical Computing, Vienna, Austria, 2007, ISBN 3-900051-07-0.

[4] D. P. Sangurdekar, F. Srienc, and A. B. Khodursky, "A classification based framework for quantitative description of large-scale microarray data," *Genome Biology*, vol. 7, 2006.

[5] L. J. Heyer, S. Kruglyak, and S. Yooseph, "Exploring expression data: Identification and analysis of coexpressed genes," *Genome Research*, vol. 9, pp. 1106–1115, 1999.

[6] R. Gentleman, V. J. Carey, W. Huber, R. A. Irizarry, and S. Dudoit, Eds., *Bioinformatics and Computational Biology Solutions Using R and Bioconductor*, Statistics for Biology and Health. Springer-Verlag, New York, 2005, ISBN 978-0-387-25146-2.

[7] V. J. Carey, R. Gentleman, W. Huber, and J. Gentry, "Bioconductor software for graphs," in *Bioinformatics and Computational Biology Solutions Using R and Bioconductor*, R. Gentleman, V. J. Carey, W. Huber, R. A. Irizarry, and S. Dudoit, Eds., Statistics for Biology and Health. Springer-Verlag, New York, 2005, ISBN 978-0-387-25146-2.

[8] The Gene Ontology Consortium, "Gene ontology: tool for the unification of biology.," *Nature Genetics*, vol. 25, pp. 25–29, 2000.

[9] M. Serres, S. Goswami, and M. Riley, "Genprotec: an updated and improved analysis of functions of escherichia coli k-12 proteins.," *Nucleic Acids Res.*, vol. 32, pp. D300–2, 2004.

[10] H. Salgado, S. Gama-Castro, M. Peralta-Gil, E. Diaz-Peredo, F. Sanchez-Solano, A. Santos-Zavaleta, I. Martinez-Flores, V. Jimenez-Jacinto, C. Bonavides-Martinez, J. Segura-Salazar, A. Martinez-Antonio, and J. Collado-Vides, "Regulondb (version 5.0): Escherichia coli k-12 transcriptional regulatory network, operon organization, and growth conditions.," *Nucleic Acids Res.*, vol. 34, pp. D394–7, 2006.

CORRELATION PATTERNS OF CELLULAR GENEALOGIES

Nico Scherf, Ingo Roeder, Ingmar Glauche

Institute for Medical Informatics, Statistics and Epidemiology
University of Leipzig
Haertelstr. 16/18, D-04107 Leipzig, Germany
nico.scherf@ ingo.roeder@ ingmar.glauche@imise.uni-leipzig.de

ABSTRACT

Time lapse video microscopy facilitates the observation and analysis of individual cell fates. Information on cellular development, divisional history, and differentiation are naturally comprised into a pedigree-like structure, denoted as cellular genealogy. Characteristics of the differentiation process are potentially imprinted in these cellular genealogies. Here we study a set of topological measures that are specifically tailored to extract typical correlation patterns between characteristic cell death events and to relate them to relevant biological processes. Using a single-cell based, mathematical model of hematopoietic stem cell organization we compare differentiation strategies that are based on either the instructive or the selective action of cell fate specific signals and show their consequences on the level of the cellular genealogies.

1. INTRODUCTION

Although somatic stem cells play a central role in tissue maintenance and repair as well as in cancer initiation and progression, many questions about their organizational principles are still unresolved. For example, it is an open question whether asymmetric cell division events play a functional role for the maintenance of the stem cell pool or if the observed developmental patterns are induced by asymmetric cell fates which are not necessarily linked to the cell division event [1, 2]. Moreover, the nature of multipotency as well as of the dynamic processes that initiate and regulate the specification of the diversity of functional cells (lineage specification) is only insufficiently understood [3, 4]. In particular, there are reports that lineage specification is an *instructive* process in which a combination of cytokines and cell fate specific signals influences the gene expression pattern of an undifferentiated cell such that certain lineages are promoted whereas others are not. In contrast, it has been argued that lineage specification is *selective* in the sense that the intrinsic influence on the gene expression pattern is negligible, but the regulation occurs on the level of differential survival signals. In the latter setting, cytokines promote the survival of certain lineages whereas cell determined towards other lineages are not supported and consequently undergo cell death [1, 3, 5].

Experimental approaches based on cell population averages are mostly not able to answer the outlined questions for two reasons: first, stem cell populations have a certain, hardly reducible, degree of inherent heterogeneity which makes it extremely difficult to initiate cultures of identical and synchronized cells. Second, the population approaches do not capture the temporal evolution and chronology of cellular development as it occurs within a single cell. However, it is precisely the development of each individual cell and its progeny that represents a possible realization of the developmental sequence and retains much of the necessary information: on the correlations between differentiation and cell cycle regulation, on the timing of lineage specification processes and cell death events as well as on the role of asymmetric developments.

The application of time lapse video microscopy for the analysis of cell cultures facilitates the tracing of single cells, including all its progeny over extended time periods up to several days. This comprises the temporal analysis of cell specific parameters like morphology, cell cycle time, motility or the occurrence of cell death within the population context. All the different information on cellular development, divisional history, and differentiation can be comprised into a pedigree-like structure in which the founder cell represents the root and the progeny is arranged in the branches. These pedigrees are referred to as cellular genealogies.

Although the numerical methods are still under development, the automised analysis of time lapse videos from cell cultures will soon allow the simultaneous tracking of a multitude of root cells. The expected resulting cellular genealogies represent unique examples of the developmental sequence as they occur under the particular assay conditions. Statistical analysis of these cellular genealogies can reveal typical patterns of cellular development as they are imprinted in the topology. The main objective of this work is the application of a set of recently proposed topological measures [9] to characterize the differences in the cellular genealogies that have been derived using either the *instructive* or the *selective* mode of lineage specification. Since difficulties in the automatic identification and tracing of single cells in current image-processing techniques still limit the availability of experimentally derived cellular genealogies, we use simulated in silico cell cul-

tures in order to approach the stated question. In particular, we obtain cellular genealogies from a single-cell based computer-model of hematopoietic stem cell organization which is able to describe self-renewal, differentiation and lineage specification within heterogeneous cell populations and which has been verified for different in vivo and in vitro situations [6, 7, 8]. Based on this model we show how changes in the particular mode of lineage specification (*instructive* vs. *selective*) influence the topology of the cellular genealogies.

2. METHODS

Characterization of cellular genealogies. Cellular genealogies are derived from the tracking of a single, specified cell object (root cell) and its entire clonal offspring.

Technically, a cellular genealogy is an unordered tree graph $\mathcal{G} = \{\mathcal{C}, \mathcal{D}\}$ in which the edges $\mathcal{C} = \{c_i;\, i = 1, \ldots, N\}$ represent cells and the branching points $\mathcal{D} = \{d_i;\, i = 1, \ldots, m\}$ represent division events. Unordered trees are characterized as trees in which the parent-daughter relationship is significant, but the order among the two daughter cells is not relevant. Each genealogy $\mathcal{G}$ is uniquely identified by its root cell $c_0 \in \mathcal{C}^0$ which is the cell that had been chosen as the initial cell of the tracking process. Within such a structure cells are ordered into subset $\mathcal{C}^g$ according to their generation g, starting with the root cell $c_0 \in \mathcal{C}^0$ and followed by the daughter cells in the first to the gth generation ($c_i \in \mathcal{C}^1, \mathcal{C}^2, \ldots$). To each cell c_i belongs either a subsequent division event d_j, giving rise to two daughter cells ($c_i \in \mathcal{C}^{\mathrm{div}}$, with $\mathcal{C}^{\mathrm{div}}$ representing the subset of all cells which undergo division), or the cell's existence terminates without a further division either by cell death ($c_i \in \mathcal{C}^{\mathrm{death}}$, with $\mathcal{C}^{\mathrm{death}}$ representing the subset of all cells which die within the observation period) or by termination of the tracking process ($c_i \in \mathcal{C}^{\mathrm{term}}$, with $\mathcal{C}^{\mathrm{term}}$ representing the subset of all cells with censored observation, i.e. no information about future cell fate available). Final cells are termed leaf cells, i.e. $\mathcal{C}^{\mathrm{leaf}} = \mathcal{C}^{\mathrm{death}} \vee \mathcal{C}^{\mathrm{term}}$. The degree of relation r_{pq} between any two cells c_p and c_q is defined as a topological distance which measures the number of divisions between cells c_p and c_q. Daughter cells that share the same parental cell are termed siblings. A schematic representation of a cellular genealogy and an illustration of the distance measure are provided in Figure 1.

The temporal dimension of the tracking process is usually encoded in the length of the edges; however this is an associate information rather than a genuine topological parameter.

Generation of cellular genealogies. Cellular genealogies are generated from a single-cell based, mathematical model of hematopoietic stem cell organization that has been developed in our group [6, 7, 8]. Within the model stem cells are able to reversibly switch between two characteristic states: proliferating and quiescent. Cells that have lost their propensity to change into the quiescent state continue regular cell divisions within a proliferation phase (differentiating cells) and are finally removed from the system after a subsequent maturation phase without further divisions.

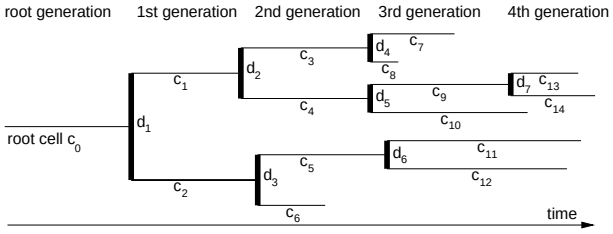


Figure 1. **Schematic sketch of a cellular genealogy.** Within the given five generation genealogy $\mathcal{G}$ the thin horizontal lines represent the cells $c_i \in \mathcal{C}$ whereas the divisions $d_i \in \mathcal{D}$ are marked by the thick vertical bars. The horizontal dimension is time t with the founding root cell c_0 indicated on the left side. The degree of relation r_{pq} between any two cells c_p and c_q is given by the number of divisions between them. For example, cells c_6 and c_8 have a degree of relation $r_{6,8} = 4$ (separated by the divisions d_3, d_1, d_2, and d_4).

In this model lineage specification is described by intracellular propensities for the development of particular lineage fates. Whereas the quiescent state equalizes the lineage specific propensities (uncommitted state), the dominance of one or another lineage is established in a stochastic process during proliferation, indicating the process of lineage commitment. In particular it has been assumed that bi-potent progenitor cells are influenced by the (in silico) conditions such that only one of the two possible lineage fates is promoted and the other one is largely suppressed. For the scope of this work two different modes of lineage specification have been applied. In the *selective* mode, we assume that the cell-intrinsic commitment process is unbiased and promotes the development of both possible lineages. However, there is a targeted cell death process preferentially affecting cells that initiated development towards the suppressed lineage, whereas the preferred lineage is largely unaffected. In contrast, in the *instructive* mode, the cell-intrinsic commitment process is biased towards the preferred lineage. In this scenario, cell death occurs randomly in all cells. The parameter configuration has been chosen such that the population kinetics are indistinguishable for both scenarios (Figure 2).

For the application of a number of statistical measures we compare two sets of 500 cellular genealogies, derived either under the *instructive* or the *selective* mode of lineage specification. In particular, we have initiated two cell populations of 500 initially undifferentiated, bi-potent cells with impaired self-renewal ability which undergo the desired lineage specification process generating one of the two possible cell types. The tracking process for each of the genealogies extends over 200 hours.

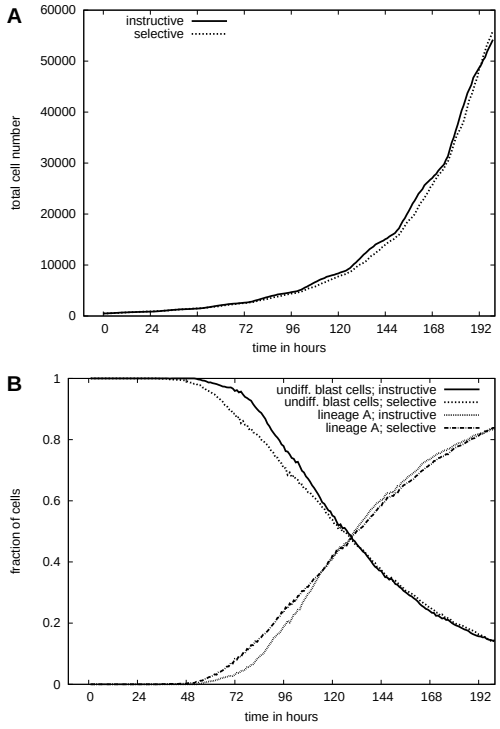


Figure 2. **Population development.** The growth kinetics (A) and the temporal development of the lineage specification (B) are shown for both cell populations (500 initial cells in either the *instructive* or *selective* mode of lineage specification). In (B) the decline of undifferentiated blast cells and the appearance of committed cells of generic type A are provided.

3. RESULTS

Addressing the structural differences in the shapes of the genealogies generated by use of either the *instructive* or the *selective* mode of lineage specification we apply a set of measures that we described previously [9]. In particular the measures on the total number of leaves and the characteristic path lengths are indicators of the expansion. In Figure 3A a boxplot for the distribution of the number of leaves ($L = |\mathcal{C}^{\mathrm{leaf}}|$) is provided for both modes of lineage specification. Since the population kinetics in Figure 2 have been fitted to resemble almost identical growth behavior of the cell cultures, these findings are reflected on the single cell basis, too. Also for the application of weighted Colless' index C^w in Figure 3B , which is a normalized measure of the imbalance within the tree branches, no significant differences in the frequency of occurrence for the 500 sample genealogies can be found.

However, as we have outlined previously, it is the proximity between cell death events which can potentially reveal whether two events are correlated or not. In particular, one assumes, that closely related cells share a similar stage of development, such that these cells undergo similar regulating processes, like induced cell death due to selective lineage specification. In this scenario, cell death events should occur closer to each other, and more often, preferentially in sibling cells.

In Figure 3C we show a boxplot for the distribution of the distance between a cell death event and the closest other cell death event ($r_p = \min_q(r_{pq};\, c_p, c_q \in \mathcal{C}^{\mathrm{death}})$, averaged over all "dead cells" within a particular genealogy. It is obvious that the *selective* mode of lineage specification leads to shorter average minimal distances between such cell death events. Furthermore, we have analyzed the fraction of sibling pairs (two cells directly derived from on common parental cells) in which both cells undergo cell death before they can initiate a further cell division ($c_p, c_q \in \mathcal{C}^{\mathrm{death}};\, c_p, c_q$ are siblings). As the corresponding boxplots in Figure 3D indicate, this fraction is increased for the *selective* mode of lineage specification as compared to the *instructive* mode.

4. DISCUSSION

The availability of time lapse video microscopy and the establishment of efficient image-processing methods will soon allow the "high throughput" tracing of single cells within cell cultures. The interpretation and management of the resulting cellular genealogies is a challenge to experimental and theoretical biologists alike. We showed that cellular genealogies bear a number of additional information which is not accessible on the population level. We demonstrated that the application of suitable measures, such as the average minimal distance between cell death events or the fraction death sibling cells, is appropriate to distinguish different modes of lineage specification. In particular, the *selective* mode of lineage specification is characterized by an increased fraction of death siblings as compared to the *instructive* mode while the average minimal distance between cell death events is considerably reduced.

We are aware that the application of the outlined measures to a set of experimentally derived cellular genealogies does not ultimately allow the identification of the particular mode of lineage specification since the necessary reference scenario is missing. However, we take this as a strong argument in favor of our modeling approach. Given the population kinetics for the cell culture in question, the mathematical model can be adapted using either the *instructive* or the *selective* mode of lineage specification. The resulting genealogies can act as the reference scenarios to which the experimental data is finally compared.

5. ACKNOWLEDGEMENT

The authors thank Dirk Hasenclever and Ronny Lorenz for discussion and programming. This research was funded by European Commission project EuroSyStem (200270).

6. REFERENCES

[1] T. Schroeder, "Tracking hematopoiesis at the single cell level.," *Ann N Y Acad Sci*, vol. 1044, pp. 201–9, 2005.

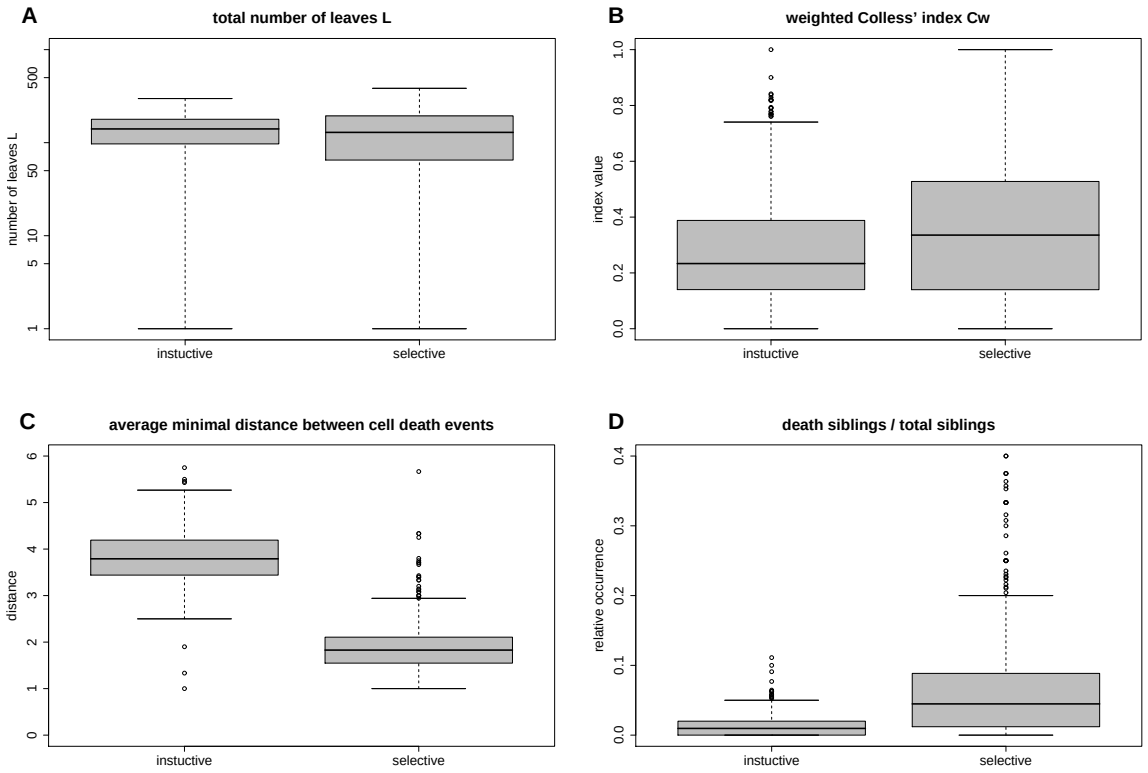


Figure 3. **Measures of tree shape.** Shown are boxplots of the distributions for the topological measures (A) total number of leaves L (shown on a logarithmic scale), (B) weighted Colless index C^w, (C) minimal distance between cell death events, (D) fraction of death siblings. Median values are shown by the thick bars, boxes correspond to the first and third quartile. Whiskers extend to the most extreme data point which is no more than 1.5 times the interquartile range from the box.

[2] I. Roeder and R. Lorenz, "Asymmetry of stem cell fate and the potential impact of the niche observations, simulations, and interpretations.," *Stem Cell Rev*, vol. 2, no. 3, pp. 171–80, 2006.

[3] S. J. Morrison, N. M. Shah, and D. J. Anderson, "Regulatory mechanisms in stem cell biology.," *Cell*, vol. 88, no. 3, pp. 287–298, 1997.

[4] S. Soneji, S. Huang, M. Loose, I. J. Donaldson, R. Patient, B. Göttgens, T. Enver, and G. May, "Inference, validation, and dynamic modeling of transcription networks in multipotent hematopoietic cells.," *Ann N Y Acad Sci*, vol. 1106, pp. 30–40, 2007.

[5] M. A. Rieger and T. Schroeder, "Exploring hematopoiesis at single cell resolution.," *Cells Tissues Organs*, Jan 2008.

[6] I. Roeder and M. Loeffler, "A novel dynamic model of hematopoietic stem cell organization based on the concept of within-tissue plasticity," *Exp. Hematol.*, vol. 30, no. 8, pp. 853–861, 2002.

[7] I. Roeder, M. Horn, I. Glauche, A. Hochhaus, M. C. Mueller, and M. Loeffler, "Dynamic modeling of imatinib-treated chronic myeloid leukemia: functional insights and clinical implications.," *Nat Med*, vol. 12, no. 10, pp. 1181–1184, 2006.

[8] I. Glauche, M. Cross, M. Loeffler, and I. Roeder, "Lineage specification of hematopoietic stem cells: Mathematical modeling and biological implications.," *Stem Cells*, vol. 25, no. 7, pp. 1791–1799, 2007.

[9] I. Glauche, R. Lorenz, D. Hasenclever, and I. Roeder, "A novel view on stem cell development: Analyzing the shape of cellular genealogies.," accepted for publication in *Cell Proliferation*, 2008.

AUTOMATIC IDENTIFICATION AND QUANTIFICATION OF METABOLITES IN ^{1}H-NMR MEASUREMENTS

F.-M. Schleif[1], T. Riemer[2], M. Cross[2] and T. Villmann[1]

[1] Medical Department, Leipzig University, Semmelweisstrasse 22, 04103, Germany
[2] Interdisciplinary Center for Clinical Res., Hematology, Leipzig University, Inselstrasse, 04103, Germany
{schleif,villmann}@informatik.uni-leipzig.de,crossm@medizin.uni-leipzig.de,riemer@uni-leipzig.de

ABSTRACT

Stem cells therapy is currently at the frontier of biomedical research. A better understanding of the metabolism of stem cells is necessary to improve and extend initial promising results. Nuclear Magnetic Resonance Spectroscopy (NMR) allows for a precise measurement of metabolites in cell extracts. The identification and quantification of these metabolites is essential to model cellular metabolic network activity. To meet clinical standards and high throughput demands a full automatic evaluation is required. A NMR signal processing system is presented and initial results for the identification and quantification of metabolites from murine hematopoietic progenitor cell extracts (FDCPmix cells) and simulated spectra are given.

1. INTRODUCTION

Nuclear Magnetic Resonance Spectroscopy (NMR) is one of the most promising techniques for the analyses of complex substances such as cell extracts. One prominent NMR application is metabolite profiling in stem cell biology. NMR spectra are high dimensional functional signals consisting of a multitude of peaks. The peak positions describe the presence of specific chemical compounds in the analysed material while the area of the peaks are quantitative with respect to the amount of this analyte in the substance. To meet clinical standards and high throughput demands a full automatic evaluation is recommended. Here we present the basic methodology of such a system. The paper is organized as follows. First we briefly explain the bio-chemical aspect of the considered studies. Subsequently key elements of the automatic analysis system are described. Thereafter the system behavior is shown in the analysis of synthetic and real metabolite data. The paper is closed by the discussion of the results.

2. MATERIAL AND DATASETS

We consider real and simulated ^1H NMR spectra recorded at 700.15 MHz with $65K$ complex data points. The simulations are done using the gamma-library[1]. It is assumed that each sample is solved in D_2O with DSS added as reference standard set to 0.0 ppm. The simulated data are generated from known spin systems [2] and calibrated by own reference measurements. Thereby we consider the following metabolites Alanine (Ala), Citric Acid (Cit), Glycine (Gly), Lactate (Lac), Malate (Mal), Myo-Inositol (Myo), Serine (Ser) and Succinate (Suc). The biological data are taken from FDCPMix cells cultivated in three different levels of glucose concentration (1 m-mol, 5 m-mol and 25 m-mol) in growth medium as specified in [3]. For each concentration level at least 4 ^{1}H NMR spectra have been recorded.

3. AN AUTOMATIC SYSTEM FOR ^{1}H-NMR MEASUREMENTS

The NMR data of modern spectrometers are usually obtained in the time domain and thereby given as a set of sine/cosine waves measured as a function of time and decaying toward zero intensity at an exponential rate (free induction decay). Under ideal conditions we can write the signal as:

$$s(t) = \sum_j^J A_j e^{i(w_j t + \phi_j) - t/T^*_{2_j}}$$

with A_j as the amplitude, w_j as the frequency, ϕ_j as the phase and $T^*_{2_j}$ as the effective decay time of all spectral components $j \in J$. Assuming the exponential decay of $s(t)$ we obtain the signal as a sum of Lorentzian lines after application of an FFT.

$$S(w) = \sum_j^J e^{i\phi_j}(a_j(w) + d_j(w))$$

with $a_j(w) = \frac{A_j T^*_{2_j}}{1+(w-w_j)^2 T^{*2}_{2_j}}$ as the so called absorption signals and $d_j(w) = \frac{-A_j T^{*2}_{2_j}(w-w_j)}{1+(w-w_j)^2 T^{*2}_{2_j}}$ as the dispersive signal. In case of perfect phasing $\phi_j = 0, \forall j \in J$, $S(w)$ becomes:

$$S(w) = \sum_j^{|J|} \frac{A_j T^*_{2_j}}{1 + (w - w_j)^2 T^{*2}_{2_j}}$$

For real, biological spectra these assumptions are not fulfilled in general. Multiple preprocessing steps are needed to get interpretable data from the Fourier transformed NMR spectra as depicted in Figure 1.

Due to technical reasons for each NMR spectrometer a short delay between the end and the start of the measurement occurs. This implies that the sine-waves being out of phase, called the first order phase error (see Fig. 1, plot 1). Due to further imperfections a second error called zero order phase error occurs. Therefore a phase correction is desired, which can be done using the approach given in [4]. As for most spectral data a baseline correction is necessary to remove broad, baseline distorting components from the narrow metabolite NMR signals. The baseline correction is done using the a cubic interpolation approach as shown in Figure 2 (simplified).

As another (optional) step the spectra can be deconvolved see e.g [5], In the deconvolution one tries

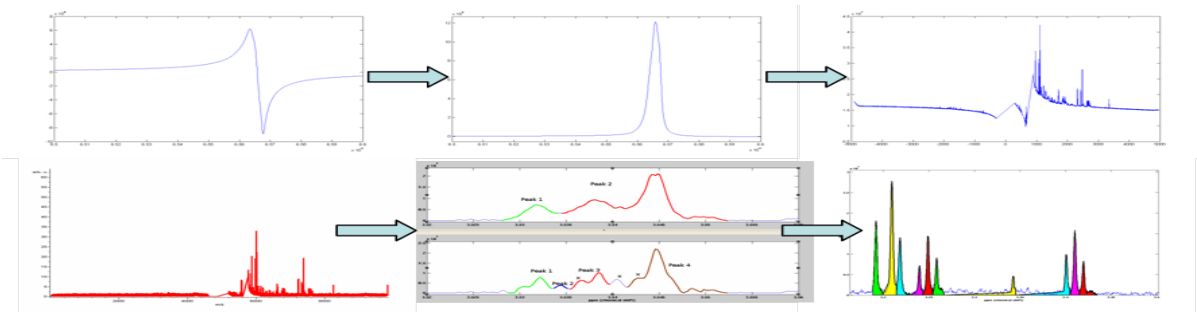


Figure 1. Workflow of the preprocessing steps applied in our NMR system. From top-left to bottom-right: We begin with a fourier transformed signal (out of phase), this error is corrected by phase correction as shown in plot 2, subsequently the water peak (with known position) is removed (interpolated by a cubic spline) and a baseline correction is applied plot $3-4$, as an optional step a deconvolution can be tried - plot 5 followed by a peak picking algorithm plot 6.

```
function [vBL] = base_det(vSignal,dDSSPoints)
% Signal size, Window width
ns = size(vSignal,1); w  = dDSSPoints;
% Empty windows whole signal
temp          = zeros(w,ceil(ns/w))+NaN;
% Fill in - over windows, min's per window
temp(1:ns)    = vSignal; [m,h] = min(temp);
g             = h>1 & h<w; % mins, not at borders
% calc minima positions with resp. to x-axis
h             = w*[0:numel(h)-1]+h;
% get valid minima and intensities
m             = m(g);  h = h(g);
% interpolate
vBaseline     = interp1(h,m,1:ns,'pchip');
```

Figure 2. Matlab code for baseline correction by piecewise cubic interpolation using a problem adequat segment width.

to remove disturbances by an inverse filtering process. Thereby we take the DSS signal - detected by the peak picking mentioned later on - as a reference $s_{ref}(t)$. The reference is considered to be a known signal, disturbed by some transformation (not considering noise effects). The signal $s(t)$ is deconvolved using the reference. An ideal reconstruction $s_{ideal}(t)$ is convolved (*) with the modified $s(t)$, subsequently. Under ideal conditions this leads to an improvement of signal resolution as shown in [5]. This procedure can be summarized very briefly as:

$$s_{comp}(t) = \frac{s(t) \cdot s_{ideal}(t)}{s_{ref}(t)}$$

the signal $s_{comp}(t)$ is the ideal spectrum of interest $s(t)$ i.e. without disturbances. Here we use the convolution theorem $FFT(f * g) = FFT(f) \cdot FFT(g)$ and add appropriate zero-padding procedures in the deconvolution and the convolution step. The deconvolution can be helpful to identify signals which are not completely resolved in the original measurement. The assumption of $s(t)$ being free of noise, is a critical point, which results in an increase of the noise level for real signals in general, therefore the deconvolution has not been used in this experiments. However it may be desirable if a large number of measurements of the same experiment are available to compensate artificial - noise related - peaks.

In standard NMR the further steps of metabolite identification and quantification are done (in general) manually by fitting a known metabolite spectrum against the signal, subtracting this pattern from the signal and repeating the prior steps until the given signal can be reliably reconstructed from the fitted patterns. This approach is very time consuming and subjective. Alternatively the data are binned, leading to a data reduction, the areas in the bins are calculated and these features of the bins are fed into a Principal Component Analysis or another analysis method. Binning in general leads to a very strong data reduction, is difficult to parametrize adequately (e.g. width of the bins) and removes a lot of the resolution of the measurement system. To overcome this a curve fitting approach (called targeted profiling) was proposed recently in [6]. Thereby a set of Lorentzians with known positions and intensity proportions are fitted against the signal and a subsequent analysis is carried out on the coefficients of the superpositions of these Lorentzians only. This greatly improves the former approach of binning leading to a compact, reliable encoding of the NMR spectra. A critical point of this approach is the complex fitting procedure the complexity of which is linearly increasing with the number of tested patterns. Further the assumption of a Lorentzian peak model for a real NMR peak is subject of discussion in the NMR community. Typical - *real measurements* - show a non Lorentzian peak shape and a method which would be able to deal with the real shape would be more desirable and accurate with respect to experimental practice. Taking this into account we focus on a peak picking approach which explicitly looks for peaks having a shape similar to the DSS signal and combine this approach with the targeted profiling suggested in [6]. The approach is sketched in Figure 3 and can be summarized as follows: On the prepared signal, as mentioned before, we apply a hill climbing search [7] for potential local maxima. Thereby only those maxima are kept which have a comparable height with respect to the DSS signal. Further constraints such as minimal/maximal peak width/height can be added to reduce the number of false hits. A line spectrum is generated at the identified peak positions and intensities which is convolved with the reference signal (here the DSS peak). This signal is subtracted from the original signal and the process is iterated until no further peaks can be detected. The obtained final peak list consists of multiple, potentially overlapping, peaks which can be considered as an information preserving reduced representation of the original NMR signal. These peak lists are subsequently compared with respect to known simulated or real metabolite spectra as depicted in Figure 4. We calculate a list of peak center positions for the measurement, considering the middle of the half peak height for each peak. This list is compared to the detected peaks in the patterns. Here a tolerance of $0.005ppm$ is chosen. In each case a matching peak must be in a (size limited) range of the start/end positions of the metabolite peak pattern. A high match of the number of peaks from the test pattern (metabolite) with respect to the peak list in the measured

```
function [peakBin, peakMag] = hillClimbing(x, negMagThresh, posMagThresh)
% container with peak positions and intensities
peakBin = []; peakMag = [];
xLen    = length(x);    % signal length
minPeakMag  = min(x);       % minimal intensity in the signal
tempPeakMag = minPeakMag;   tempPosMagThreshOffset = 0.0;

foundPeak = 0;              % indicator for a starting peak
peakCount = 1;              % peak counter
i         = 1;              % current index in the signal
slope     = x(i+1)-x(i);    % slope of the currently investigated region
while i < xLen-1            % while terminate
% scan positive slope (slope,position,indicator, temporary height)
[slope,i,foundPeak,tempPeakMag] = positive_slope_start
% temporarily store peak candidate
if x(i) > tempPeakMag % new potential peak maximum?
        tempPeakBin = i;    % position
  tempPeakMag = x(i); % local maximum to compare with
end
% scan negative slope
[slope,i,bAddPeak] = negative_slope_start
if(bAddPeak) % negative sloop search successful
        peakBin(peakCount) = tempPeakBin; % store position
  peakMag(peakCount) = tempPeakMag; % store magnitude
  peakCount                 = peakCount+1;
  foundPeak = 1;
  tempPosMagThreshOffset = x(i);
end
end
return
```

Figure 3. Pseudo code (simplified) for peak picking - hill climbing part - using problem adequat negative (0.0) and positive (90% of minimal peak height) magnitude thresholds (in acc. to DSS).

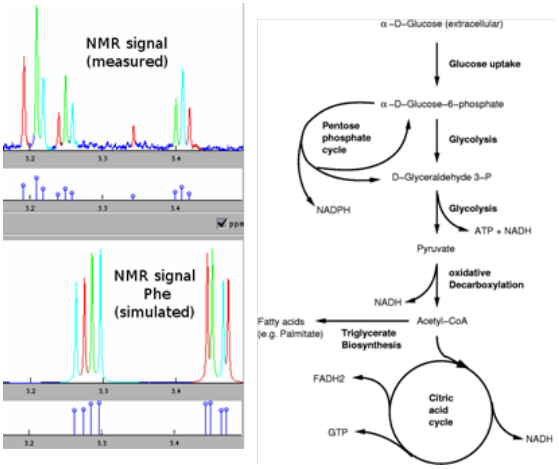


Figure 4. Identification of metabolites is based on comparison of the peak list (colored and as stems) obtained from the measured signal with respect to the peak list of a simulated metabolite (here Phenylalanine [Phe])(left). In general multiple metabolites can be detected in a spectrum. The area of the matching peaks can be used to calculate an estimate of the concentration of the metabolite. Results are serialized to an XML model which is further used to model a metabolic network (right).

spectrum is an indicator that the pattern may be present in the signal. However further checks are needed to support this hypothesis, e.g. the intensity proportions between associated peaks (e.g. in a quartet) must be checked. For all identified patterns a quantification can be tried. Thereby the area under the matching peaks is calculated and associated to the area of the DSS signal, further a scaling by the number of protons of the DSS (9) with respect to the number of protons in the metabolite e.g. Ala (4) is done to obtain a concentration (c) in m mol:

$$c(Ala) = \frac{area(Ala - {}^1H) \cdot c(DSS) \cdot 9}{area(DSS - {}^1H) \cdot 4}$$

The identified and quantified metabolites are stored with further meta informations (e.g. preprocessing parameters) in an XML file, which can be considered as a metabolite model for the analysed data set. This metabolite model will be further subject of a pathway analysis to determine models for the chemical pathways of the cell estimated by the observed metabolite concentrations with respect to the growing medium conditions.

4. EXPERIMENTS

To test the methodology, we start with simulated spectra of the considered metabolites. Thereby for each metabolite a simulation was generated with an intensity value (I) of $I = 30$ for the spin system of the metabolite, with $I = 1$ for the DSS signal and $I = 100$ for the water signal. These data form the theoretical model for the metabolite spectra database in our experiments. The application of this model against itself (the simulations which build this model) results in a perfect recognition (100% peak match) and good intensity quantifications. In a next step a mixture of all considered metabolites was simulated at different concentration levels. Again a recognition of 100% was found. The results are good for the obtained quantifications but also some under/overestimations can be observed. A closer inspection reveals these effects caused by some overlapping of peaks. For example this effect can be observed for Myo-Inositol and Glycine which do not exactly share a common peak, but the glycine peak is very close to one of the triplets of Myo-Inositol. The same argumentation applies for Lactate which is close with its quartet to another Myo-Inositol-Triplet.

After this initial experiments of pure synthetic data we take real measurements of the considered metabolite spectra. The results are depicted in Table 1. One observes that the metabolites could be identified in general, with the exception of Lac and Mal which where given in very low concentrations. To limit the effect of noise in the identifications only peaks with a minimal height of 1% of the DSS size are allowed, relaxing this criterion slightly solves this problem. However to avoid a large number of false hits and to keep detected signal intensities sufficiently above the noise level we will keep the minimal peak height of 1% of the DSS size for subsequently analyses.

In a next step data from the glucose experiment have been analysed, which are real biological data generated by metabolic process of FDCPmix cells on three types of growing media. Some results are depicted in Table 2. These results are very preliminary due to multiple reasons. For example the number of spectra for each condition is very small. Under this light the experiments should only be considered as an illustrative real life example how to use the presented system for such types of experiments. The results are also shown in Figure 5. As already mentioned the data support is very limited so its hard to give any interpretation of the results, but considering the different graphs one may get the impression that for Alanine the third condition will cause a decreased expression. For succinate one may conclude that the conditions do not have any effect and for lactate a small increase of the concentration can be seen. These results have been checked by manual inspection and appear to be correct. However, as already mentioned, there are only few spectra supporting the data such that new measurements are necessary to prove this initial hypotheses. We also compared our findings with respect to an alternative method using the data analysis package Chenomx 5.0 [6]. Thereby we found a perfect agreement of the mean concentrations as depicted in Figure 5. However for some of the metabolites the esti-

Metabo (S)	Metabo (I)	PM	EC	QC
Ser	Ser	100%	0.50	0.73
Myo	Myo	93%	0.34	0.45
Lac	(Lac)	(100%)	0.07	(0.2)
Gly	Gly	100%	0.19	0.43
Suc	Suc	100%	0.25	0.41
Cit	Cit	100%	0.20	0.34
Mal	(Mal)	(100%)	0.27	(0.39)
Ala	Ala	71%	0.38	0.52

Table 1. Analysis of real measured pure metabolite data (S) with respect to a synthetic metabolite database (identification - I). At standard conditions (minimal peak height 1% of DSS almost all metabolites could be identified successfully. The experimental protocols note for *Lac* bad solving conditions. A closer (manual) inspection of the spectrum reveals, that the doublet of Lac has been detected but the quartet is very small and could not be detected. Without the quartet a concentration of 0.19 is quantified. A similar situation occurred for malate, lowering the minimal peak height of the peak picking algorithm to 0.5% of the DSS signal both metabolites can be detected (results in brackets). The percentage of matched peaks is given in the column (PM), the expected concentration in (EC) and the quantified concentration in (QC).

mations of Chenomx appear to be unlikely due to artificial fittings, not sufficiently supported by the analysed data. These effects have not been found using a peak based approach, because fits are only tried for identified peaks.

Condition	Metabolite	PM	QC (mean/std)
1	Ala	$71-85\%$	$0.8/0.47$
1	Gly	100%	$1.25/0.49$
1	Lac	71%	$0.2/0.28$
1	Suc	100%	$0.07/0.1$
2	Ala	$71-100\%$	$0.89/0.7$
2	Gly	100%	$1.8/0.65$
2	Lac	85%	$0.6/0.48$
2	Suc	100%	$0.17/0.09$
3	Ala	$71-85\%$	$0.54/0.37$
3	Gly	100%	$0.74/0.51$
3	Lac	$71-100\%$	$0.73/0.08$
3	Suc	100%	$0.09/0.1$

Table 2. Analysis of real measured extracts of growing media with FDCPmix cells. Only those metabolites are shown which are frequent within the specific conditions (Ala,Gly,Lac,Suc). Concentrations are given as mean concentration values over multiple spectra for a metabolite in a condition. Condition 1 accounts for a glucose level of $1mM$ (5 spectra), condition 2 accounts for glucose of $5mM$ (6 spectra) and the last condition for a glucose level of $25mM$ (4 spectra). If a metabolite has not been detected its concentration is assumed as 0.0 in the calculations (this is in general correct - verified by manual inspection). PM and QC like in Table 1

5. DISCUSSION AND CONCLUSIONS

We presented a system for the automatic identification and quantification of metabolites from ^{1}H-NMR-measurements. The approach is based on peak lists generated by a hill climbing peak picker combined with a measurement specific shape subtraction. This allows a sensitive and measurement specific detection of peaks, which is in general more appropriate then a modeling with a Lorentzian (only) peak assumption. Another advantage of our method is the automatic reliability estimation of the identifications such that false positives are reduced, because it is only using the identified peaks and does not fit arbitrary metabolites against the signal.

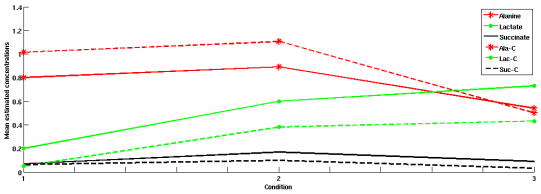


Figure 5. Mean estimated concentration values calculated for the three conditions using the prominent metabolites Ala (*), Lac (o) and Suc. The results are compared with an analysis done using Chenomx 5.0 (dashed lines - scaled)

We have shown that the method can be successfully applied on simulated spectra, real pure metabolite spectra and real experimental NMR spectra obtained from growing medium experiments. Beside of these positive aspects there are also some remaining challenges. First, the shape modeling, which is currently based on the DSS signal could be made more general by use of e.g. a wavelet based fitting procedure applied on multiple peaks, this would reduce the effect of noise or artifacts which may be present on the DSS reference and interfere the subsequently peak detection. Further the quantified concentrations are strongly affected by overlapping peaks. A rule based, knowledge driven, correction of plain area calculations may be desirable. Constraints on the accounted peak areas with respect to concurrent metabolites with potential fuzzy peak sets could be an interesting option to get more reliable estimates. In a next step the initial results must be verified by a larger amount of measurements combined with a modeling of the chemical reactions which also may be helpful to improve the metabolite model[1].

6. REFERENCES

[1] S. Smith, T. Levante, B. Meier, and R. Ernst, "Computer simulations in magnetic resonance. an object oriented programming approach," *J. Magn. Reson.*, vol. 106a, pp. 75–105, 1994.

[2] V. Govindaraju, K. Young, and A. A. Maudsley, "Proton NMR chemical shifts and coupling constants for brain metabolism," *NMR in Biomedicine*, vol. 13, pp. 129–153, 2000.

[3] O. Kan, A. D. Whetton, and C. Heyworth, "Development of haemopoietic cells in liquid culture," in *Haemopoiesis: a Practical Approach*, N. Testa and G. Molineaux, Eds., pp. 123–137. IRL Press, Oxford, 1993.

[4] L. Chen, Z. W. an Laiyoong Goh, and M. Garland, "An efficient algorithm for automatic phase correction of nmr spectra based on entropy minimization," *Journal of Magnetic Resonance*, vol. 158, no. 1-2, pp. 164–168, 2002.

[5] K. R. Metz, M. M. Lam, and A. G. Webb, "Reference deconvolution: A Simple and Effective Method for Resolution Enhancement in Nuclear Magnetic Resonance Spectroscopy," *NMR in Biomedicine*, vol. 13, pp. 129–153, 2000.

[6] A. M. Weljie, J. Newton, P. Mercier, E. Carlson, and C. M. Slupsky, "Targeted profiling: Quantitative analysis of ^1H nmr metabolomics data," *Anal. Chem.*, vol. in press, pp. 4430–4442, 2006.

[7] T. H. Park, *Towards Automatic Musical Instrument Timbre Recognition*, Ph.D. thesis, Princeton University, 2004.

[1] ACKNOWLEDGMENT: We are grateful to C. Wierling at MPI f. Molecular Genetics and the whole MetaSTEM team. This work was supported by the Federal Ministry of Education and Research under PNR:934000-545, in the Project NMR Metabolic Profiling of the Stem Cell Niche (MetaSTEM).

A STOCHASTIC FRAMEWORK FOR THE QUANTIFICATION OF SYNCHRONOUS OSCILLATION IN NEURONAL NETWORKS

Gaby Schneider[1] *and Danko Nikolić*[2,3]

[1]Dept. of Computer Science & Mathematics, University Frankfurt,
Robert-Mayer-Str. 10, 60325 Frankfurt/Main, Germany
[2]Frankfurt Institute for Advanced Studies, University Frankfurt/Main, Germany,
[3]Max-Planck-Institute for Brain Research, Frankfurt/Main, Germany,
schneider@math.uni-frankfurt.de, danko@mpih-frankfurt.mpg.de

ABSTRACT

Synchronous oscillations are believed to be important for neuronal information processing. We use a stochastic model for parallel point processes to estimate the strength of synchrony in an oscillating network of neurons recorded in cat visual cortex. The model has the surprising ability to predict interactions between the neurons solely on the basis of the individual processes, i.e., the autocorrelograms. The strength of synchronization is defined as the mismatch between the predicted and the observed strength of interaction. This method has the advantage of distinguishing changes in the strength of synchrony from changes in the properties of the underlying processes. Thus, the model provides new approaches for the investigation of dynamical changes in the joint oscillatory activity of neuronal networks.

1. INTRODUCTION

The synchronization of oscillatory neuronal responses is likely to play an important role in cortical processing and is commonly investigated using pairwise cross-correlation histograms (CCHs; [1], Figure 1).

In a CCH, one uses either the height, the width or the area of the central peak to investigate the amount of synchronous firing. Such measures are then evaluated statistically by comparing to independent processes [2, 3, 4]. However, this null hypothesis of independent processes is insufficient to describe the nature of processes with a common oscillatory rhythm. Therefore, such methods can only indicate statistically significant deviations from independent processes and can thus be only related indirectly to the properties of the underlying processes.

In the present work, we use a stochastic spike-train model [5] that describes the oscillatory properties of the underlying processes and makes simple assumptions about their interactions. Therefore, the model offers a framework for relating the properties of individual processes, visible in the auto-correlation histograms (ACHs; [6]), to the properties of interactions between the processes, visible in CCHs. This allows for a direct measure of synchrony, which we will define here as the percentage of spike pairs that take part in the same rhythm.

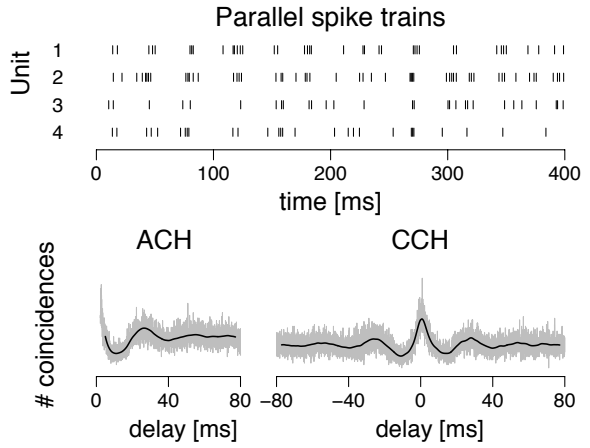


Figure 1. Parallel processes with a common oscillatory rhythm, reflected in ACHs of individual processes and in CCHs computed for pairs of processes (gray: raw counts, black: counts smoothed with Gaussian kernel; sd $= 1$ ms).

2. THE SPIKE-TRAIN MODEL

2.1. Model assumptions

We use a spike-train model for parallel point processes called the ELO model (Exponential LOcking to a free oscillator), which is described in detail elsewhere [5]. The model assumes a global oscillatory rhythm (called the packet onset process, POP), shared across all processes and described by a stationary random walk $(B_n)_{n\in\mathbb{Z}}$ with independent and normally distributed increments $B_{i+1}-B_i$ with mean μ and variance σ^2 (Figure 2, top line). An event in the POP marks the time points at which the firing intensities rise for all processes simultaneously (cycle onset). In each process j, an onset B_i gives rise to an independent Poissonian spike packet with an expected number of spikes α_j and exponentially decreasing firing intensity with time constant τ_j. With B_{n_t} denoting the last onset before t, the firing intensity of process j at time t is described by

$$\frac{\alpha_j}{\tau_j}\sum_{i=-\infty}^{n_t} e^{\frac{-(t-B_i)}{\tau_j}} + \beta_j. \qquad (1)$$

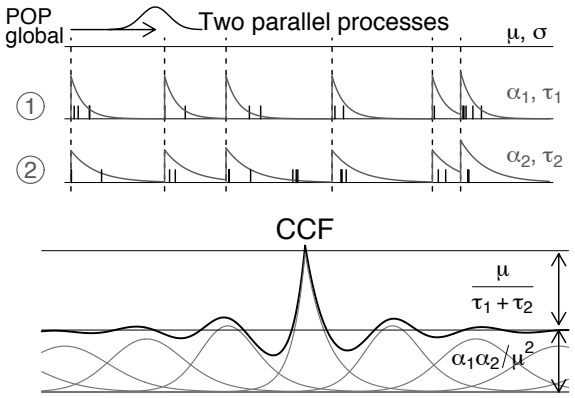


Figure 2. The ELO model for two parallel processes: The global POP with $\mathcal{N}(\mu, \sigma^2)$-distributed increments gives rise to simultaneous spike packets with exponentially decreasing firing intensities with parameters τ_1 and τ_2, respectively. The corresponding theoretical CCF shows a central peak and an oscillatory shape.

The smaller τ_j, the more densely the spikes cluster at the packet onsets (Figure 2, $\tau_1 < \tau_2$). Since the POP is shared by all processes, μ and σ are global parameters. In contrast, α and τ may differ across units.

2.2. Cross-correlation function

Within this framework, the auto- and cross-correlation functions (ACF, CCF) of processes that comply with the model assumptions can be derived by decomposition of the processes into different packets (Figure 2, bottom panel). The CCF $F_{ab}(s)$ at shift $s \geq 0$ between processes a and b is then given by (for a proof see [5])

$$F_{ab}(s) = \frac{\alpha_a \alpha_b}{\mu(\tau_a + \tau_b)} \Bigg\{ e^{-\frac{s}{\tau_a}} + \sum_{j \in \mathbb{Z} \setminus \{0\}} e^{\frac{s-\mu_j}{\tau_a} + \frac{\sigma_j^2}{2\tau_a^2}} \Phi\left(\frac{\mu_j - s - \sigma_j^2/\tau_a}{\sigma_j}\right) + \sum_{j \in \mathbb{Z} \setminus \{0\}} e^{\frac{\mu_j - s}{\tau_b} + \frac{\sigma_j^2}{2\tau_b^2}} \Phi\left(\frac{s - \mu_j - \sigma_j^2/\tau_b}{\sigma_j}\right) \Bigg\}, \quad (2)$$

where Φ denotes the standard normal distribution function. $F_{ab}(s) = F_{ba}(-s)$, and the ACF of a equals F_{aa}.

2.3. Relation between ACF and CCF

It follows that the CCH can be predicted directly from the properties of the individual ACHs because it depends on the same parameters. The smaller τ_a and τ_b, the higher the respective ACH peaks, and the higher also the corresponding CCH peak.

This relation can be quantified with the first term of the CCF, $\alpha_a \alpha_b / (\mu(\tau_a + \tau_b)) e^{-\frac{s}{\tau_a}}$, which describes the intensity of spike pairs that belong to simultaneous packets and thus, determines the shape of the central peak. Since the level of the asymptotic baseline in a CCF is given by the product of the firing intensities, $\frac{\alpha_a \alpha_b}{\mu^2}$ (Figure 2), the fraction f_h of the total peak height cut at baseline level is given by

$$f_h^{ab} = \frac{\text{baseline}}{\text{peak height}} = \frac{\tau_a + \tau_b}{\mu}, \quad (3)$$

and by $f_h^{aa} = 2\tau_a/\mu$ in ACF$_a$. Thus, the simple relation

$$f_h^{ab} = 1/2 \cdot (f_h^{aa} + f_h^{bb}) \quad (4)$$

shows the direct relation between the height of the CCH peak and the respective ACHs peaks. We will use this relation to estimate the degree to which two processes are locked to the same oscillatory rhythm.

3. FITTING THE MODEL TO A DATA SET

3.1. Parameter estimation

We fitted the ELO model to a sample data set consisting of neuronal firing activity of 14 multi-units recorded in parallel in cat primary visual cortex under visual stimulation (stimuli are shown in Figure 7, see [7] for experimental methods). We first estimated the times of the global cycle onsets by smoothing the firing activity of all units with a Gaussian kernel. Packet onsets were identified as the points at which 60% of the maximum was reached (gray dots in the upper panel of Figure 3). This analysis suggested that independence and normal distribution of intervals between spike packets were appropriate assumptions for the POP. We then estimated the parameters by fitting the theoretical ACFs (Equation (2)) to the observed ACHs using a nonlinear least squares algorithm. As mentioned, μ and σ were chosen to be identical in all units. The fitted ACFs corresponded well to the empirical ACHs (Figure 3, bottom panel). For stimulation condition 1, the parameter estimates were $\hat{\mu} = 25.3$ ms, $\hat{\sigma} = 7.3$ ms. The values of $\hat{\tau}_1, \ldots, \hat{\tau}_{14}$ were in the range of $3.5 - 8$ ms. Approximations for variances of the parameter estimates were derived both numerically by the least squares algorithm and by splitting the data into smaller groups. Both methods yielded comparable results, with standard errors smaller than 0.1 ms for μ and σ and $0.1 - 0.9$ ms for $\tau_1, \ldots, \tau_{14}$.

3.2. Prediction of interactions

With the parameters derived from the ACHs, we predicted the shape of each CCH by using Equation (2). In many cases, this prediction corresponded well to the empirically obtained CCH (Figure 4).

In some cases, the units showed nonstationary rate responses within trials that were different in both units (Figure 5, left panel). As a consequence, the observed CCHs were lower than those predicted from the ACHs (medium panel). Therefore, nonstationarity was taken into account by using a correction factor proposed in [5]: We described the firing rate of a unit as a step function, which we estimated from the overall firing rate across all trials, measured in windows of 200 ms (bold curves in the left panel of Figure 5). With the given rate estimates $\lambda_{1,a}, \ldots, \lambda_{k,a}$

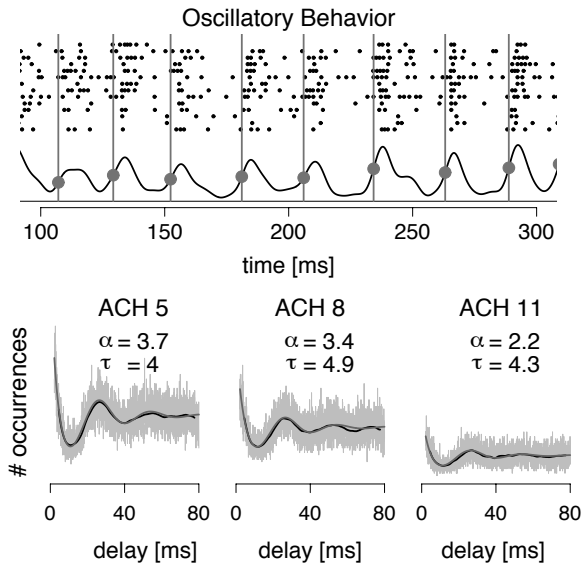


Figure 3. Investigation of model assumptions and parameter estimation. Upper panel: The spikes recorded in the 14 units show a joint oscillatory rhythm that can be described by independent and normally distributed intervals. Bottom: The observed ACHs (colors as in Figure 1) correspond well to the fitted ACFs (medium gray).

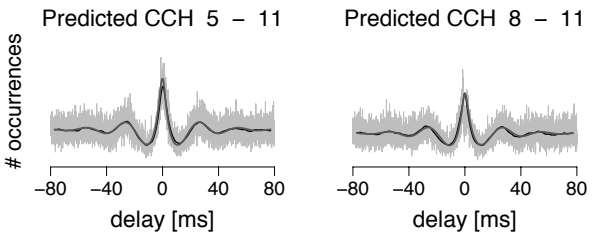


Figure 4. Observed CCHs (colors as in Figure 3) and theoretical CCFs (medium gray) predicted from the parameters derived from the corresponding ACHs.

and $\lambda_{1,b}, \ldots, \lambda_{k,b}$, the raw CCF prediction uses the product of the firing rates estimated from the ACHs,

$$\hat{\alpha}_a \hat{\alpha}_b = \sqrt{\sum_i \lambda_{i,a}^2} \sqrt{\sum_j \lambda_{j,b}^2}. \quad (5)$$

However, the correct prediction would be

$$r = \sum_i \lambda_{i,a} \lambda_{i,b}. \quad (6)$$

We therefore corrected each predicted CCF with the term

$$c_{ab} = r / \hat{\alpha}_a \hat{\alpha}_b. \quad (7)$$

Most correction factors ranged between 0.9 and 1 and resulted in good agreement between the predicted and the empirical CCHs (Figure 5, right panel).

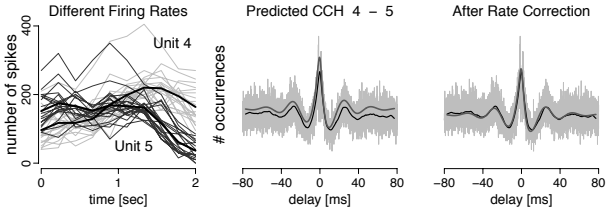


Figure 5. Nonstationary rate responses along a trial need to be corrected when predicting a CCH. Left panel shows firing rates of units 4 (light gray) and 5 (dark gray) recorded in 20 trials in stimulation condition 1. Direct prediction leads to an erroneous height of the CCH (medium panel), which can be corrected with Equation (7).

4. THE DEGREE OF UTILIZED SYNCHRONY

A good agreement of the CCF predictions with some of the empirically obtained CCHs suggests that the model assumption about all units sharing the same oscillatory rhythm describes the data well. Therefore, this prediction can be used as a reference: The CCF predicted from the ACHs indicates the *maximal possible strength of synchrony* that can be obtained for the given pair of units. This predicted maximum depends on the ACHs in the following way: If ACHs have small peaks, the predicted CCH will also have a small peak and vice versa.

This perspective allows one to define the strength of *utilized synchrony*, which is the degree to which the observed CCH peak corresponds to the peak predicted under the above assumption that the units share the same rhythm (100% locking). Indeed, a number of CCHs showed lower peaks than predicted from their ACHs (Figure 6, bottom right panel). This indicates that the units utilize less than 100% of their potential to synchronize, indicating in turn that oscillation shared across units is weaker than the oscillation of each unit individually.

Within the spike-train model, utilized synchrony can be estimated as follows (Figure 6, upper panel): We assume that the units share the same POP only sometimes (A), while on other occasions they are locked to independent POPs with the same parameters (B). The resulting CCH is a linear combination of the CCF predicted from the ACHs (black curve in the second panel) and a flat correlogram resulting from independent processes:

$$CCH = \vartheta \cdot CCF_{predicted} + (1 - \vartheta) \cdot baseline. \quad (8)$$

The parameter ϑ indicates the percentage of spike pairs that share the same oscillatory rhythm. This number can be estimated with a least squares approach when comparing the predicted CCF to the observed CCH. When applying this measure to stimulation condition 1, the estimates of ϑ ranged between $0.4 - 0.9$, with standard errors of about 0.03. Analogous results were obtained for the other stimulation conditions.

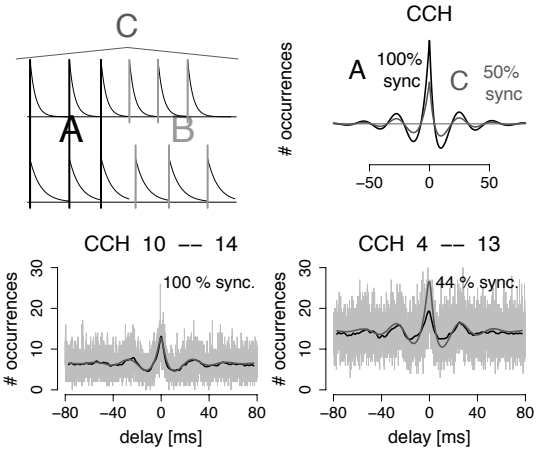


Figure 6. If a predicted CCF is higher than the observed CCH, this could indicate that some spikes share the same rhythm (A), but others engage in independent rhythms (B). Such a combination (C) reduces the CCH amplitude and allows estimating the fraction of spikes in the same rhythm (utilized synchrony, bottom, colors as Figure 3).

4.1. Changes in utilized synchrony across stimuli

This method allows one to investigate whether the degree of utilized synchrony changes across stimulation conditions. Even if ACHs do not change and thus the potential to synchronize is constant, it is possible that units change utilized synchrony. This directly affects the CCH peak and can thus account for classical results based on measures of the peak height [4]. For example, we found that utilized synchrony was increased for a stimulus with one moving bar, as compared to two conflicting bars (Figure 7), which is consistent with previous reports, e.g. [8].

However, it is also possible that utilized synchrony provides a different kind of information than the classical measures. An indication of such information is shown in Figure 7 where utilized synchrony could distinguish between two groups of units, Group A (orientation preference 30°/ 210°) showing much higher utilization of the potential to synchronize for stimulus 5 than Group B (orientation preference 150°/330°), while Group B synchronized more strongly in stimulus 6. The functional significance of these results is yet to be investigated. However, it indicates that utilized synchrony might provide important information about the dynamics of neuronal oscillation.

5. DISCUSSION

We use a stochastic model that describes parallel processes with a joint oscillation and that can predict a CCH directly from the ACHs. By comparing the observed and the predicted CCHs, we propose to estimate to which degree units utilize their potential to synchronize. This allows one also to distinguish whether changes in a CCH are due to changes in the individual processes or to changes in utilized synchrony. The method may therefore provide new information on the dynamics of neuronal synchronization.

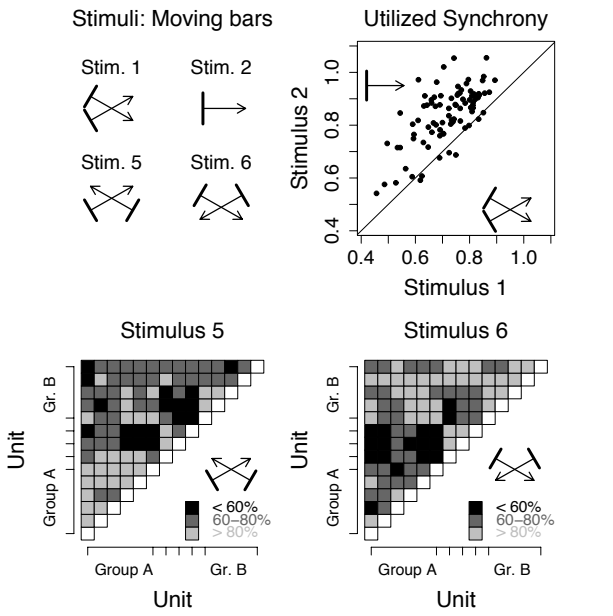


Figure 7. Changes in utilized synchrony across stimuli. Utilized synchrony increases from stimulus 1 to 2 (upper right). Bottom: Group A shows high utilized synchrony in stim. 5, group B shows high synchrony in stim. 6.

6. REFERENCES

[1] D. H. Perkel, G. L. Gerstein, and G. P. Moore, "Neuronal spike trains and stochastic point processes II. Simultaneous spike trains," *Biophys. J.*, vol. 7, no. 4, pp. 419–440, 1967.

[2] M. Abeles, "Quantification, smoothing, and confidence limits for single-units' histograms," *J. Neurosci. Methods*, vol. 5, pp. 317–325, 1982.

[3] A. M. H. J. Aertsen and G. L. Gerstein, "Evaluation of neuronal connectivity: Sensitivity of cross-correlation," *Brain Res.*, vol. 340, pp. 341–354, 1985.

[4] P. König, "A method for the quantification of synchrony and oscillatory properties of neuronal activity," *J. Neurosci. Methods*, vol. 54, pp. 31–37, 1994.

[5] G. Schneider, "Messages of oscillatory correlograms - a spike-train model," *Neural Comp.*, vol. 20, no. 5, 2008.

[6] D. H. Perkel, G. L. Gerstein, and G. P. Moore, "Neuronal spike trains and stochastic point processes I. The single spike train," *Biophys. J.*, vol. 7, no. 4, pp. 391–418, 1967.

[7] G. Schneider and D. Nikolić, "Detection and assessment of near-zero delays in neuronal spiking activity," *J. Neurosci. Methods*, vol. 152, pp. 97–106, 2006.

[8] A. K. Engel, P. König, and W. Singer, "Direct physiological evidence for scene segmentation by temporal coding," *Proc. Natl. Acad. Sci. USA*, vol. 88, no. 20, pp. 9136–9140, 1991.

ABOUT BOOLEAN NETWORKS WITH NOISY INPUTS

Steffen Schober

Ulm University
Institute of Telecommunications and Applied Information Theory
Albert-Einstein-Allee 43, 89081 Ulm, Germany

ABSTRACT

Consider a feed-forward Boolean network with i input and o output nodes. This system can be decomposed into a collection of o Boolean functions with i arguments. Assume that each argument is assigned a 1 with probability $0 \leq p \leq 1$ and 0 otherwise. Further assume that noise is applied to the input by independently flipping the value of each argument with an average probability ϵ. It is shown, that the probability that a function output is affected is less or equal $\epsilon \cdot \mathrm{as}(f)$, where $\mathrm{as}(f)$ denotes the average sensitivity of the function.

1. INTRODUCTION

In the late 1960s Stuart Kauffman [1] was among the first researchers that came up with the idea to model genetic regulatory networks by interacting binary memory elements, whereas the interactions are described by Boolean functions.

From the networks dynamics point of view, Kauffman found that there exist two broad regimes under which such networks occur. In the ordered phase single transient errors tend to vanish, therefore they act only on a *local* level. In the disordered phase such errors are likely to affect large part of the network. It has been argued, that for living organisms it is preferable to exist at the border of the two regimes, which seems to represent a trade-off between *robustness* and *adaptivity*.

Kauffman also came up with the idea of studying random networks to learn more about the general conditions under which *order* can exist in Boolean networks [1]. Nowadays it is well known that, at least for the simple type of networks studied by Kauffman, the so called NK-networks, the *sensitivity* of the functions, more precisely the expectation of it, plays a key role [2]. Roughly speaking, the sensitivity is a measure for the probability that a perturbation at the input will affect the output of the function (see Equation (1) in Section 2 for the precise definition). From this point of view it is easy to see that only networks with comparably low sensitivity of their function will be dynamically stable.

An interesting question arises in this context. Consider a feed-forward Boolean network with i input and o output nodes. This might be either the whole system or a part of it, in the simplest case a single Boolean function (if $o = 1$). If this system is robust against a single error at its inputs, will it also be robust against multiple random errors, i.e., noise, occurring with low probability at its inputs? To be more precise, if we randomly flip the value at each input with probability ϵ, what is the probability that the system still computes the correct value (as if ϵ would be 0)? Quite obviously we can view this network as a collection of o Boolean functions with i inputs, for simplicity we assume that all of them have the same delay, that is the time the functions need to compute their value. Note that we make no assumptions about the implementation of this functions except that there are no feedback loops. Hence it is to sufficient to study single Boolean functions. We will show the following: Given a Boolean function with n arguments. To each argument we independently assign a 1 with probability $0 \leq p \leq 1$ and 0 otherwise. Further assume that noise is applied by independently flipping the value of each input with a average probability ϵ, the precise procedure will be described in the next section. Then the probability that the output of the function is affected is at most $\epsilon \cdot \mathrm{as}(f)$, where $\mathrm{as}(f)$ denotes the average sensitivity of the function. We will formalize and prove the result in the following two sections.

Let us now return to our initial question. The result shows that a feed-forward Boolean network is able to deal with noisy inputs, if the sensitivity of each of the o Boolean functions is at most 1. From an other point of view this means that it does not amplify noise, a quite important property if you think of them as a part of a larger system.

2. MAIN RESULT

We consider the set of all Boolean functions with n arguments, denoted by $\mathcal{B}_n \doteq \{f : \Omega^n \to \Omega\}$, where $\Omega = \{0, 1\}$. Throughout the paper we assume that the inputs of the functions are chosen at random from the set of all possible vectors of dimension n, according the probability measure μ_p which is defined as

$$\mu_p(\mathbf{x}) = \prod_{i=1}^{n} \mu_p(\mathbf{x}_i)$$

for $\mathbf{x} \in \Omega^n$, where $\forall i \in \{1, \ldots, n\}$

$$\mu_p(\mathbf{x}_i) = \begin{cases} p & \text{if } x_i = 1 \\ 1-p & \text{if } x_i = 0 \end{cases} \quad \text{for} \quad 0 < p < 1.$$

The sensitivity of a function f at input $\mathbf{x} \in \Omega^n$ is defined as

$$\mathrm{s}(f, \mathbf{x}) \doteq \#\left\{\mathbf{y} \in \Omega^n | \mathrm{d_H}(\mathbf{x}, \mathbf{y}) = 1 \text{ and } f(\mathbf{x}) \neq f(\mathbf{y})\right\}.$$

Here $\mathrm{d_H}(\cdot, \cdot)$ denotes the Hamming distance between two vectors. The average sensitivity is defined as

$$\mathrm{as}(f) \doteq \sum_{\mathbf{x} \in \Omega^n} \mu(\mathbf{x}) \mathrm{s}(f, \mathbf{x}). \tag{1}$$

Now suppose we apply noise to the input of the function. Usually it is assumed that each input is flipped with a low probability, $0 \leq \epsilon \leq 1/2$. Denoting the resulting *noisy* input $N_\epsilon(\mathbf{x})$ one is interested in

$$\tilde{\rho_f} = \Pr_{\mathbf{x} \sim \mu_p}[f(\mathbf{x}) \neq f(N_\epsilon(\mathbf{x}))]. \tag{2}$$

Note that we usually will omit the subscript. For reasons that will become obvious later, we will use a different *noise model*. Let $0 \leq \delta \leq 1$. Then the input $\mathbf{y} = N_\delta(\mathbf{x})$ is obtained as follows: With probability δ we set $\mathbf{y}_i = \mathbf{x}_i$ and with probability $1 - \delta$

$$y_i = \begin{cases} 1 & \text{with probability } p \\ 0 & \text{with probability } 1 - p \end{cases}.$$

This δ-noise is closely related to the ϵ-noise. Let

$$\delta = 1 - \frac{\epsilon}{2p(1-p)} \quad \text{and} \quad 0 \leq \epsilon \leq 2p(1-p). \tag{3}$$

Note that if $p \neq \frac{1}{2}$ we need to restrict ϵ to values less than $\frac{1}{2}$ to ensure $\delta \geq 0$. Now

$$\Pr[y_i \neq x_i | x_i = 0] = \tfrac{\epsilon}{2(1-p)}$$
$$\text{and} \qquad \Pr[y_i \neq x_i | x_i = 1] = \tfrac{\epsilon}{2p}.$$

As can be seen, for $p \neq 1/2$ the probability of flipping an input depends on its value. For example, for $p < 1/2$ an error is more likely if $x_i = 1$ than for $x_i = 0$. However we note that the *average* probability for an error is always given by

$$\Pr[y_i \neq x_i] = \epsilon.$$

Instead of $\tilde{\rho_f}$ we will study

$$\rho_f = \Pr[f(\mathbf{x}) \neq f(N_\delta(\mathbf{x}))]. \tag{4}$$

Now our result can be formulated as follows:

Lemma 1. *Given a Boolean function $f : \Omega^n \to \Omega$ with average sensitivity $\mathrm{as}(f)$, suppose the inputs are chosen at random according to μ_p, with $0 \leq p \leq 1$. Then*

$$\rho_f \leq \mathrm{as}(f) \cdot \epsilon \tag{5}$$

where ρ_f is given according (4) *and δ according* (3)*. Hence if a function has average sensitivity of at most* 1*, it will not amplify the noise at the input.*

The proof will be given in the following section. In fact, once we set up the mathematical machinery, Lemma 1 is a nearly trivial corollary. It is worth noting that if we consider ρ_f as a function of ϵ then

$$\left.\frac{\mathrm{d}\rho_f(\epsilon)}{\mathrm{d}\epsilon}\right|_{\epsilon=0} = \mathrm{as}(f), \tag{6}$$

a fact already known for $p = \frac{1}{2}$ (see [3],[4]).

3. PROOF OF MAIN RESULT

As our proof bases on the Fourier analysis of Boolean functions we will first give a brief summary of the topic in the first subsection. In this work we will stick closely to [5, 6, 7]. Note that such techniques were used much earlier, as stated by Xiao and Massey [8]. The first who used the Fourier (or Walsh) transform on Boolean functions was Golomb [9] in the 1950's. The second subsection will cover the spectral representation of the average sensitivity for non-uniform measures [7], where the third subsection will discuss the so called *noise* operator [5]. Lemma 1 will then follow as a simple corollary.

3.1. Fourier transform on the cube

In the following we will consider $\mathcal{F}_n \doteq \{f : \Omega^n \to \mathbb{R}\}$. Obviously $\mathcal{B}_n \subset \mathcal{F}_n$. As we defined a probability measure on the possible inputs, we can view a function $f \in \mathcal{F}_n$ as a random variable mapping into the real numbers. We will denote the expectation value of a function f with respect to μ_p as

$$\mathop{\mathbb{E}}_{\mathbf{x} \sim \mu_p}[f] = \sum_{\mathbf{x} \in \Omega^n} f(\mathbf{x}) \mu_p(\mathbf{x}).$$

Similarly

$$\mathop{\mathbb{E}}_{\mathbf{x} \sim \mu_p}[f \cdot g] = \sum_{\mathbf{x} \in \Omega^n} g(\mathbf{x}) f(\mathbf{x}) \mu_p(\mathbf{x}). \tag{7}$$

Note that we usually omit the subscript if it is understood from the context. Equation (7) gives rise to an *inner product* defined on $\mathcal{F}$, let $f, g \in \mathcal{F}$ then define

$$\langle f | g \rangle \doteq \mathop{\mathbb{E}}_{x \sim \mu_p}[f \cdot g].$$

We note that the inner product depends on μ_p and it is *bi linear* that is for any function $f_1, f_2, f, g \in \mathcal{F}$

$$\langle f_1 + f_2 | g \rangle = \langle f_1 | g \rangle + \langle f_2 | g \rangle$$

and

$$\langle f | g \rangle = \langle g | f \rangle.$$

If $\langle f | g \rangle = 0$ we say that the functions are orthogonal. The inner product can be used to define the *Euclidean norm*, i.e.,

$$||f||_2 \doteq \sqrt{\langle f | f \rangle} = \sqrt{\mathbb{E}[f^2]}.$$

Now we are ready to define an *orthonormal* basis for the vector space $\mathcal{F}$ as introduced by Talagrand [6].

For all $i \in \{1, \ldots, n\}$ we define

$$\Phi_i(\mathbf{x}) \doteq \begin{cases} \sqrt{\frac{p}{1-p}} & \text{if } x_i = 0 \\ -\sqrt{\frac{1-p}{p}} & \text{if } x_i = 1 \end{cases}, \tag{8}$$

and for all $\mathbf{0} \neq \mathbf{u} \in \Omega^n$

$$\Phi_{\mathbf{u}}(\mathbf{x}) = \prod_{\{i | u_i = 1\}} \Phi_i(\mathbf{x}). \tag{9}$$

For the all zero vector $\mathbf{0}$ we set

$$\Phi_{\mathbf{0}}(\mathbf{x}) \doteq 1.$$

Now the set of functions

$$\{\Phi_{\mathbf{u}}(\mathbf{x})\}_{\mathbf{u}\in\Omega^n}$$

forms an orthonormal basis of $\mathcal{F}_n$, that is

$$\langle\Phi_{\mathbf{u}}|\Phi_{\mathbf{v}}\rangle = \begin{cases}1 & \text{if } \mathbf{u}=\mathbf{v}\\ 0 & \text{if } \mathbf{u}\neq\mathbf{v}\end{cases},$$

and, obviously from above, $||\Phi_{\mathbf{u}}||_2 = 1$. Hence any function f can be expressed as a *Fourier series*, i.e., a linear combination of base vectors

$$f(\mathbf{x}) = \sum_{\mathbf{u}\in\Omega^n} \hat{f}(\mathbf{u})\Phi_{\mathbf{u}}(\mathbf{x})$$

where the coefficients $\hat{f}(\mathbf{u})$ can be found by the *Fourier transform*

$$\hat{f}(\mathbf{u}) = \langle f|\Phi_{\mathbf{u}}\rangle\,.$$

Note that Parseval's theorem also holds for the biased Fourier transform, namely for any function $f\in\mathcal{F}_n$

$$||f||_2^2 = \langle f|f\rangle = \sum_{\mathbf{u}\in\Omega^n}\hat{f}(\mathbf{u})^2.$$

Although one can apply the Fourier transform to a Boolean function f directly, we will usually consider the Fourier transform of the character function of f

$$\chi_f : \Omega^n \to \{1,-1\},\quad \chi_f(\mathbf{x}) = (-1)^{f(\mathbf{x})},$$

that is, we simply substitute $0\to 1$ and $1\to -1$. Note that

$$\hat{\chi}_f(\mathbf{u}) = \langle\chi_f|\Phi_{\mathbf{u}}\rangle$$

and

$$\sum_{\mathbf{u}\in\Omega^n}\hat{\chi}_f(\mathbf{u})^2 = 1. \tag{10}$$

3.2. Average sensitivity

It is well known that for the uniform measure $\mu(\mathbf{x})$ with $p=1/2$ the average sensitivity can be related to the Fourier spectra of the function, [10, 11]. For non-uniform product measures a similar relation holds (for example [7], but note the different constant factor due to the different range):

Lemma 2. *For any Boolean function $f\in\mathcal{B}_n$,*

$$\mathrm{as}(f) = \frac{1}{4p(1-p)}\sum_{\mathbf{u}\in\Omega^n}|\mathbf{u}|\hat{\chi}_f(\mathbf{u})^2. \tag{11}$$

Here, and in the following, $|\mathbf{u}|$ denotes the Hamming weight of $\mathbf{u}$. Note that for $p=1/2$ we get the usual relationship [11], but with a different constant factor. For completeness, we will give the proof of (11) in the remaining part of this subsection.

Proof. Define the *influence* [12] of variable i on the function f as

$$\mathrm{I}_i(f) \doteq \Pr[f(\mathbf{x})\neq f(\mathbf{x}\oplus i)]$$

where $\mathbf{x}\oplus i$ is the vector obtained from $\mathbf{x}$ by flipping its ith component. Note that

$$\mathrm{as}(f) = \sum_{1\le i\le n}\mathrm{I}_i(f), \tag{12}$$

(see [5] which also holds in our case). Further, for a function in $\mathcal{F}_n$, define the operator

$$\Delta_i f(\mathbf{x}) \doteq \begin{cases}(1-p)\cdot(f(\mathbf{x})-f(\mathbf{x}\oplus i)) & \text{if } \mathbf{x}_i = 1\\ p\cdot(f(\mathbf{x})-f(\mathbf{x}\oplus i)) & \text{if } \mathbf{x}_i = 0\end{cases}.$$

The operator Δ_i is designed such that

$$\Delta_i\Phi_{\mathbf{u}}(\mathbf{x}) = \begin{cases}\Phi_{\mathbf{u}}(\mathbf{x}) & \text{if } u_i = 1\\ 0 & \text{if } u_i = 0\end{cases}, \tag{13}$$

see [6]. Therefore, as Δ_i is a linear operator, for any function $f\in\mathcal{F}_n$

$$\begin{aligned}\Delta_i f &= \sum_{\mathbf{u}\in\Omega^n}\hat{f}(\mathbf{u})\Delta_i\Phi_{\mathbf{u}}\\ &= \sum_{\mathbf{u}:u_i=1}\hat{f}(\mathbf{u}).\end{aligned}$$

It can be easily checked that for a Boolean function $f\in\mathcal{B}_n$ the influence of the argument i can be get from the Euclidean norm of $\Delta_i\chi_f$

$$\begin{aligned}\mathrm{I}_i(f) &= \frac{1}{4p(1-p)}||\Delta_i\chi_f||_2^2\\ &= \frac{1}{4p(1-p)}\sum_{\mathbf{u}:u_i=1}\hat{\chi}_f(\mathbf{u})^2\end{aligned}$$

where the second line follows from Parseval's theorem and (13). Now (11) follows by summing up over all i (as of (12)). □

3.3. Applying noise

To describe the effect of the δ-noise to a function f we define the *noise operator* T_δ [5, 13] as follows

$$(T_\delta f)(\mathbf{x}) \doteq \mathbb{E}[f(\mathbf{N}_\delta(\mathbf{x}))].$$

From the linearity of the expectation, we immediately see that T_δ is a linear operator. Now it will become obvious, why it is more convenient to study T_δ. One can easily check that

$$T_\delta\Phi_{\mathbf{u}} = \delta^{|\mathbf{u}|}\Phi_{\mathbf{u}}, \tag{14}$$

hence the base vectors $\Phi_{\mathbf{u}}$ are Eigenvectors of T_δ with corresponding Eigenvalue $\delta^{|\mathbf{u}|}$ (see also [7]). The noise operator can now be used to compute ρ_f.

Lemma 3. *Assume that f is Boolean, then*

$$\rho_f = \frac{1}{2}\left(1-\langle T_\delta\chi_f|\chi_f\rangle\right). \tag{15}$$

Proof.

$$\begin{aligned}\langle \chi_f | T_\delta \chi_f \rangle &= \mathbb{E}[\chi_f \cdot T_\epsilon \chi_f] \\ &= \mathbb{E}[\chi_f(\mathbf{x}) \cdot \chi_f(N_\delta(\mathbf{x}))].\end{aligned}$$

For any binary random variable $X : \Omega^n \to \{-1, 1\}$

$$\mathbb{E}[X] = 1 - 2\Pr[X = -1],$$

and on the other hand

$$\Pr[\chi_f(\mathbf{x}) \cdot \chi_f(N_\delta(\mathbf{x})) = -1] = \rho_f.$$

□

The inner product on the left hand side of (15) is closely connected to the spectra of χ_f. Applying Fourier expansion together with the linearity of T_δ and its property (14) yields

$$\begin{aligned}\langle T_\delta \chi_f | \chi_f \rangle &= \sum_{\mathbf{u} \in \Omega^n} \hat{\chi}_f(\mathbf{u}) \delta^{|u|} \langle \chi_f | \Phi_\mathbf{u} \rangle \\ &= \sum_{\mathbf{u} \in \Omega^n} \delta^{|\mathbf{u}|} \hat{\chi}_f(\mathbf{u})^2.\end{aligned}$$

Hence we proved the following lemma:

Lemma 4. *Given $f \in \mathcal{B}_n$.*

$$\rho_f = \frac{1}{2}\left(1 - \sum_{\mathbf{u} \in \Omega^n} \delta^{|\mathbf{u}|} \hat{\chi_f}(\mathbf{u})^2\right). \tag{16}$$

Now we can finally prove our Lemma 1. In fact not much work is left (note also that (6) follows from above by differentiating).

Proof of Lemma 1. We recall that

$$\delta = 1 - \frac{\epsilon}{2p(1-p)}.$$

Note that for any natural number $x \geq 0$

$$\delta^x = \left(1 - \frac{\epsilon}{2p(1-p)}\right)^x$$

can be bounded as follows

$$\delta^x \geq 1 - \frac{x}{2p(1-p)}\epsilon.$$

Applying this inequality to (16) and remembering (10) we get Equation (5):

$$\begin{aligned}\rho_f &\leq \epsilon \cdot \frac{1}{4p(1-p)} \sum_{\mathbf{u} \in \Omega^n} |\mathbf{u}| \hat{\chi_f}(\mathbf{u})^2 \\ &= \epsilon \cdot \mathrm{as}(f)\end{aligned}$$

where the second line follows from (11). □

4. REFERENCES

[1] S. Kauffman, "Metabolic stability and epigenesis in randomly constructed nets," *Journal of Theoretical Biology*, vol. 22, pp. 437–467, 1969.

[2] J. F. Lynch, "Dynamics of random boolean networks," in *Current Developments in Mathematical Biology: Proceedings of the Conference on Mathematical Biology and Dynamical Systems*, R. C. K. Mahdavi and J. Boucher, Eds. 2007, pp. 15–38, World Scientific Publishing Co.

[3] J. Kesseli, P. Rämö, and O. Yli-Harja, "On spectral techniques in analysis of Boolean networks," *Physica D: Nonlinear Phenomena*, vol. 206, pp. 49–61, 2005.

[4] R. W. O'Donnell, *Computational Applications of Noise Sensitvity*, Ph.D. thesis, MIT, June 2003.

[5] J. Kahn, G. Kalai, and N. Linial, "The influence of variables on Boolean functions," in *Proc. 29th Ann. IEEE Foundations of Comp. Sci.* IEEE, Oct 1988, pp. 68–80.

[6] M. Talagrand, "On Russos's approximate zero-one law," *The Annals of Probability*, vol. 22, no. 3, pp. 1576–1587, 1994.

[7] E. Friedgut, "Boolean functions with low average sensitivity depend on few coordinates," *Combinatorica*, vol. 18, no. 1, pp. 27–35, 1998.

[8] G.-Z. Xiao and J. Massey, "A spectral characterization of correlation-immune combining functions," *Transaction on Information Theory*, vol. 34, no. 3, May 1988.

[9] S. Golomb, "On the classification of Boolean functions," *Transaction on Information Theory*, vol. 5, pp. 176–186, May 1959.

[10] S. Hurst, D. Miller, and J. Muzio, "Spectral method of Boolean function complexity," *Electronics Letters*, vol. 18, no. 13, pp. 572–574, June 1982.

[11] A. Bernasconi, *Mathematical Techniques for the Analysis of Boolean Functions*, Ph.D. thesis, Dipartimento di Informatica, Universita di Pisa, March 1998.

[12] M. Ben-Or and N. Linial, "Collective coin flipping," in *Randomness and Computation*, S. Micali, Ed. Academic Press, New York, 1990.

[13] I. Benjamini, G. Kalai, and O. Schramm, "Noise sensitivity of Boolean functions and applications to percolation," *Publications Mathématiques de l'Institut des Hautes Études Scientifiques*, vol. 90, pp. 5–43, 1999.

TOPMODULE: PATHWAY DETECTION IN BIOLOGICAL NETWORKS

Angela Simeone[1]*, Jacob Michaelson*[1]*, Antigoni Elefsinioti*[1] *and Andreas Beyer*[1]

[1] Biotechnology Center (BIOTEC), Technische Universität Dresden,
Tatzberg 47/49, 01307 Dresden, Germany

angela.simeone@biotec.tu-dresden.de, jacob.michaelson@biotec.tu-dresden.de,
antigoni.elefsinioti@biotec.tu-dresden.de, andreas.beyer@biotec.tu-dresden.de

ABSTRACT

Integrating high-throughput measurements with information about gene and protein interactions reduces noise and supports a mechanistic interpretation of the data. Data from phenotypic screens such as RNAi or genetic association studies can be mapped onto protein interaction networks to identify the pathways responsible for the observed phenotypes.
Here, we propose a new algorithm, termed TopModule, for the identification of functional modules in biological networks. The algorithm extracts statistically and biologically reliable subnetworks associated with the respective phenotypic effects. TopModule accounts for node and edge scores, considers the network topology when assessing significance of modules, and is designed to deal with sparse data. Example applications of TopModule to expression measurements, RNAi screens, and expression quantitative trait locus (eQTL) data are discussed.

1. MOTIVATIONS

The improvement and the automation of genome-wide screens provides a huge amount of data turning the data analysis and the data interpretation into a bottleneck [1]. The phenotypic screen data are subject to noise because the detection of the phenotype can be inaccurate or affected by off-target effects (example: RNAi screens). Moreover it can be difficult to explain observed genotype-phenotype associations exclusively based on phenotypic data.
In order to address these issues and to correctly understand the role of each gene in the specific (emergent) cellular process it is necessary to integrate the phenotypic information with other genomic and proteomic data, and to map these data onto functional networks [2]. Functional networks are a simple and useful graph representation of the interactions of biological entities (genes, proteins, metabolites) in living systems.
The aim of this method is to analyse phenotypic screening data in combination with gene interaction data in order to explain cellular processes and molecular functions at a molecular and genetic level. With this approach it is possible to generate concrete hypotheses about the underlying mechanism governing the observed phenotypic changes.
Here we assume screens associating a quantitative phenotype with individual genes. Examples of such screens are RNAi screens or genome wide association studies.
One way to integrate phenotypic screening data with network information is to screen the interaction network for active subnetworks, i.e. connected regions of the network related to significant changes of the observed phenotype [2]. Such modules would consist of components involved in the same or related pathways.

2. METHOD

The method proposed here, TopModule, works using an interaction network and the phenotypic data. The interaction network represents the model of how elements of the whole system act together. The output of TopModule is a set of statistically significant modules that could explain the observed phenotype.
TopModule allows for using quantitative edge scores indicating the confidence level of the interaction. The network represents the scaffold on which TopModule works. The first step of the method is to map the phenotypic data onto this scaffold. The mapping is done assigning the phenotypic values to the genes (nodes) as attributes. In this way we obtained a network where:

- the edge scores are the confidence levels taken from the database;
- the node scores represent each gene's contribution to the observed phenotype.

TopModule applies a greedy search for finding connected regions of the network that show significantly enriched node scores.
There are two pre-processing steps before performing the searching:

1. Rescaling of all node scores using a smoothing procedure (optional step, Fig.1a). The goal of rescaling is to correct for gaps in the network. The rescaling takes into account scores of neighboring nodes.
2. Definition of a seed node list on the basis of the user preferences (known relevant genes, best scoring nodes, etc. ...). Each node of this set will itera-

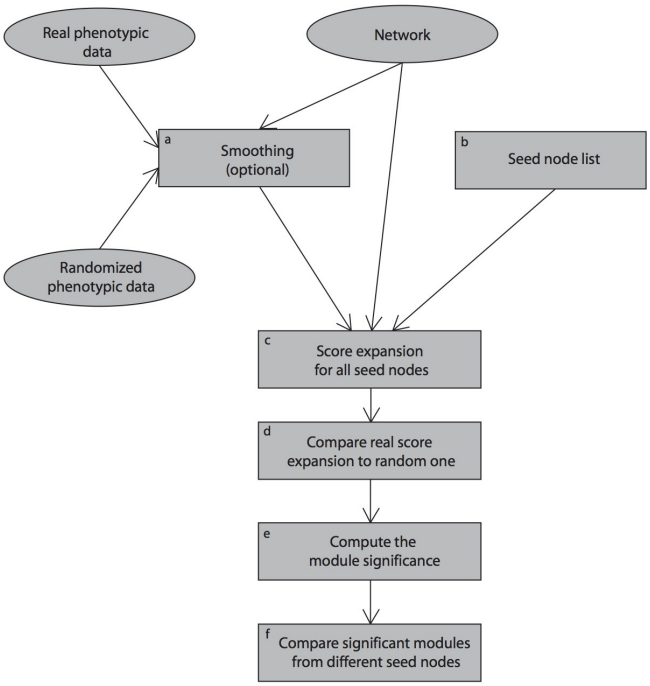


Figure 1. TopModule pipeline.

tively be the starting point for the searching procedure (Fig.1b).

TopModule proceeds as follows :

1. Score expansion using the real (non randomized) phenotypic data (Fig.1c). Starting from the seed the searching procedure moves in the direction of the highest scoring neighbour (best score) of the current module. Nodes are iteratively added and after each iteration a module score is computed (average node score).

2. A user defined number of score expansions on randomly shuffled node scores (Fig.1d). Each score expansion starts from the same topological seed node.

3. Assessment of the statistical significance of each module (Fig.1e-f). This assessment is done by comparing the module score to scores obtained from the score expansion on the randomized data.

For each seed node TopModule determines a potential module along with its statistical significance. Modules can be compared across different seed nodes to gain higher statistical confidence.

3. RESULTS

We report and discuss the results obtained after applying TopModule to RNAi screens, quantitative trait loci (QTL) data and expression measurements.

3.1. RNAi data

We have applied TopModule to a recently published RNAi screen [3]. After the initial genome-wide screen (primary screen) more refined experiments have been done for a smaller set of candidate genes. Finally 1351 genes were assigned to four functional classes: G0/G1 arrest, S arrest, G2 arrest and cell division defects. The remaining genes were not classified [3].

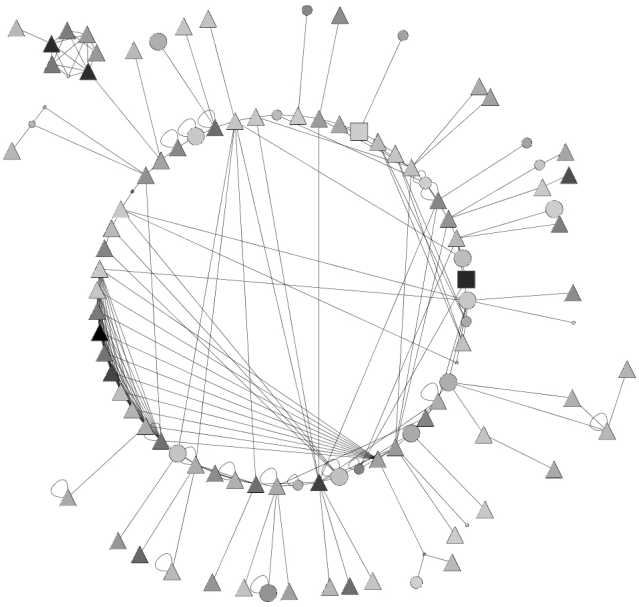


Figure 2. The most significant module obtained using TopModule on RNAi data. The colour of the nodes codifies the strength of the phenotype observed in the initial genome-wide screen (DNA content in G1 phase). Darker colour indicates a stronger phenotype. Triangles are genes which were confirmed to be G1-arrest related in detailed subsequent experiments. Circles are genes that were not tested/verified even if they showed a G1-arrest phenotype. Squares are genes identified as S-arrest related (there are only 2).

We hypothesized that TopModule would be able to find the 'true' cycling genes by identifying a connected module of significant hits from the first (genome-wide) screen. Hence, we applied TopModule to the G1-arrest phenotype (DNA content in the G0/1 phase) by mapping the phenotype scores onto a high confidence human interaction network consisting of STRING [4] (edge confidence scores > 0.7) and the in-vivo interactions from HPRD [5] (9513 nodes, 64019 interactions).

TopModule identified a significant module of 106 nodes (Fig.2) containing 76 genes that were confirmed in the detailed experiments of the original study (71.7%, P-value $< 10^{-16}$). Hence, TopModule was able to identify a significant number of true G1 arrest-related genes exclusively based on the primary screen and without further experimental testing. Also, the network module contained genes that did not show a significant phenotype in the primary screen, but which were later confirmed to be also related to G1-arrest [3], i.e. TopModule was able to detect false negatives. We compared TopModule results with the results of another published method called Active Module[2]. Performing a seaching with this second method we obtained a

significant module of 106 nodes (nodes appearing in more than 50% of the searches) containing only 47 genes that were confirmed to be G1-arrest related (40%, P-value= 0.36). Hence, TopModule performed better than Active Module.

3.2. eQTL data

TopModule was applied to expression quantitative trait loci (eQTL) data to assess its ability to recapitulate known regulatory pathways [6]. Genotype and expression data were downloaded from WebQTL [7]. Random Forests [8], a tree-based ensemble classification and regression method, has previously been used to identify genetic loci linked to phenotypes [9]. Using Random Forests in a genetic linkage study has several benefits, including implicit cross-validation of the model, the inclusion of many loci simultaneously in a predictive model, and the natural context tree-based regression provides for using categorical variables (markers) to predict a continuous value (expression). Here we used Random Forests to calculate the contribution of each marker to the accurate prediction of expression. Each marker was assigned an importance score which indicates how much the model's predictive ability is reduced when the marker's values are randomly permuted. This importance value was divided by its standard error to obtain a Z-score, from which a P-value is calculated.
Two approaches for identifying significant eQTL were compared. First, we calculated the Benjamini-Hochberg FDR [10] from the obtained P-values, and used FDR < 0.01 as a cut-off for significant loci. Since there are more genes in the network than markers, we mapped the marker scores to the genes in the network by assigning each gene the score of the nearest marker. Second, we used the unadjusted P-values as input for TopModule. Again, we mapped marker scores to genes as previously described. TopModule was performed using the STRING [4] network as the topology (scores > 0.7). Thirty seed nodes were selected based on their ability to lead to high-scoring modules within 20 steps. Score expansions were cut at the last local minimum before exceeding a permutation-based P-value of 0.01 (1000 permutations). A smoothing coefficient of 0.25 was used. Modules were defined as comprising nodes appearing in more than 50% of the searches that achieved significance at the stated P-value < 0.01 level.
We evaluated eQTL by examining the expression of the ID1 gene, a well-known target of the TGF-β signaling pathway. Using an FDR cut-off of 0.01, we identified 6 genes as putative eQTL for ID1 (Ltbp1, Wisp1, Myc, Sla, Ddef1, and Tg), two of which are annotated in the TGF-β pathway (Ltbp1 and Myc). Using TopModule, we identified a significant module of 6 genes (Ltbp1, Tgfb1, Bmp2, Thbs3, Comp, and Smad3), all of which are upstream of ID1 in the TGF-β pathway. Interestingly, several of the module nodes did not have significant P-values ($P < 0.05$) when considered in isolation, and would certainly have been excluded from further investigation under a cut-off approach. However, when considered together in their network context, they yield both statistical and biological significance.

3.3. Expression data

During differentiation progenitor cells significantly change concentrations of a range of proteins that are relevant for the new character of the cells. Czupalla et al. [11] have investigated the changes of protein and mRNA concentrations in differentiating osteoclasts. The model system for these experiments was the mouse myeloid Raw 264.7 cell line which differentiates *in vitro* into osteoclasts. The authors found very little overlap between genes that significantly increased their mRNA concentrations and those that increased protein levels. We applied TopModule in order to better understand the causes and mechanisms of this finding. Here, we focus on genes with either elevated mRNA or protein levels in the fully differentiated osteoclasts.
We applied TopModule independently using first protein concentrations and subsequently mRNA concentrations as node scores. The interaction network was obtained from STRING [4] (edge confidence scores > 0.7). TopModule determined two distinct network modules (Fig.3) that are regulated either transcriptionally or post-transcriptionally. Finally, we compared the resulting network modules.
The protein and mRNA modules contained 119 and 100 genes, respectively, with only one node (gene) being common to both modules.

Interactions in the protein module are more dense than those in the mRNA one (densities 0.11 and 0.06 respectively), indicating that the first module presumably comprises proteins that form complexes whereas the latter genes probably belong to one or more biological pathways present in mature osteoclasts. Although there are many interactions connecting the two modules, they are clearly distinct sub-networks with a higher connectivity within than between modules (density of the complete network in Fig.3: 0.04). Next, we searched for the enrichment of certain gene functions in the two modules. We found that genes belonging to the protein module are primarily located in the mitochondrion - which can perhaps explain the high connectivity of the subnetwork. On the other hand, genes from the mRNA module are mainly located in ER, Golgi apparatus, lysosomes and hyperoxysomes, which are all known to be involved in the specific functioning of osteoclasts.

4. CONCLUSION

We presented here a novel method, TopModule, to identify statistically and biologically significant subnetworks in biological networks using a greedy search. This method takes into account not only the network topology but also the node scores. One important feature of TopModule is its broad applicability. We have tested it on different kinds of phenotypic data: RNAi, eQTL and gene expression data. In all cases the main benefit of using TopModule is its ability to identify interactions between relevant genes, which may lead to detailed hypotheses about underlying

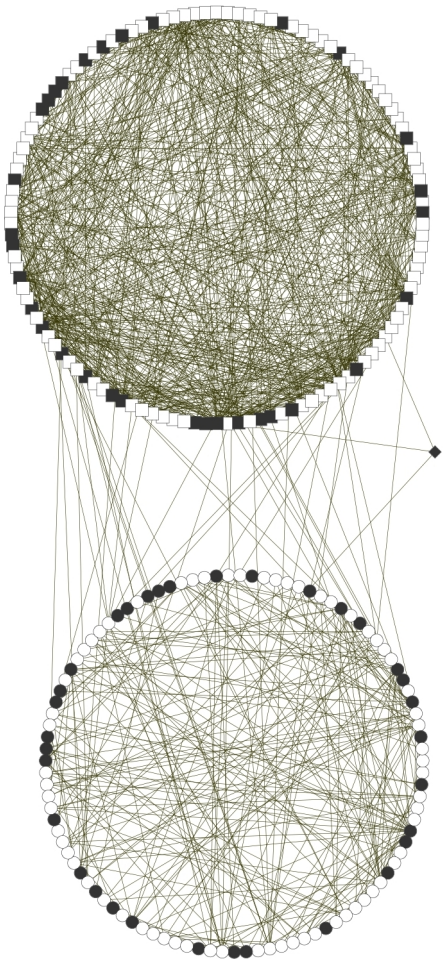


Figure 3. Network modules with significantly elevated protein (**top**) and mRNA (**bottom**) levels in maturated osteoclasts. The only gene present in both modules is shown between them (**diamond node shape**). Node colour indicates measured concentration changes. Black: significantly elevated protein/mRNA concentration. White nodes did not show a significant concentration change in the experiments.

molecular mechanism. TopModule also showed the ability to detect false positives and false negatives present in the original experimental data.

5. ACKNOWLEDGMENTS

We would like to acknowledge funding from the Klaus Tschira Foundation and from the Helmholtz Alliance on Systems Biology (Network: Contaminant Molecules).

6. REFERENCES

[1] A. Beyer, S. Bandyopadhyay, and T. Ideker, "Integrating physical and genetic maps: from genomes to interaction networks," *Nature Reviews Genetics*, vol. 8, pp. 699–710, 2007.

[2] T. Ideker, O. Ozier, B. Schwikowski, and A. Siegel, "Discovering regulatory and signalling circuits in molecular interaction networks," *Bioinformatics*, vol. 18, pp. 233–240, 2002.

[3] R. Kittler, L. Pelletier, A.-K. Heninger, M. Slabicki, M. Theis, L. Miroslaw, I. Poser, S. Lawo, H. Grabner, K. Kozak, J. Wagner, V. Surendranath, C. Richter, W. Bowen, A. L. Jackson, B. Habermann, A. A. Hyman, and F. Buchholz1, "Genome-scale rnai profiling of cell division in human tissue culture cells," *Nature Cell Biology*, vol. 9, pp. 1401 – 1412, Nov. 2007.

[4] C. von Mering, L. J. Jensen, M. Kuhn, S. Chaffron, T. Doerks, B. Krüger, B. Snel, and P. Bork, "String 7—recent developments in the integration and prediction of protein interactions," *Nucleic Acids Research*, vol. 35, pp. D358–D362, Nov. 2007.

[5] G. R. M. e t al., "Human protein reference database—2006 update," *Nucleic Acids Research*, vol. 34, pp. D411–D414, Jan. 2006.

[6] M. Kanehisa, M. Araki, S. Goto, M. Hattori, M. Hirakawa, M. Itoh, T. Katayama, S. Kawashima, S. Okuda, T. Tokimatsu, and Y. Yamanishi, "Kegg for linking genomes to life and the environment," *Nucleic Acids Research*, vol. 36(Database issue), pp. D480–D484, 2008.

[7] J. Wang, R. W. Williams, and K. F. Manly, "Webqtl: web-based complex trait analysis.," *Neuroinformatics*, vol. 1, no. 4, pp. 299–308, 2003.

[8] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[9] A. Bureau, J. Dupuis, K. Falls, K. L. Lunetta, B. Hayward, T. P. Keith, and P. Van Eerdewegh, "Identifying SNPs predictive of phenotype using Random Forests," *Genet. Epidemiol.*, vol. 28, no. 2, pp. 171–82, 2005.

[10] Y. Benjamini and Y. Hochberg, "Controlling the false discovery rate: A practical and powerful approach to multiple testing," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 57, no. 1, pp. 289–200, 1995.

[11] C. Czupalla, H. Mansukoski, T. Pursche, E. Krause, and B. Hoflack, "Comparative study of protein and mrna expression during osteoclastogenesis," *Proteomics*, vol. 5, pp. 3868 – 3875, Oct. 2005.

ADAPTIVE MATRIX METRICS FOR ATTRIBUTE DEPENDENCE ANALYSIS IN DIFFERENTIAL HIGH-THROUGHPUT DATA

M. Strickert[1*], *K. Witzel*[1], *J. Keilwagen*[1], *H.-P. Mock*[1], *P. Schneider*[2], *M. Biehl*[2], *and T. Villmann*[3]

[1]Leibniz Institute of Crop Plant Research Gatersleben, Germany,
[2]Institute for Math. and Computer Science, University of Groningen, NL,
[3]Research group Computational Intelligence, University of Leipzig, Germany.
[*]Corresponding author email: stricker@ipk-gatersleben.de

ABSTRACT

Data-driven metric adaptation is proposed for proteome analysis of 2D-gel electrophoretic plots aiming at identification of stress related proteins in two barley cultivars with different response towards different salt stress conditions. Gradient descent is applied to the ratio of intra- and inter-class distance sums to optimize the matrix parameters of generalized Mahalanobis distances in order to separate the several hundred dimensional data of protein intensities in the transformation space. The resulting matrix contains mutual dependence of spots, explaining differential stress reactions and putative protein interactions. We present interesting results obtained by the new metric learning method that possesses general applicability in biomedical data analysis.

Keywords: Supervised feature characterization, adaptive matrix metric, attribute dependence modeling.

1. INTRODUCTION

The identification of gene and protein dependences is an essential step for the inference of interaction networks from experimental data. Both network inference and the exploration of the obtained connectivity structure are hot topics in systems biology [5]. Typical approaches for the automatic reconstruction of network topologies make use of correlation measures [4], Bayesian inference [9], or information theoretic statistics [8] in order to model the mutual dependence of network nodes. Among different beneficial properties these models also possess some unwanted properties, ranging from being rather simplistic or suffering from high computational complexity to requiring additional assumptions like density estimates. The assessment of the quality of the inferred networks is usually problematic. One general reason is that test statistics might be inappropriate for reflecting biological experience [3]. A more specific problem is the biological probing and confirmation of the huge number of potential interaction partners. Alternatively, the promising concept of learning metrics from the area of machine learning research [6, 13] can be utilized for network construction by modeling attribute pairs. Data-driven metric adaptation also helps to reduce the curse of dimensionality occurring during the analysis of high-throughput data. In our case, protein data of 2D electrophoretic gels are considered providing intensities of many protein spots measured in a relatively low number of available experiments. A minimalistic attribute characterization method is used for rating the influence of attribute pairs on the spatial arrangement of class-specific data clouds in vector space, expressed by an adaptive matrix metric, as recently utilized in matrix learning vector quantization [11]. The method presented here aims at minimizing within-class differences while maximizing inter-class distances by rescaling the data space based on a trained transformation matrix without building an explicit classification model [12]. Although this aim resembles the one of linear discriminant analysis (LDA) [2], the transform maintains the original data dimensionality and is thus not reduced to an a priori low-dimensional LDA subspace. The new method yields an estimate of a label-specific inverse covariance matrix and might be considered as supervised whitening operation. Some concepts can be related to the threshold gradient descent method [7].

2. METHOD – MATRIX LEARNING

As input q-dimensional row vectors $\mathbf{x} \in \mathbb{R}^{1\times q}$ are assumed to be taken from a set containing n data vectors $\{\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^n\}$. The proposed metric adaptation requires a class-specific label $\mathrm{c}(k)$ for each data vector $\mathbf{x}^k$. We define the main building block of the method, the matrix-based metric $\mathrm{d}^{ij}_{\mathbf{\Omega}} \in [0;\infty)$ for data vectors $\mathbf{x}^i$ and $\mathbf{x}^j$:

$$\mathrm{d}^{ij}_{\mathbf{\Omega}} = \mathrm{d}_{\mathbf{\Omega}}(\mathbf{x}^i, \mathbf{x}^j) = (\mathbf{x}^i - \mathbf{x}^j) \cdot \mathbf{\Lambda} \cdot (\mathbf{x}^i - \mathbf{x}^j)^{\mathsf{T}}\,, \quad (\mathbf{\Lambda} = \mathbf{\Omega} \cdot \mathbf{\Omega}^{\mathsf{T}}) \in \mathbb{R}^{q\times q}\,. \qquad (1)$$

The identity matrix $\mathbf{\Lambda} = \mathbf{\Omega} = \boldsymbol{I}$ induces the special case of the squared Euclidean distance; other diagonal matrices yield weighted squared Euclidean distances. Arbitrary positive-definite matrices $\mathbf{\Lambda}$ lead to very general metrics that can express rotation and translation which do not affect distances between points, and scaling and shearing which do affect them. A triangular or symmetric matrix $\mathbf{\Omega}$ would be sufficient to express any such configuration by Eqn. 1. Faster convergence can be observed, though, if the full matrix is adapted in the matrix optimization scheme

for minimizing the label-specific metric stress criterion:

$$\mathrm{s}(\mathbf{\Omega}) \quad := \frac{\sum_{i=1}^{n}\sum_{j=1}^{n} \mathrm{d}_{\mathbf{\Omega}}(\mathbf{x}^i,\mathbf{x}^j)\cdot\delta_{ij}}{\sum_{i=1}^{n}\sum_{j=1}^{n} \mathrm{d}_{\mathbf{\Omega}}(\mathbf{x}^i,\mathbf{x}^j)\cdot(1-\delta_{ij})} = \frac{\mathrm{d}_C}{\mathrm{d}_D}$$
$$\text{with} \qquad \delta_{ij} = \begin{cases} 0 : \mathrm{c}(i)\neq\mathrm{c}(j) \\ 1 : \mathrm{c}(i)=\mathrm{c}(j) \end{cases} . \tag{2}$$

Distances $\mathrm{d}_{\mathbf{\Omega}}^{ij}$ between all n data vectors $\mathbf{x}^i$ and $\mathbf{x}^j$ depend on the adaptive matrix parameters $\mathbf{\Omega} = (\Omega_{kl})_{\substack{k=1\ldots q \\ l=1\ldots m}}$ of interest. The numerator represents within-class data variability, which should be small. The denominator is related to inter-class distances, which should be large. Thus, optimization of $\mathrm{s}(\mathbf{\Omega})$ handles both parts of the fraction simultaneously. Compromise solutions must be found in cases when within-class variation, potentially caused by outliers, needs compression, while inter-class separability would require inflation.

Although similar at first glance, the proposed approach is structurally different to LDA, because the inverse LDA-like ratio in Eqn. 2 is optimized in the original data space, not in the projection to the most prominent class separating LDA direction [12]. In contrast to LDA where covariance matrices and class centers can be initially computed and then reused, this is not possible in the proposed method, because the metric adaptation affects both class centers and data covariances. Full matrix adaptation, though, creates higher computational demands of the optimization method described in the following.

The cost function $\mathrm{s}(\mathbf{\Omega})$ gets iteratively minimized by gradient descent. This requires adaptation of the matrix $\mathbf{\Omega}$ in small steps γ into the direction of steepest gradient

$$\mathbf{\Omega} \leftarrow \mathbf{\Omega} - \gamma \cdot \frac{\partial \mathrm{s}(\mathbf{\Omega})}{\partial \mathbf{\Omega}} \tag{3}$$

obtained by the chain rule

$$\frac{\partial \mathrm{s}(\mathbf{\Omega})}{\partial \mathbf{\Omega}} = \sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\partial \mathrm{s}(\mathbf{\Omega})}{\partial \mathrm{d}_{\mathbf{\Omega}}^{ij}} \cdot \frac{\partial \mathrm{d}_{\mathbf{\Omega}}^{ij}}{\partial \mathbf{\Omega}} . \tag{4}$$

The derivative of the fraction $\mathrm{s}(\mathbf{\Omega}) = \mathrm{d}_C/\mathrm{d}_D$ in Eqn. 2 is

$$\begin{aligned} \frac{\partial \mathrm{s}(\mathbf{\Omega})}{\partial \mathrm{d}_{\mathbf{\Omega}}^{ij}} &= \frac{\delta_{ij}\cdot \mathrm{d}_D}{\mathrm{d}_D^2} + \frac{(\delta_{ij}-1)\cdot \mathrm{d}_C}{\mathrm{d}_D^2} \\ &= \begin{cases} 1/\mathrm{d}_D \ : \ \mathrm{c}(i)=\mathrm{c}(j) \\ -\mathrm{d}_C/\mathrm{d}_D^2 \ : \ \mathrm{c}(i)\neq\mathrm{c}(j) \end{cases} . \end{aligned} \tag{5}$$

The right factor in Eqn. 4 is the matrix derivative of Eqn. 1:

$$\frac{\partial \mathrm{d}_{\mathbf{\Omega}}^{ij}}{\partial \mathbf{\Omega}} = 2\cdot(\mathbf{x}^i-\mathbf{x}^j)^{\mathsf{T}}\cdot(\mathbf{x}^i-\mathbf{x}^j)\cdot\mathbf{\Omega} . \tag{6}$$

In practice, the gradient from Eqn. 4, is computed and reused as long the cost function decreases. Increase of $\mathrm{s}(\mathbf{\Omega})$ triggers a recomputation of the gradient. The step size γ is dynamically determined as the initial size γ_0, being exponentially cooled down by rate η, divided by the maximum absolute element in the matrix $\partial \mathrm{s}(\mathbf{\Omega})/\partial\mathbf{\Omega}$.

For running the iterative optimization, the initial step size γ_0 can be chosen as a value below one, such as 0.01 used here. In general, between 50 and 2500 iterations are necessary, depending on the saturation characteristics of the logged cost function value. It was set to 50 in this study. The exponential cooling rate was set to $\eta = 0.995$. For initialization of matrix $\mathbf{\Omega}$ random matrix element sampling from uniform noise in the interval $[-0.5; 0.5]$ is proposed as first step. This noise matrix $\boldsymbol{A} \in \mathbb{R}^{q\times q}$ is then broken by QR-decomposition into $\boldsymbol{A} = \boldsymbol{Q}\cdot\boldsymbol{R}$, of which the $\boldsymbol{Q}$-part is known to form an orthonormal basis with $\boldsymbol{Q}\cdot\boldsymbol{Q}^{\mathsf{T}} = \boldsymbol{I}$. Thus, although $\mathbf{\Omega} = \boldsymbol{Q}$ contains random configurations, its self-product leads to the intuitive squared Euclidean distance in the beginning of optimization.

3. RESULTS – PROTEOME DATA ANALYSIS

Abiotic stress factors have severe effects on the growth as well as on the yield of crop plants, and proteome analysis of stress responses is widely used for unraveling tolerance mechanisms for crop improvement [1, 10]. Our data has been created in a proteomic study concerning metabolic reactions of two barley cultivars, Steptoe and Morex, to different salt stress conditions, ranging from zero NaCl concentration via 100mM to 150mM. The main task is the identification of protein pairs in root parts affected by salt stress, but with different regulation dynamics between the salt-sensitive Steptoe line and the salt-tolerant Morex line. Using 2D-gels separating along pH and mass gradient, images with protein-specific spot distributions were obtained. After image processing, a number of 997 common spots in all gel images was obtained for further analysis. Since three technical replicates per experimental condition were taken, a total number of 18 images was available for differential analysis of the spot combinations characteristic of the three salt treatments.

Matrix learning has been done independently for the Morex and Steptoe lines. In order to increase the reliability of the results, 100 repetitions with random matrix initializations have been created, leading to a total number of 200 trained 997x997 matrices $\mathbf{\Lambda}_i = \mathbf{\Omega}_i\cdot\mathbf{\Omega}_i^{\mathsf{T}}$. Within each such symmetric matrix the ranks of its lower triangular elements, including the diagonal, were calculated. Especially high and low ranks are linked to protein pairs separating between the three salt stress conditions. Since metabolic differences of Steptoe and Morex regarding salt treatments are looked for, only those pairs with very different ranks between both lines are of interest. Thus, the absolute differences of average ranks of the 100 Steptoe and 100 Morex results were taken as ordering criterion of all protein pairs. For illustration, the top 100 protein pairs are considered in more detail. In that list all standard deviations of ranks are below 12.8, which indicates a high reproducibility of the found protein pairs; for comparison, the expectation of randomly drawn rank differences would be $1/3\cdot 997\cdot(997+1)/2 = 165834.3$.

The connectivity structure of the strongly associated top 100 protein pairs is shown in Fig. 1. Two protein spots, 543 and 94, can be identified as network hubs. These are linked to many other spots of interest. Spots within bold ellipses were identified as candidate proteins in an

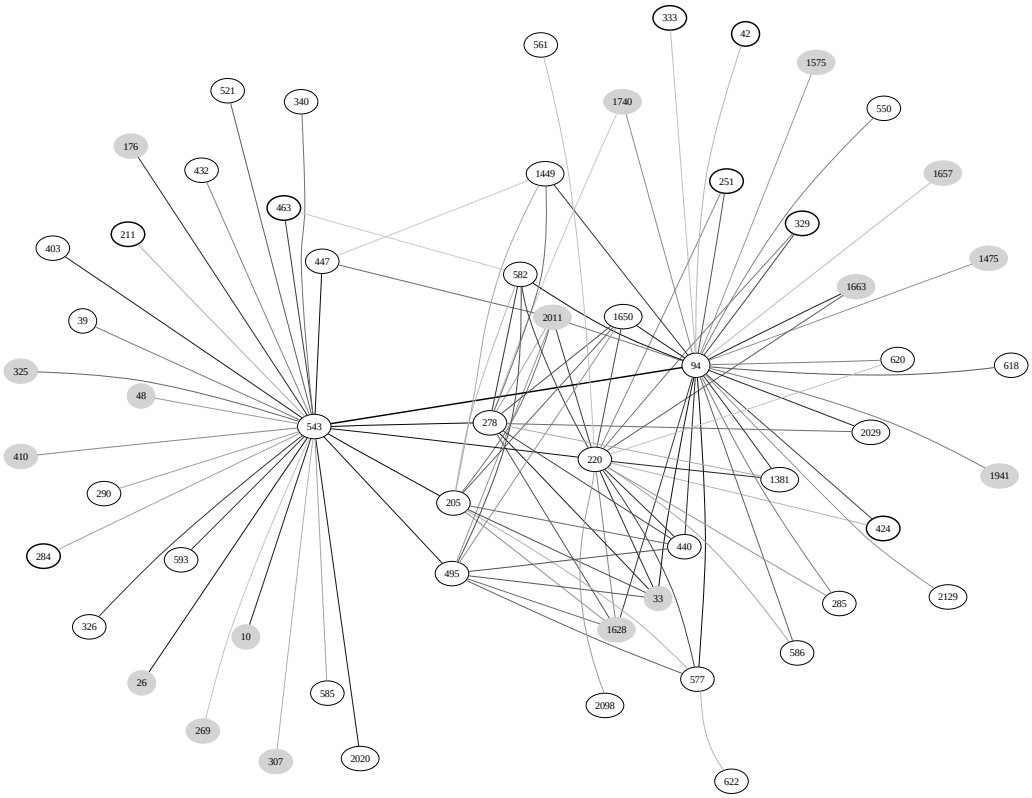


Figure 1. Protein-protein network derived from 2D gels containing patterns of differential protein abundance induced by salt-stress. The top 100 dependent pairs of protein spots, indexed by numbers, are shown. Edge gray levels indicate the ranking, the darker the stronger. The connection from 543 to 94 is the strongest. Bold face ellipses denote spots identified as interesting in previous studies, gray filling indicates spots close to the background intensity.

independent previous study. Plain ellipses are new candidates that have not been detected previously by single spot analysis. It must be stated, though, that also spots close to the background intensity have been found. These, shaded in gray, cannot be considered as biologically relevant. Yet, scale-free inspection indicates that magnitudes alone are not the only consistent class-separation criteria.

Model compression. Since the matrix model of 997x997 is huge in contrast to the (2x9) x 997 experimental protein spots, compression is an important issue. Eigen decomposition of $\mathbf{\Lambda} = \boldsymbol{S} \cdot \mathbf{W} \cdot \mathbf{W}^{-1}$ into the diagonal eigenvalue matrix $\boldsymbol{S}$ and the eigenvector matrix $\mathbf{W}$ helps to reach substantial reduction. For the protein-specific matrices the largest eigenvalues are about 7-fold greater than their predecessors which themselves are twice larger than their predecessors. These first two eigenvectors $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ therefore define outstanding directions in the scaling matrix $\mathbf{\Lambda}$. This matrix can be approximately reconstructed by $[\boldsymbol{w}_1\boldsymbol{w}_2] \cdot [\boldsymbol{w}_1\boldsymbol{w}_2]^\intercal$. Each experiment $\mathbf{x}$ is projected into a class-separating subspace by $\mathbf{x} \cdot [\boldsymbol{w}_1\boldsymbol{w}_2]^\intercal$. This is shown in the right panel of Fig. 2, where the within-class variation of the technical repetitions is virtually completely suppressed in contrast to the scatter plot obtained by ordinary PCA projection, displayed in the left panel of Fig. 2. This result indicates that relevant directions for noise cancellation have been found by matrix learning.

4. CONCLUSIONS

The presented matrix metric learning approach offers a new way to extracting biomarkers, advancing the traditional assessment of individual data attributes to attribute pairs. As illustrated for protein data, dependent treatment-specific substances can be identified. This allows the construction of undirected network structures with weighted edges, a first step towards the inspection of possible protein interactions. Multi-parallel data sources like the considered protein gels create big challenges, because the number of experiments are usually substantially lower than the number of attributes. Therefore, metric adaptation is generally considered as beneficial to counter-act the curse of dimensionality. Confidence in the proposed method is derived from the observation that training showed very stable results despite random initializations of $\mathbf{\Omega}$. However, additional data for the validation of the trained metric are needed, and attention must be put to the role of pairs with low-intensity partners. In order to force further model regularization and for a significant speedup of adaptation, the direct training of only the first k eigenvectors of $\mathbf{\Lambda}$ are currently considered.

Thanks to the anonymous reviewer for the valuable comments. The work is supported by grant XP3624HP/0606T, Ministry of Culture Saxony-Anhalt, Germany.

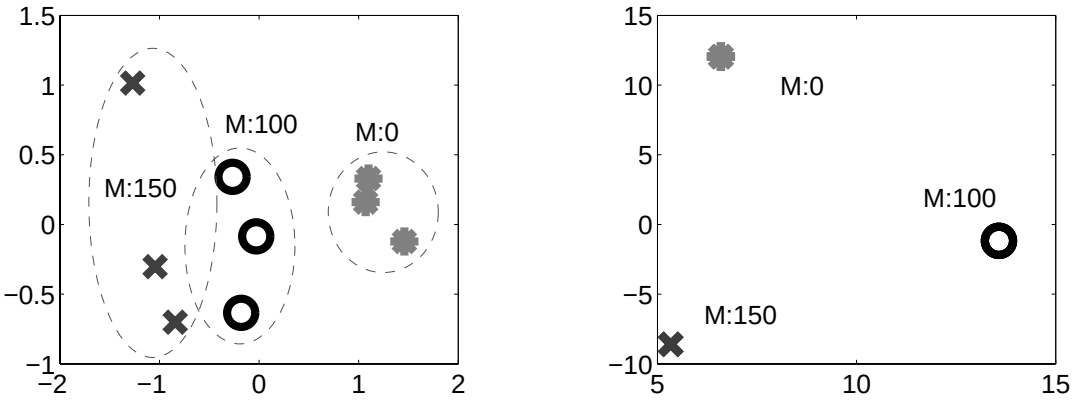


Figure 2. Scatter plots of 2D gels of Morex roots under salt stress. Left: PCA projection of original data to second vs. first eigenvector of data covariance matrix. Right: projection to second vs. first eigenvector of the trained metric matrix $\mathbf{\Omega} \cdot \mathbf{\Omega}^{\mathsf{T}}$. Labels M:0–M:150 denote salt stress concentrations in mM NaCl. The three technical replicates, belonging to specific salt levels, constitute a class.

5. REFERENCES

[1] S. Amme, A. Matros, B. Schlesier, and H.-P. Mock. Proteome analysis of cold stress response in arabidopsis thaliana using dige-technology. *Journal of Experimental Botany*, (57):1537–1546, 2006.

[2] C. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[3] D. Johnson. The insignificance of statistical significance testing. *Journal of Wildlife Management*, 63(3):763–772, 1999.

[4] F. Jourdan, R. Breitling, M. P. Barrett, and D. Gilbert. MetaNetter: inference and visualization of high-resolution metabolomic networks. *Bioinformatics*, 24(1):143–145, 2008.

[5] B. Junker and F. Schreiber. *Analysis of Biological Networks*. John Wiley and Sons, 2008.

[6] S. Kaski. From learning metrics towards dependency exploration. In M. Cottrell, editor, *Proceedings of the 5th International Workshop on Self-Organizing Maps (WSOM)*, pages 307–314, 2005.

[7] H. Li and J. Gui. Gradient directed regularization for sparse Gaussian concentration graphs, with applications to inference of genetic networks. *Biostatistics*, 7(2):302–317, 2006.

[8] P. E. Meyer, K. Kontos, F. Lafitte, and G. Bontempi. Information-theoretic inference of large transcriptional regulatory networks. *EURASIP J. Bioinformatics Syst. Biol.*, 2007(1):8–8, 2007.

[9] B.-E. Perrin, L. Ralaivola, A. Mazurie, S. Bottani, J. Mallet, and F. d'Alche Buc. Gene networks inference using dynamic Bayesian networks. *Bioinformatics*, 19:138–148, 2003.

[10] M. Rossignol, J.-B. Peltier, H.-P. Mock, A. Matros, A. Maldonado, and J. Jorrn. Plant proteome analysis: A 2004-2006 update. *Proteomics*, (6):5529–5548, 2006.

[11] P. Schneider, M. Biehl, and B. Hammer. Relevance Matrices in LVQ. In M. Verleysen, editor, *European Symposium on Artificial Neural Networks (ESANN)*, pages 37–42, Bruges, Belgium, 2007.

[12] M. Strickert, P. Schneider, J. Keilwagen, T. Villmann, M. Biehl, and B. Hammer. Discriminatory data mapping by matrix-based supervised learning metrics. In L. Prevost, S. Marinai, and F. Schwenker, editors, *Lecture Notes in Computer Science, LNCS 5065*, to appear. Springer, 2008.

[13] K. Weinberger, J. Blitzer, and L. Saul. Distance metric learning for large margin nearest neighbor classification. In Y. Weiss, B. Schölkopf, and J. Platt, editors, *Advances in Neural Information Processing Systems 18*, pages 1473–1480. MIT Press, Cambridge, MA, 2006.

ANALYSIS OF BIOLOGICAL NETWORK DATA USING LIKELIHOOD-FREE INFERENCE TECHNIQUES

Carsten Wiuf[1], *Oliver Ratmann*[2] *and Michael Knudsen*[1]

[1]Bioinformatics Research Center, University of Aarhus, 8000 Aarhus C, Denmark
[2]Department of Public Health and Epidemiology, Imperial College London, W2 1PG London, UK
wiuf@birc.au.dk, micknundsen@gmail.com, o.ratmann@imperial.ac.uk

ABSTRACT

Biological Networks have received much attention in recent years, but statistical tools for network analysis are still in their infancy. In this paper we focus on Protein Interaction Networks (PINs) that typically comprise thousands of proteins and interactions. PINs are the result of long evolutionary histories. Here we adopt simple mathematical models that capture essentials of protein evolution and develop statistical methods to estimate evolutionary PIN parameters. Our initial approach is based on a recursion for the likelihood, but it becomes computationally intractable for reasonably sized networks. Our second approach is based on summary statistics and likelihood-free inference. We discuss problems with selection of summaries, convergence, and credibility and apply the methods on *Helicobacter pylori* and *Plasmodium falciparum* data.

1. INTRODUCTION

Today it is possible to obtain massive amounts of data relating to the molecular complexity, organization and structure of a single cell or organism. These data can be obtained in a single experiment and have thus geared the biosciences towards system-level science or systems biology, where the attempt is to understand the system and its organization in broader and overall terms, rather than understanding the system's individual components one by one.

One system-level data type that is becoming available is PIN data. A PIN data set is a collection of experimentally determined interactions (physical binding between proteins). As such, a PIN data set is an incomplete observation of the interactome, the entire collection of all proteins in a cell or organism together with their interactions.

Evolution has shaped the form of an organism's interactome. In principle, we should therefore be able to learn about the processes responsible for this evolution by analyzing PIN data sets from the organism. The idea is that different evolutionary processes leave different traces in the PIN data but also that parameters describing the processes may differ between organisms. For example the authors of [1] investigate which type of model best explains a *D. melonagaster* PIN data set. However, they do not attempt to estimate the parameters in the models, but base their conclusions on how well the models (evaluated over a range of parameters) account for the motifs seen in the PIN data.

In [2] (and references therein) different distributions are fitted to the degree sequence observed in various PIN data sets. While this provides insight into the differences between organisms, it does not provide insight into the processes generating the differences – simply because the distributions are not based on evolutionary models. The approach taken in [1] has the strength that it utilizes more information in the data than just the degree sequence and thus has a higher chance of uncovering relevant features.

In this paper we present statistical analysis of PIN data sets based on mathematical model of network evolution. We focus on two data sets, a *H. pylori* data set [3] and a *P. falciparum* data set [4]. Statistical analysis of network data is far from straightforward and we discuss different approaches to inference [5, 6]. We first develop a scheme for maximum likelihood inference using a full data set (i.e. an entire network), but find that it is limited in several respects. Subsequently, we develop a likelihood-free inference (LFI) approach, based on Approximate Bayesian Computation (ABC) and summary statistics [7], and show that it is much more flexible than the likelihood approach. Importantly, we find that reliable inference requires consideration of many, carefully chosen network summaries simultaneously.

Having settled on a statistical method we apply the method to the two data sets and discuss the results in relation to biological knowledge and mathematical properties of the underlying model.

2. EVOLUTION OF THE INTERACTOME

Various processes contribute to the evolution of the interactome [8, 9]. The importance of gene duplication to biological evolution has long been recognized and substantial evidence that elucidate the importance and the mechanisms of this process in higher organisms has been collected from genomic sequence data, either in the form of whole genome duplication (WGD) or as single gene duplication (SGD) [10, 9]. In the two species we use here, *H. pylori* and *P. facliparum* there is no recorded evidence of WGD and we will simply ignore it in the following dis-

cussion, though we note that for other species such as *S. cerevisiea* WGDs have played an important role [11].

2.1. Single gene duplication

In most SGDs, a gene is tandemly duplicated. Just after a successful duplication, the child and the parental genes have exactly the same functions, but over a relatively short evolutionary time [10, 9], the two genes may diverge, resulting in different fates of the duplicates: i) one gene may be silenced (non-functionalization), ii) both genes are preserved such that one is redundant to the other, iii) one gene may acquire a new function while the function of the other is retained (neo-functionalization), and iv) both genes are changed through mutations and partly acquire new functions (sub-functionalization). The latter is very attractive [10] as it does not rely on sparse occurrences of benefial mutations, but on loss-of-function mutations in regulatory regions. Further, sub-functionalization is a natural mechanism for specialization of gene products to different tissues and cells. In contrast, for iii) to occur the acquisition of novel interactions through benefial mutations is required.

2.2. Attachment processes

Besides SGD (and WGD) a number of other processes contribute to the evolution of the interactome, which we collectively refer to as *attachment* processes. These include various forms of horizontal transfer of genetic material between organisms (typically bacteria), integration of viral DNA into the host genome and translocation of genetic material within an organism. All of these may lead to the formation of novel genes.

2.3. The model

We adopt a model that emphasises iv) as the most important consequence of SGD and distinguish two processes, SGD and *preferential attachment* (PA). The model is a Randomly Grown Graph (RGG) and has four parameters $\theta = (\alpha, p, q, r)$. A RGG is a Markov chain in the sense that the graph (network) $\mathcal{G}_{t+1} = (\mathcal{V}_{t+1}, \mathcal{E}_{t+1})$ at step $t+1$ only depends on the graph $\mathcal{G}_t = (\mathcal{V}_t, \mathcal{E}_t)$ at step t, where t denotes the size of the network. At step $t+1$ do:

[SGD] With probability α choose a node, $v_{\rm old}$, at random in $\mathcal{G}_t$ and introduce a new node $v_{\rm new}$. For each neighbour, v, of $v_{\rm old}$, create a link between $v_{\rm new}$ and v with probability p; otherwise with probability r erase the link $(v_{\rm old}, v)$ and create the link $(v_{\rm new}, v)$. Create a link between $v_{\rm old}$ and $v_{\rm new}$ with probability q.

[PA] With probability $1-\alpha$ choose a node, $v_{\rm old}$, with probability proportional to its degree in $\mathcal{G}_t$ and introduce a new node $v_{\rm new}$. Create a link between $v_{\rm old}$ and $v_{\rm new}$.

The model is symmetric in r in the sense that r and $1-r$ produce statistically indistinguishable networks. In our analysis we fix r to 0.5 to reduce the number of parameters. PIN data sets are incomplete and noisy in several respects; e.g. they only contain a fraction of all proteins in the interactome (incomplete sampling) and also contain both false positive and false negative links (noise). Here we only consider incomplete sampling and assume that the sampling fraction is known (estimated from the number of open reading frames in the genome).

3. STATISTICAL METHODS

We first discuss the likelihood of an entire network, and then move on to LFI.

3.1. The likelihood of a full network

Ideally, we would compute the likelihood of an oberserved PIN data set,

$$L(\theta, \mathrm{Data}) = P(\mathrm{Data}|\theta).$$

This would allow us to perform a likelihood analysis or engage in a Bayesian analysis of the posterior distribution

$$P(\theta|\mathrm{Data}) \propto P(\mathrm{Data}|\theta)p(\theta),$$

where $p(\theta)$ is a prior on θ. However, calculating $P(\mathrm{Data}|\theta)$ is computationally very demanding even for small networks.

In [5] the likelihood is calculated recursively. Denote by $\delta(\mathcal{G}_t, v)$ the graph $\mathcal{G}_t$ with the node v removed. If it is possible to go from $\delta(\mathcal{G}_t, v)$ to $\mathcal{G}_t$ by SGD or PA then we say that v is *removable* and denote the set of removable nodes by $\mathcal{R}(\mathcal{G}_t)$. Armed with this notation, the likelihood of an entire network, $\mathcal{G}_t$, takes the form

$$L(\theta, \mathcal{G}_t) = \frac{1}{t} \sum_{v \in \mathcal{R}(\mathcal{G}_t)} \omega(\theta, \mathcal{G}_t, v) L(\theta, \delta(\mathcal{G}_t, v)), \quad (1)$$

where

$$\omega(\theta, \mathcal{G}_t, v) = P(\mathcal{G}_t | \delta(\mathcal{G}_t, v), \theta)$$

is the conditional probability of generating $\mathcal{G}_t$ from $\delta(\mathcal{G}_t, v)$. The factor $1/t$ is the probability that v is the last added node and the quantity ω is a sum over all nodes that could have given rise to v by SGD or PA.

We note that the likelihood is written in a form that may facilitate approximate procedures such as Importance Sampling (IS) or MCMC [12, 13]. However, only in the cases $\alpha = 0$ and/or $r = 0, 1$ is the set of removable nodes fairly small; in all other cases the set consists of all nodes in the network [5] and it becomes computationally untractable.

Furthermore, as an additional complication, sampling is not taken into account in the recursion above, because sampling cannot be considered at each step in the recursion, and is best implemented after the network has achieved the desired size. Other approaches are therefore required.

3.2. Likelihood-free inference

To circumvent the problems with calculating the likelihood we turn to methods of ABC and LFI [7].

The basic idea in ABC is to combine Bayesian approaches with summary approaches. Rather than targeting the posterior distribution given the full data we aim at

calculating the posterior distribution given a summary of the data. This approach in addition requires to choose a reasonable set of summaries.

For a given set of summary statistics $\mathcal{S} = (S_1, \ldots, S_k)$ we adopt a MCMC scheme to simulate the posterior distribution $P(\theta|\mathcal{S})$ – now conditional on the set of summaries and not on the full PIN data. Denote by $\mathcal{S}_0$ the set of observed summary statistics. We proceed in the following way:

[A] If now at θ, propose a move to θ' according to the proposal density $q(\theta \to \theta')$

[B] Generate a network according to θ', sample the required number of nodes and calculate the summaries $\mathcal{S}'$

[C] Define $C = \prod_{i=1}^{k} \mathbf{1}(d_i(S_i', S_{i0}) < \epsilon_i)$ and calculate

$$h(\theta, \theta') = \min\left(1, \frac{p(\theta')q(\theta' \to \theta)}{p(\theta)q(\theta \to \theta')}C\right),$$

where $\epsilon_i > 0$ is a threshold and d_i a distance measure

[D] Accept θ' with probability $h(\theta, \theta')$ and otherwise stay at θ; go to [A].

Besides the summaries, we need to choose ϵ_i and d_i. For the thresholds we choose a tempering scheme such that the thresholds decrease during the burn-in period. The final thresholds are decided upon based on MCMC diagnostics (see e.g. [12]). The d_is are taken to be Euclidian.

4. STATISTICAL ANALYSIS OF PIN DATA

In this section we present results from the analyses of the *H. pylori* and the *P. falciparum* PIN data sets. Due to space limitations we are unable to present these results in full, but refer the reader to [6].

4.1. Summary statistics

Table 1 shows the effect of varying the summary statistics. In earlier papers only the degree sequence is used (see e.g. [14, 2]) and Table 1 clearly demonstrates that inference is unreliable when judged solely from the degree sequence. Interestingly, the estimate of p is much lower when based on the degree sequence only. However, as soon as several summary statistics are applied, the exact number and the particular choice of summaries become less important.

Choosing a distance measure and a precision threshold ϵ further influences the inference. As expected, credibility intervals become more narrow when smaller thresholds are applied; however this is at the cost a lower acceptance probability in the MCMC (h is lower) and additionally, burn-in occurs later in the MCMC.

4.2. *H. pylori* and *P. falciparum*

The *H. pylori* PIN data set comprises 675 proteins and 1,096 links [3]. The sampling fraction is estimated to 45% [6]. In contrast, the *P. falciparum* PIN data set is larger, comprising 1,271 proteins and 2,642 links [4]. The sampling fraction is 24% [6]. Table 2 shows the estimates of the three parameters.

	p	q	α
I	0.32 (0.09,0.69)	0.55 (0.19,0.87)	0.57 (0.24,0.87)
II	0.57 (0.44,0.75)	0.05 (0.01,0.10)	0.78 (0.64,0.92)
III	0.56 (0.44,0.79)	0.05 (0.00,0.09)	0.79 (0.64,0.93)

Table 1. Shown are the maximum posterior estimates of p, q and α, together with 80% credibility intervals for three sets of summary statistics. I) Degree Sequence (ND); II) Distribution of distances between nodes ('within reach', WR), Diameter (DIA), Cluster coefficient (CC), Average degree (AD), and size of largest connected component; III) WR, ND, CC and FRAG. The *H. pylori* data set is used.

	p	q	α
Hp	0.57 (0.44,0.75)	0.05 (0.01,0.10)	0.78 (0.64,0.92)
Pf	0.52 (0.46,0.59)	0.05 (0.00,0.09)	0.93 (0.87,0.98)

Table 2. Shown are the maximum posterior estimates of p, q and α, together with 80% credibility intervals for Hp) *H. pylori* and Pf) *P. falciparum*. Summary statistics: WR, DIA, CC, AD and FRAG.

The estimates are very similar for p and q. However, the 80% credibility intervals are wider for *H. pylori* than for *P. falciparum* which we attribute to the difference in network order – the *P. falciparum* PIN data set is almost twice as big. Intuitively, the difference in the estimates of α are biologically reasonable: *H. pylori* is a small bacterium, and bacteria are often subject to horizontal transfer of genetic material. In contrast, *P. falciparum* is a unicellular eukaryote, and attachment processes are believed to occur rarely in eukaryotes [9].

5. MATHEMATICAL INSIGHT

The Markov property of the model allow us to deduce a number of statements about the model. The expected number, $n_t(k)$, of nodes of degree k fulfills the relation

$$n_{t+1}(k) = \left(1 - \frac{1+kp}{t}\right) n_t(k) + \frac{1+(k-1)p}{t} n_t(k-1)$$

$$+2 \sum_{j \geq k-1} \binom{j}{k-1} \psi^k (1-\psi)^{j-k+1} \frac{n_t(j)}{t},$$

where $\psi = (1+p)/2$, $r = 1/2$ and $q = \alpha = 1$ (for convenience). A similar recursion can be obtained for an arbitrary set of parameters, but is more complicated. An argument for the correctness of the recursion can be found in [15, 16].

Here we are concerned with the existence of a limiting degree distribution as the network becomes large. We distinguish several different scenarios:

- If $\alpha p < 0.5$ then there exists an equilibrium distribution (ergodic recurrent solution)
- If $\alpha = 1$ and $p < 0.533...$ then an infinitely large network has infinitely many nodes of arbitrary degree, but

an equilibrium distribution is not guarenteed to exist (recurrent solution)

• If $\alpha = 1$ and $p > 0.562...$ then an infinitely large network has finitely many nodes of arbitrary degree, but potentially an infinite number of degree 0 (transient solution)

• If $\alpha < 1$ then an infinitely large network has infinitely many nodes of arbitrary degree, but an equilibrium distribution is not guarenteed to exist (recurrent solution).

Note that for $\alpha = 1$, there is a small window between 0.533 and 0.562 where we do not know what happens. The first bullet point is closely related to the average degree in the (infinitely large) network,

$$\frac{2 - 2(1-q)\alpha}{1 - 2\alpha p},$$

if $\alpha p < 0.5$ and otherwise infinity. Assuming the estimates in Table 2, both networks have a stable or an equilibrium distribution over time: For *H. pylori*, $\alpha p = 0.44$ and for *P. falciparum*, $\alpha p = 0.48$. However, in both cases αp is close to the point where we do not know whether the network stabilizes or not.

6. CONCLUSION

We have demonstated that using advanced statistical tools such as ABC or LFI it is possible to achieve inference on parameters describing the evolution of the interactomes of *H. pylori* and *P. falsiparum*. However, the matahematical models we apply are very basic and only mimic true evolution in an approximate sense. Nonetheless, the parameter estimates we find are in accordance with intuition and biological knowledge achieved by other means.

7. ACKNOWLEDGEMENTS

CW is supported by the Danish Cancer Society and the Danish Research Councils; OR is supported by the Welcome Trust, UK. Enette Berndt Knudsen is thanked for technical assistance.

8. REFERENCES

[1] M. Middendorf, E. Ziv, and C. H. Wiggins, "Inferring network mechanisms: The drosophila melanogaster protein interaction network," *Proc. Natl. Acad. Sci. USA*, vol. 102, pp. 3192 – 3197, 2005.

[2] M. P. H. Stumpf, P. J. Ingram, I. Nouvel, and C. Wiuf, "Statistical model selection methods applied to biological network data," *Lect. Notes Comput. Sc.*, vol. 3737, pp. 65 – 77, 2005.

[3] J. C. Rain, L. Selig, H. De Reuse, V. Battaglia, and et al., "The protein-protein interaction map of *helicobacter pylori*," *Nature*, vol. 409, pp. 211 – 215, 2001.

[4] D. J. LaCount, M. Vignali, R. Chettier, A. Phansalkar, and et al., "A protein interaction network of the malaria parasite *Plasmodium falciparum*," *Nature*, vol. 438, pp. 103 – 107, 2005.

[5] C. Wiuf, M. Brameier, O. Hagberg, and M. P. H. Stumpf, "A likelihood approach to analysis of network data," *Proc. Natl. Acad. Sci. USA*, vol. 103, pp. 7566 – 7570, 2006.

[6] O. Ratmann, O. Jørgensen, T. Hinkley, M. P. H. Stumpf, S. Richardson, and C. Wiuf, "Using likelihood-free inference to compare evolutionary dynamics of the protein networks of *H. pylori* and *P. falciparum*," *PLoS Comp. Biol.*, vol. 3, pp. e320, 2007.

[7] P. Marjoram and S. Tavare, "Modern computational approaches for analysing molecular-genetic variation data," *Nat. Rev. Genet.*, vol. 7, pp. 759 – 770, 2006.

[8] S. Ohno, *Evolution by Gene Duplication*, Springer Verlag, New York, 1970.

[9] M. Lynch, *The Origins of Genome Architecture*, Sinauer Press, New York, 2007.

[10] M. Lynch and J. S. Conery, "The evolutionary fate and consequences of duplicate genes," *Science*, vol. 290, pp. 1151 – 1155, 2000.

[11] B. Dujon, D. Sherman, G. Fischer, P. Durrens, and et al, "Genome evolution in yeasts," *Nature*, vol. 430, pp. 35 – 44, 2004.

[12] W. R. Gilks, S. Richardson, and D. Spiegelhalter, *Markov Chain Monte Carlo in practice*, Chapman & Hall, New York, 1995.

[13] P. J. Green, N. L. Hjort, and S. Richardson, *Highly Structured Stochastic Systems*, Oxford University Press, Oxford, 2003.

[14] A. Barabasi and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, pp. 509 – 512, 1999.

[15] O. Hagberg and C. Wiuf, "Convergence properties of the degree distribution of some growing network models," *Bull. Math. Biol.*, vol. 68, pp. 1275 – 1291, 2006.

[16] M. Knudsen and C. Wiuf, "A markov chain approach to randomly grown graphs," *J. Applied Math.*, vol. 2008, pp. 190836, 2008.

TICSP Series

1 Egiazarian, Saramäki, Astola. Proceedings of Workshop on Transforms and Filter Banks.
2 Yaroslavsky. Target Location: Accuracy, Reliability and Optimal Adaptive Filters.
3 Astola. Contributions to Workshop on Trends and Important Challenges in Signal Processing.
4 Creutzburg, Astola. Proceedings of Second International Workshop on Transforms and Filter Banks.
5 Stankovic, Moraga, Astola. Readings in Fourier Analysis on Finite Non-Abelian Groups.
6 Yaroslavsky. Advanced Image Processing Lab.: An educational and research package for Matlab.
7 Klapuri. Contributions to Technical Seminar on Content Analysis of Music and Audio.
8 Stankovic, Stankovic, Astola, Egiazarian. Fibonacci Decision Diagrams.
9 Yaroslavsky, Egiazarian, Astola. Transform Domain Image Restoration Methods: Review, Comparison and Interpretation.
10 Creutzburg, Egiazarian. Proceedings of International Workshop on Spectral Techniques and Logic Design for Future Digital Systems, SPECLOG'2000.
11 Katkovnik. Adaptive Robust Array Signal Processing for Moving Sources and Impulse Noise Environment.
12 Danielian. Regularly Varying Functions, Part I, Criteria and Representations.
13 Egiazarian, Saramäki, Astola. Proceedings of the 2001 International Workshop on Spectral Methods and Multirate Signal Processing, SMMSP2001.
14 Stankovic, Sasao, Astola. Publications in the First Twenty Years of Switching Theory and Logic Design.
15 Saramäki, Yli-Kaakinen. Design of Digital Filters and Filter Banks by Optimization: Applications.
16 Danielian. Optimization of Functionals on Classes of Distributions with Moments' Constraints, Part I, Linear Case.
17 Saramäki, Egiazarian, Astola. Proceedings of the 2002 International TICSP Workshop on Spectral Methods and Multirate Signal Processing, SMMSP2002.
18 Danielian. Optimization of Functionals on Classes of Distributions with Moments' Constraints, Part II, Nonlinear Case.
19 Katkovnik, Egiazarian, Astola. Adaptive Varying Scale Methods in Image Processing, Part I Denoising and Deblurring.
20 Huttunen, Gotchev, Vasilache. Proceedings of the 2003 Finnish Signal Processing Symposium, Finsig'03.
21 Yli-Harja, Smulevich, Aho. Proceedings of the 1st TICSP Workshop on Computational Systems Biology, WCSB 2003.
22 Saramäki, Egiazarian, Astola. Proceedings of the 2003 International TICSP Workshop on Spectral Methods and Multirate Signal Processing, SMMSP2003.
23 Sarukhanyan, Agaian, Egiazarian, Astola. Hadamard Transforms.
24 Aho, Lähdesmäki, Yli-Harja. Proceedings of the 2nd TICSP Workshop on Computational Systems Biology, WCSB 2004.
25 Astola, Egiazarian, Saramäki. Proceedings of the 2004 International TICSP Workshop on Spectral Methods and Multirate Signal Processing, SMMSP2004.
26 Yaroslavsky. Discrete Sinc Interpolation Methods and their Applications in Image Processing.
27 Astola, Danielian. Regularly Varying Skewed Distributions generated by Birth-Death Process.
28 Kulemin, Zelensky, Astola, Lukin, Egiazarian, Kurekin, Ponomarenko, Abramov, Tsymbal, Goroshko, Tarnavsky. Methods and Algorithms for Pre-processing and Classification of Multichannel Radar Remote Sensing Images.
29 Manninen, Linne, Yli-Harja. Proceedings of the 3rd TICSP Workshop on Computational Systems Biology, WCSB 2005.
30 Astola, Egiazarian, Saramäki. Proceedings of the 2005 International TICSP Workshop on Spectral Methods and Multirate Signal Processing, SMMSP2005.

31 Astola, Danielian. Frequency Distributions in Biomolecular Systems and Growing Networks
32 Ruusuvuori, Manninen, Huttunen, Linne, Yli-Harja. Proceedings of the 4th TICSP Workshop on Computational Systems Biology, WCSB 2006.
33 Lugmayr. Proceedings of The TICSP Workshop on Ambient Media and Home Entertainment at the EuroITV 2006.
34 Astola, Egiazarian, Saramäki. Proceedings of the 2006 International TICSP Workshop on Spectral Methods and Multirate Signal Processing, SMMSP2006.
35 Lugmayr, Golebiowski. Interactive TV: A Shared Experience TICSP Adjunct Proceedings of EuroITV 2007.
36 Stankovic, Astola. Reprints from the Early Days of Information Sciences, 2007.
37 Bregovic, Gotchev. Proceedings of the 2007 International TICSP Workshop on Spectral Methods and Multirate Signal Processing, SMMSP2007.
38 Grünwald, Myllymäki, Tabus, Weinberger, Yu. Festschrift in Honor of Jorma Rissanen on the Occasion of his 75th Birthday.
39 Stankovic, Astola. Gibbs Derivatives – the First Forty Years.
40 Stankovic, Astola. Reprints from the Early Days of Information Sciences, 2008.