GI-Edition

Lecture Notes in Informatics

Andreas Beyer, Michael Schroeder (Eds.)

German Conference on Bioinformatics

GCB 2008

September 9 – 12, 2008
Dresden, Germany

Proceedings

Andreas Beyer, Michael Schroeder (Eds.)

German Conference on Bioinformatics

GCB 2008

09. – 12.09.2008
in Dresden, Germany

Gesellschaft für Informatik e.V. (GI)

Lecture Notes in Informatics (LNI) - Proceedings
Series of the Gesellschaft für Informatik (GI)

Volume P-136

ISBN 987-3-88579-230-7
ISSN 1617-5468

Volume Editors
Dr. Andreas Beyer
Biotechnologisches Zentrum der TU Dresden BIOTEC
Tatzberg 47/49, 01307 Dresden, Germany
Email: andreas.beyer@biotec.tu-dresden.de
Prof. Dr. Michael Schroeder
Biotechnologisches Zentrum der TU Dresden BIOTEC
Tatzberg 47/49, 01307 Dresden, Germany
Email: michael.schroeder@biotec.tu-dresden.de

printed by Köllen Druck+Verlag GmbH, Bonn

Conference Chairs
Michael Schroeder, Biotechnology Center of the TU Dresden
Andreas Beyer, Biotechnology Center of the TU Dresden

Local Organizers
Andreas Beyer, Biotechnolgy Center of the TU Dresden
Lisa Beyer, Biotechnology Center of the TU Dresden
Andreas Deutsch, ZIH of the TU Dresden
Mandy Erlitz, Biotechnology Center oft he TU Dresden
Mandy Gläser, Biotechnology Center oft he TU Dresden
Bianca Habermann, MPI für molekulare Zellbiologie und Genetik
Robert Männel, Center for Regenerative Therapies Dresden
Michael Schroeder, Biotechnology Center of the TU Dresden
Pavel Tomancak, MPI für molekulare Zellbiologie und Genetik

Program Committee
Rolf Backofen, Freiburg
Michael R. Berthold, Konstanz
Andreas Beyer, Dresden
Sebastian Böcker, Jena
Albert Burger, Edinburgh
Thomas Dandekar, Würzburg
Werner Dubitzky, Belfast
Roland Eils, Heidelberg
Dmitrij Frishman, Munich
Georg Fuellen, Greifswald
Robert Giegerich, Bielefeld
Bianca Habermann, Dresden
Volkhard Helms, Saarbrücken
Janet Kelso, Leipzig
Ina Koch, Berlin
Jacob Köhler, Tromsø
Michael Lappe, Berlin
Ulf Leser, Berlin
Urban Liebl, Karlsruhe
Jens Meiler, Vanderbuilt
Imtraud Meyer, Vancouver
Steffen Möller, Lübeck
Burkhard Morgenstern, Göttingen
Kay Nieselt, Tübingen
Markus Porto, Darmstadt
Stefan Posch, Halle
Matthias Rarey, Hamburg
Dietrich Rebholz-Schuhmann, Hinxton
Dietmar Schomburg, Cologne
Michael Schroeder, Dresden
Stefan Schuster, Jena
Joachim Selbig, Potsdam
Peter Stadler, Leipzig
Robert Stevens, Manchester
Martin Vingron, Berlin
Thomas Wilhelm, Norwich
Ralf Zimmer, Munich

Additional Referees
Bill Andreopoulous, Florian Battke, Stephan Bernhart, Jong Bhak, Fabian Birzele, Dan Bolser, Roman Brinzanik, Benedikt Brors, Lutz Brusch, Hauke Busch, Nicolas Cebron, Matthieu Defrance, Frank Dressel, Antigoni Elefsinioti, Caroline Friedel, Irit Gat-Viks, Thasso Griebel, Stefan Haas, Ian Henry, Andreas Henschel, Ronny Herzog, Steffen Jaensch, Shobhit Jain, Yannis Kalaidzidis, Ina Koch, Inke Koenig, Rainer König, Michael Kücken, Robert Küffner, Dirk Labudde, Phillip Lord, Thomas Manke, Annalisa Marsico, Roxana Merino Martinez, Sven Mesecke, Jacob Michaelson, Ralf Mikut, Burkhard Morgenstern, Uwe Ohler, Anton Pervukhin, Tobias Petri, Sebastian Pfeiffer, Conrad Plake, Stephan Preibisch, Sathyapriya Rajagopal, Matthias Reimann, Peter Robinson, Loic Royer, Stephan Saalfeld, Gunnar Schramm, Marcel Schulz, Stefan Schuster, Joachim Selbig, Juergen Suehnel, Stephan Symons, Till Tantau, Andrea Tanzer, Anke Tru, Anne Tuukkanen, Boris Vassilev, Anja Voss-Boehme, Lukas Windghager, Rainer Winnenburg, Christof Winter, Tomasz Zemojtel, Shobhit Jain, Vineeth Surendranath

Sponsors of GCB2008

Scientific societies

Non-profit societies

Commercial sponsors

Preface

This volume contains papers presented at the German Conference on Bioinformatics, GCB 2008, held in Dresden, Germany, September 9-12, 2008 at the Deutsches Hygiene Museum Dresden.

GCB is an annual, international conference, which provides a forum for the presentation of current research in bioinformatics and computational biology. It is organized on behalf of the Special Interest Group on Informatics in Biology of the German Society of Computer Science (GI) in cooperation with the German Society of Chemical Technique and Biotechnology (Dechema) and the German Society for Biochemistry and Molecular Biology (GBM) with support of the European Life Science Organization.

GCB2008 comprises six invited talks by Michael Ashburner, Janusz Bujnicki, David Gilbert, Trey Ideker, Jens Reich and Marino Zerial. The talk by Jens Reich on a person's dignity in the age of the genome chip was co-organised by the Deutsches Hygiene Museum Dresden and GCB. It was held in German and open to the general public. GCB also featured four tutorials by Jens Meiler (Rosetta in computational structural biology), Steffen Möller (expression QTL and their analysis), Johannes Schindelin (image analysis), and Stefan Schuster (metabolic pathway analysis).

GCB received 62 submissions for regular papers, which were reviewed by the PC and additional reviewers. After reviewing, 19 were accepted for publication in this volume (30% acceptance rate).

Thanks to the programme committee members and reviewers, to the local organizers, and to the sponsors.

Dresden, July 2008

Andreas Beyer and Michael Schroeder

Table of Contents

Consecutive KEGG pathway models for the interpretation of high-throughput genomics data

Alexey V. Antonov[1*], Sabine Dietmann[1], Hans W. Mewes[1,2]

[1] Helmholtz Center Munich, Institute for Bioinformatics and Systems Biology, Ingolstädter Landstraße 1, D-85764 Neuherberg, Germany

[2]Department of Genome-Oriented Bioinformatics, Wissenschaftszentrum Weihenstephan, Technische Universität München, 85350 Freising, Germany

[*]Corresponding author.

Abstract: A common strategy to deal with the interpretation of gene lists is to look for overrepresentation of Gene Ontology (GO) terms or pathways. In related computational approaches the cell is formalized as genes that are grouped into functional categories. As output, a list of interesting biological processes is provided, which seems to be mostly covered by the supplied gene list. However, it is more natural to model the cell as a network that reflects relations between genes. For many biological processes such information is available, but it is not used to the full extent in interpretational analyses. In this paper, we propose to interpret gene lists in network terms to provide the most probable scenario of gene interactions based on the available information about the topology of metabolic pathways. The proposed approach is an effort to exploit the biological information available in public resources to a greater extent in comparison to the existing techniques. Applying our approach to experimental data, we demonstrate that the currently widely employed strategy produces an incomplete interpretation, whilst our procedure provides deeper insights into possible molecular mechanisms behind the experimental data.

1. Introduction

In the post-genomic era the targets of many experimental studies are complex cell disorders. A standard experimental strategy is to compare the genetic signatures of the cells in normal and anomalous states. As a result, a set of genes, whose measured activity differs between considered cell states, is delivered. In the next step, an interpretation of the identified genes is required. A common bioinformatics strategy is to infer biological processes that are most relevant to the analyzed gene list. The inference is based on a prior knowledge about individual gene properties, such a molecular functions or biological processes.

In the standard bioinformatics framework, the cell is modeled as a set of genes that splits into known functional categories (Khatri et al., 2007; Antonov and Mewes, 2006; Khatri and Draghici, 2005; Subramanian et al., 2005; Berriz et al., 2003; Khatri et al., 2002). We will refer to this approach as "categorical". It is obvious, that this approach has a number of shortcomings. First, the categorical approach discards from consideration a lot of valuable information that is available in public databases. The relations between genes inside each category, like the pathway topology, as well as the relations between genes from different categories are not considered within the standard categorical framework. Second, the output of the categorical approach is a list of categories that are overrepresented among the analyzed genes. This is evidently helpful information; however, in most cases, this is not exactly what experimentalists are looking for. A basic premise in the application of high-throughput methodologies for studying molecular mechanisms of complex cell disorders is cooperative gene behavior. The change in the state of one (or

several) gene(s) leads to cooperative changes in the state of several dependent genes, and so on. Ideally, as an interpretational model of the gene list, the experimentalist would prefer to obtain a network model that proposes the most probable scenario of gene relations, which cover most of the genes from the supplied experimental list, i.e. gene A interacts with gene B, gene B interacts with gene C, gene C interacts with gene D, and genes A, B, C are from the same metabolic pathway, while genes C and D are regulatory genes. Thus, one seeks not only the information that the corresponding metabolic and regulatory pathways are enriched within a gene list, but also the way, in which the genes are interacting between and within the pathways.

Efforts have been made to overcome the first limitation of the categorical approach in order to take into account the pathway topology. Rahnenfuhrer et al., 2004 used, in addition to pathway categories, the distance between genes within the metabolic pathway. In this case, the impact of a pair of genes was weighted with respect to the distance between genes within a metabolic pathway. Another procedure, proposed recently by Draghici et al., 2007, exploited the hierarchical structure of signaling pathways and weighted the impact of genes with respect to their position in a pathway hierarchy. Genes at the top of signaling cascade received higher impacts in comparison to downstream genes. However, in both cases the second limitation has not been overcome, i.e. both approaches did not provide significant relationships between genes from different pathways. The output was still a list of categories that were enriched.

We propose a fundamentally different technique for the analysis of gene list referred to as the network-based approach (vs. categorical). The cell is modeled as set of genes that are connected into a global network. The input gene list is translated into a network model according to the global network, which reflects the most probable scenario of how genes affect the state of each other. As output, along with a list of enriched categories, our procedure provides a model of gene interactions that present a description of how different and apparently independent biological processes are interconnected. The statistical significance of the inferred network model is computed by a random simulation procedure. We demonstrate on several experimental data sets that our approach provides deeper insight into biological mechanisms that unites the supplied gene lists in comparison to currently available methods.

2. Network based approach

In general, an enrichment analysis is based on the available information about individual gene properties. In most cases, the experimental knowledge is formalized in a categorical format, as provided by several functional classification schemes (Mewes et al., 2004; Apweiler et al., 2001; Ashburner et al., 2000). Genes are subsequently assigned to the pre-defined classes. A straightforward way to use this information is to select those categories that have a statistically significant intersection with the analyzed gene list.

In some cases, like for metabolic processes, the experimental knowledge is stored in more complex forms to represent, for example, associations between genes and metabolites. This information can be easily converted into a pairwise distance between genes, and can be used to infer the optimal network model from a gene list. The distance between genes can be counted as the minimal number of consecutive steps required to get from one gene to another by working through existing paths on the global metabolic network. The inferred network model has several statistical properties which reflect the closeness of connected genes in the network. Based on the distribution of these properties for random gene lists one can estimate the statistical significance of the inferred model.

We used KEGG reference maps (Ogata et al., 1999) of metabolic pathways to generate pairwise distances between available genes. For each gene the set of associated compounds was defined. Genes and compounds are considered to be associated if they are assigned to the same reaction, e.g. a compound is either a substrate or product of the reaction and the gene is mapped to the enzyme that catalyses the reaction. In the same way, for each compound the set of associated genes is defined. A pair of genes is referred to as neighbors, if they have at least one common associated compound. While connecting neighbors via edges, we generate a global Gene Association Network (GAN) by integrating all available metabolic pathways. The distance between neighbors is set to “1” (one step to get from one gene to

another). The distance between two arbitrary genes is computed as a minimal number of steps required to get from one gene to another through available paths on the GAN.

Given a gene list, our purpose is to infer the network that minimizes the distance between each connected gene pair according to the GAN. To solve this problem, we propose to infer the network by a simple iterative procedure. In the first step, we connect by edges all gene pairs with distance 1. In the second step, isolated genes with distance 2 are connected. Genes are referred to as isolated, if there is no path in the network that connects them. Otherwise genes are referred to as connected. In the third step, isolated genes with distance 3 are connected. From our experience with experimental data a distance larger than 3 indicates that the statistical significance of the edge is low and that the genes can be considered independent. At each step (1, 2, 3) we look for connected sub-networks and identify the one with the maximal size (number of nodes or edges). The sub-network is referred to as connected, if it has only connected genes. The sub-network with maximal size is referred to as a maximal sub-network. We also refer to the size of maximal sub-network as the size of the inferred model. The model size is considered as a statistics, which is used to estimate the statistical significance of the model.

The statistical significance of the inferred model is estimated based on the distribution of the model size derived from random gene lists. The distribution is computed by a random simulation procedure (Westfall and Young, 1993). In the first step the random gene list of the size equal to the size of the input list is generated. The iterative network inference procedure described above is applied to the generated gene list. At each step (1, 2, 3) the size of the maximal sub-network is determined. By repeating the random procedure k times we get the background distribution for model size of random gene list and can estimate the statistical significance of the inferred network model up to the confidence level 1/k.

In addition to the genes from the supplied list, the inferred network model includes intermediate metabolites and genes. Intermediate genes are genes that, according to inferred model, connect genes from the list. If the distance between two genes from the list is 2 than they are not neighbors and connected via intermediate gene. Each pair of gene neighbors has a common metabolite (or several metabolites) used to connect them.

Known metabolic genes represent only 10 to 40 percent of genes from the whole genome (depending on the organism analyzed). For other genes there is no reliable information available about their network associations. Therefore, we propose to combine both approaches. Those genes from the analyzed list that are mapped to KEGG pathways are assessed by the network approach. In addition, standard enrichment analysis of GO categories (Ashburner et al., 2000) is performed. In the last step, both models are united in a final graphical representation. Significantly enriched GO terms that additionally have a statistically significant overlap with genes from the network model are selected and added to represent relationships between metabolic and other biological processes.

3. Results

We present several examples of data analyses by the network approach. We start with a simple illustrative example to demonstrate the advantages of the network approach in comparison to the categorical one. In the next step, we bring together two independent studies that performed experimental analyses to identify over- or underrepresented genes related to different biological problems. In each case, we collect the set of differentially expressed genes originally identified in each study and reanalyze them by the network approach.

3.1 Artificial data example

Let us consider an illustrative example to highlight the advantages of the network approach. Assume that as a result of some experiment one gets a list of nine metabolism-related genes, namely *ME3, MDH1, FH, ASL, ASS1, CTH, CDO1, CBS,* and *SHMT1*. Standard enrichment analysis will report several metabolic pathways as being enriched. Three genes (*CTH, SHTM1, CBS*) are mapped to "*glycine, serine and*

threonine metabolism". Two genes (*ASL, ASS1*) are mapped to "*urea cycle*" and two genes (*ME3, MDH1*) are mapped to "*citrate cycle*". No functional model that unites all 9 genes together would be supplied by any currently available analytical tool or approach.

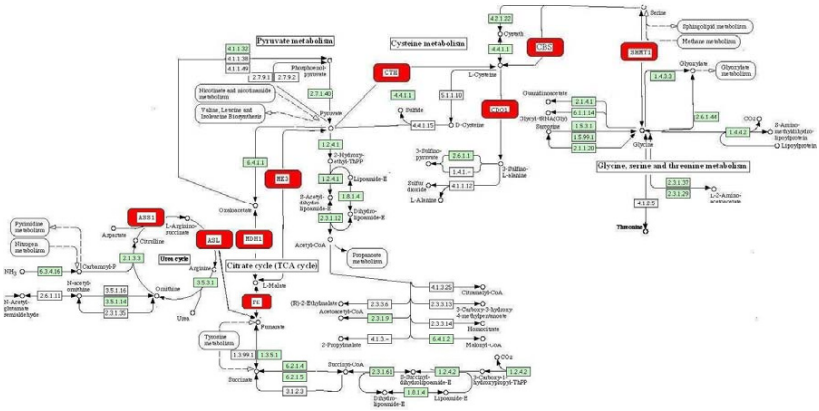


Figure 1: The genes *ME3, MDH1, FH, ASL, ASS1, CTH, CDO1, CBS, SHMT1* are presented as red boxes. Five KEGG pathway wiring diagrams ("*urea cycle*", "*citrate cycle*", "*pyruvate metabolism*", "*cysteine metabolism*", "*glycine, serine and threonine metabolism*") are linked together to demonstrate that all 9 genes are located on a consecutive metabolic path.

However, according to the KEGG pathway wiring diagrams, all 9 genes are consecutively connected via metabolites (Figure 1) and form a non-interrupted path, which runs through five canonical metabolic pathways ("*urea cycle*", "*citrate cycle*", "*pyruvate metabolism*", "*cysteine metabolism*", and "*glycine, serine and threonine metabolism*"). This illustrative example demonstrates that in many cases the knowledge of enriched individual pathways may be insufficient to get a complete understanding of the relation among genes from the supplied list. The consideration of the global gene metabolic network to interpret gene list as a network may be much more informative.

3.2. Analysis of long-lived *C. elegans daf-2* mutants using serial analysis of gene expression

Halaschek-Wiener *et al*., 2005 identified genes that are associated with longevity in a long-lived Caenorhabditis elegans daf-2 (insulin/IGF receptor) mutant using serial analysis of gene expression (SAGE). SAGE libraries were prepared from daf-2 worms at days 1, 6, and 10 of adulthood. The day 6 library represents gene expression in mid-adult life, whereas day 10 marks the latest time before the occurrence of dead animals in the population. To identify gene expression differences and metabolic changes that may lead to the increased life expectancy of *daf-2* adults, the *daf-2* and control worms at the same chronological age at day 6 were analyzed. SAGE libraries were screened for tags that had an abundance of at least 10 in one of the libraries and were differentially expressed by > 2.5-fold between *daf-2* and controls, with a P-value < 0.05. The number of selected genes was about 250 (Halaschek-Wiener *et al.*, 2003, Supplementary Data).

A standard enrichment analysis provides several GO terms that are overrepresented among the analyzed genes. Some terms were related to development and regulatory processes. Among the interesting biological processes, which may have direct links to molecular mechanisms that underlie longevity, one should

mention "*embryonic development ending in birth or egg hatching*", "*lipid transport*", and "*larval development*". Seventeen differentially expressed genes map to KEGG metabolic pathways. However, only the "*glycolysis pathway*" was identified as enriched (P-value ~ 0.05), 4 genes (*F01F1.12, GPD-4, T03F1.3* and *GPD-1*) out of 24 pathway-related genes were among those that were differentially expressed. Other 13 metabolism-related genes were not interpreted, as they represent a statistically insignificant share of genes from pathways they belong to.

In contrast, the application of our network approach reveals that 15 (out of 17) metabolism-related genes are connected into a network model with a distance between each gene pair not exceeding 2 (each pair of genes connected in the network is separated by a maximum of two metabolites). The network model runs through several canonical metabolic pathways, as presented in the overall graphical model in Figure 2. In addition, 6 genes from the inferred network model were also annotated as *"embryonic development ending in birth or egg hatching",* the GO term that was enriched among the 250 differentially expressed genes.

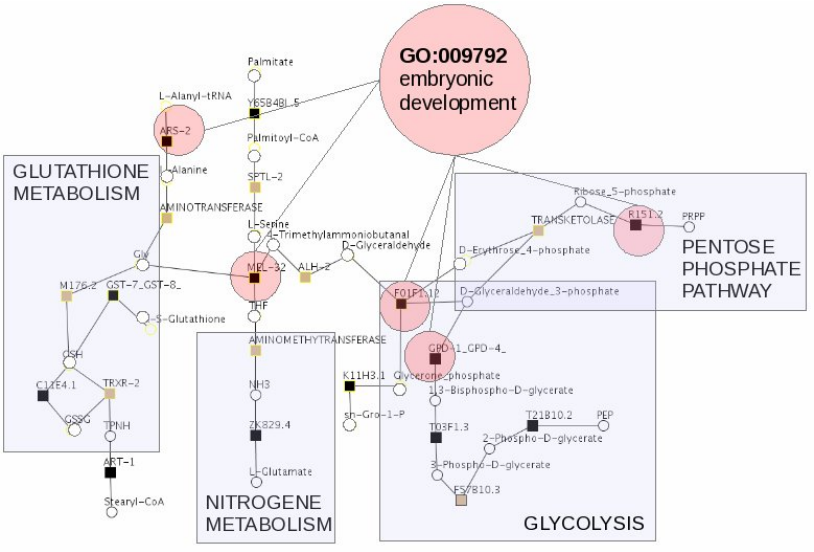


Figure 2: Network model of longevity-associated genes (Halaschek-Wiener et al., 2005) spanning four KEGG pathways. Differentially expressed genes are indicated by black rectangles, intermediate genes in the network model by brown rectangles, and chemical compounds by white circles. Genes that are involved in embryonic development (GO term: GO:009792) are highlighted by circles.

In total, 797 *C. elegans* genes can be mapped to KEGG pathways; and 154 of them are annotated with the GO term GO:009792 ("*embryonic development*"). Our network model covers 15 genes, and 6 are annotated with the term GO:009792. Based on the incidence of genes from the GO category GO:009792 among the KEGG genes (0.2 ~ 154/797) and the same incidence among network model genes (0.4 ~ 6/15), we can propose that the inferred network model has an overrepresentation of genes annotated as *"embryonic development ending in birth or egg hatching"* (P-value ~ 0.03, hypergeometric test). However, we would like to point out, that the correct estimation of the statistical significance in this case requires a non-trivial model which is beyond the scope of this paper.

The interpretational model supplied by the network approach is apparently more instructive in comparison to the categorical approach. The graphical representation of the inferred network allows one to track naturally the relation between metabolic and developmental processes.

To validate the statistical significance of the inferred network model, we computed a background distribution by a random simulation procedure. As described in the previous section, we generated randomly 1000 times the set of 17 genes from the set of C. elegans genes, which mapped to KEGG metabolic pathways. Each time we applied two steps of the proposed network inference procedure to the random set. As a result, all gene pairs from the randomly generated list with a distance equal to 2 (genes in the network model connected via 1 intermediate gene) and 1 (gene that relate to a common metabolite) were connected. Each time we computed the size (number of nodes) of the maximally connected sub-network. We considered these 1000 values as the background distribution for estimating the significance of the inferred network model. In total, 6 times the size of the network model inferred from the randomly generated gene list was greater or equal to 15. Therefore, the P-value of the inferred network model estimated by the random simulation procedure was less than 0.01 (6/1000). Figure 3 presents a plot of the generated background distribution.

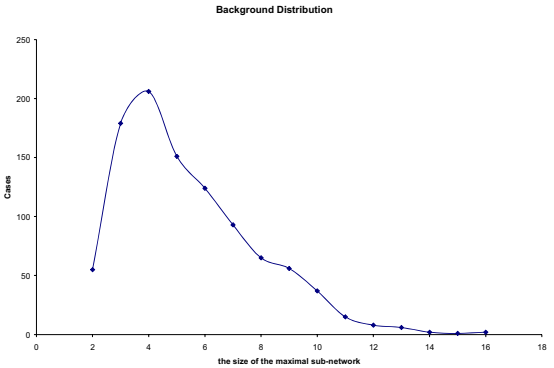


Figure 3: The distribution of the network model size generated by random simulation. The size (x-axis) of the network model is plotted against the number of times (y-axis) the network with this size was inferred during 1000 random simulations.

3.3. Pathway analysis of kidney cancer using proteomics and metabolic profiling

Perroud et al., 2006 performed a proteomic analysis of tumors to determine which pathways and processes are likely to be operative in renal cell carcinoma (RCC). By using 2-dimensional electrophoresis and mass spectrometric analysis, 31 proteins were identified to be differentially expressed in clear cell RCC as compared to adjacent non-malignant tissue. The standard categorical approach applied by the authors identified groups of genes and proteins which are organized into metabolic and signaling pathways relevant to the oncogenesis or progression of ccRCC. Several metabolic pathways closely associated with gluconeogenesis, such as "*pyruvate metabolism*", "*pentanoate metabolism*", "*butanoate metabolism*", as well as "*arginine and proline metabolism*" and the "*urea cycle*", were reported to be enriched among down-regulated genes in ccRCC. Similarly, the glycolysis pathway was identified as being significantly altered in ccRCC. In addition, a statistically significant alteration of the non-metabolic p53 signaling pathway was identified.

The authors of the paper suspect that the identified proteins from different enriched metabolic pathways are dependent. Indeed, 15 out of 16 proteins that were mapped to KEGG pathways form a statistically significant network model (P-value < 0.001). The inferred network contains 19 edges, 6 edges of length 1 and 13 edges of length 2. The models provided by the network approach are evidently more informative in comparison to the categorical one. For example, the authors report that 6 proteins (*HSP1, PKM2, GAPDH, LDHA, ANXA4 and ANXA5*) participate in the p53 signaling pathway. Three of these proteins are involved in the inferred network model. Using visualization capabilities of the network approach we can get an idea of how metabolic and signaling processes are linked in the altered cancer cells. Figure 4 presents a graphical visualization of the inferred model.

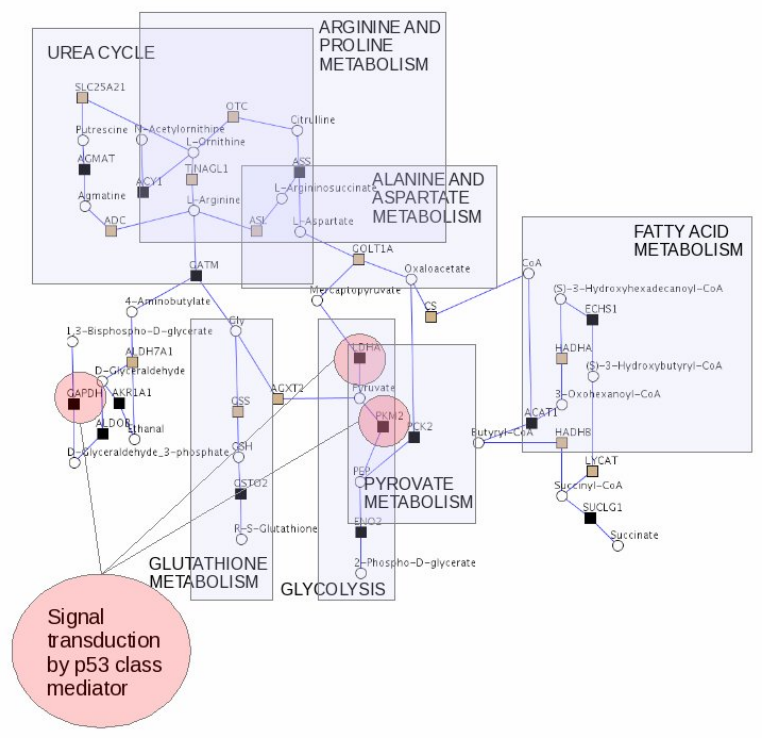


Figure 4: Network models of genes indicating a high risk of kidney cancer (Perroud et al., 2006). Differentially expressed genes are shown by black rectangles, intermediate genes by brown rectangles, and chemical compounds by white circles. Genes that are known to be involved in p53-mediated signaling are highlighted by circles.

The graphical representation of the inferred network is amenable to further analyses. In particular, it may be useful for decisions regarding potential therapy. Brief analysis of the network in Figure 4 identifies that it naturally splits into several sub-networks. Each of these sub-networks connects to the other nodes by single or double paths. For example, the sub-network, which is mostly related to fatty acid metabolism, is connected to the remaining nodes via the path ACAT1 – CoA – CS -- Oxaloacetate. Disruption of this path

may normalize fatty acid metabolism in cancer cells and, thus, reduce the potential of cancer cells to multiply. In the same way, targets for normalizing other metabolic processes may be selected. For example, to affect the urea cycle metabolism in cancer cells, at least two paths must be interrupted, i.e. the path ASS -- Laspartate -- COLT1A - Oxaloacetate and the path Larginine -- GATM.

4. Discussion

The importance of the development of network strategies for the analysis of biological systems was stressed in many studies (Lu et al., 2007; Chuang et al., 2007; Loscalzo et al., 2007; Ergun et al., 2007) . Here, we presented a network strategy for interpretational modeling of results of high-throughput genomics data. We demonstrated that the proposed procedure for translating gene lists into gene network models has a number of advantages in comparison to the widely used categorical approach. First, the coverage of the network model is higher in comparison to the categorical approach. As demonstrated, the network model usually covers a large fraction of genes that are mapped to metabolic pathways. For example, in the first case of C. elegans *daf-2* mutants among 250 selected genes 17 were mapped to metabolic pathways. The standard categorical approach was able to identify the enrichment model for only 4 of them from the glycolysis pathway. However, the network approach infers statistically valid model demonstrating that 15 genes are involved in close metabolic relation. Second, the output network model provides detailed information on pairwise gene relations among the analyzed genes. In the categorical approach this information is limited to the size of individual pathway.

At this stage, we used only metabolic pathway data for interpretational network modeling because this is one of the most reliable resources of genomics data available in network format for most model organisms. To take into account biological process other than metabolic, we combined the network approach with the standard categorical procedure. This allows for generating statistically valid hypotheses of how changes of metabolic processes interact with other non-metabolic biological processes mostly affected in the studied phenomena. However, there are no principle limitations to expand the network approach to comprise gene regulatory networks and protein interaction data of various natures. We consider extending the network approach with this kind of data in the nearest future.

References

Antonov,A.V. and Mewes,H.W. (2006) Complex functionality of gene groups identified from high-throughput data. *J. Mol. Biol.*, 363, 289-296.

Apweiler,R. et al. (2001) The InterPro database, an integrated documentation resource for protein families, domains and functional sites. *Nucleic Acids Res.*, 29, 37-40.

Ashburner,M. et al. (2000) Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat. Genet.*, 25, 25-29.

Berriz,G.F. et al. (2003) Characterizing gene sets with FuncAssociate. *Bioinformatics.*, 19, 2502-2504.

Chuang,H.Y. et al. (2007) Network-based classification of breast cancer metastasis. *Mol. Syst. Biol.*, 3, 140.

Draghici,S. et al. (2007) A systems biology approach for pathway level analysis. *Genome Res.*, 17, 1537-1545.

Ergun,A. et al. (2007) A network biology approach to prostate cancer. *Mol. Syst. Biol.*, 3, 82.

Halaschek-Wiener,J. et al. (2005) Analysis of long-lived C. elegans daf-2 mutants using serial analysis of gene expression. *Genome Res.*, 15, 603-615.

Khatri,P. and Draghici,S. (2005) Ontological analysis of gene expression data: current tools, limitations, and open problems. *Bioinformatics.*, 21, 3587-3595.

Khatri,P. et al. (2002) Profiling gene expression using onto-express. *Genomics*, 79, 266-270.

Khatri,P. et al. (2007) Onto-Tools: new additions and improvements in 2006. *Nucleic Acids Res.*, 35, W206-W211.

Loscalzo,J., Kohane,I. and Barabasi,A.L. (2007) Human disease classification in the postgenomic era: a complex systems approach to human pathobiology. *Mol. Syst. Biol.*, 3, 124.

Lu,X. et al. (2007) Hubs in biological interaction networks exhibit low changes in expression in experimental asthma. *Mol. Syst. Biol.*, 3, 98.

Mewes,H.W. et al. (2004) MIPS: analysis and annotation of proteins from whole genomes. *Nucleic Acids Res.*, 32, D41-D44.

Ogata,H. et al. (1999) KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Res.*, 27, 29-34.

Perroud,B. et al. (2006) Pathway analysis of kidney cancer using proteomics and metabolic profiling. *Mol. Cancer*, 5, 64.

Rahnenfuhrer,J. et al. (2004) Calculating the statistical significance of changes in pathway activity from gene expression data. *Stat. Appl. Genet. Mol. Biol.*, 3, Article16.

Subramanian,A. et al. (2005) Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proc. Natl. Acad. Sci. U. S. A*, 102, 15545-15550.

Westfall,P.N. and Young,S.S. (1993) Resampling-Based Multiple Testing: Examples and Methods for p-Value adjustment. John Wiley&Sons, In.,New York.

ResqMi - a versatile algorithm and software for Resequencing Microarrays

Stephan Symons [a], Kirstin Weber [b], Michael Bonin [c], and Kay Nieselt [a*]

[a] Center for Bioinformatics Tübingen, University of Tübingen
Sand 14, 72076 Tübingen, Germany
[b] Kocherstr. 8, 72768 Reutlingen-Altenburg, Germany
[c] Medical Faculty, Microarray Facility, University of Tübingen
Calwer Str. 7, 72076 Tübingen, Germany

Abstract: Resequencing microarrays are a common tool for fast monitoring of individual genetic variations. Applications include diagnosis of genetic and infectious diseases and SNP prediction. Base calling is the crucial step in the analysis of resequencing data. All current base calling algorithms produce ambiguous calls on parts of the sequence. Therefore, proper data handling, editing and visualization as well as revised calling algorithms are generally necessary for successful data interpretation. We present a base calling algorithm that uses a model-based approach using intensity comparisons and region-wise conformance assessment, as well as an algorithm to revise uncalled positions. The calling algorithm is shown to have call rates comparable to ABACUS, the currently most commonly used method. Both algorithms combined however can considerably increase the calling rate. We also present a new open source software called ResqMi, short for *Reseq*uencing using *Mi*croarrays, which focuses on the efficient and user-friendly analysis, visual inspection and easy manual editing of resequencing microarray data. Both algorithms are implemented as plugins for ResqMi. ResqMi is available at http://www-ps.informatik.uni-tuebingen.de/resqmi

1 Introduction

Microarrays have become a major tool for various analytical purposes, including sequence analysis. Resequencing microarrays are now commonly used for the fast and precise analysis of individual genetic variations. A common application is the identification of genetic diseases by resequencing the respective genes. This allows a faster and more reliable diagnosis than traditional methods and often directly displays the cause of the disease. Furthermore resequencing has been used to monitor genetic variation of infectious diseases [S$^+$06]. Resequencing arrays have been used to analyze genes prone to carry malignant mutations, which might not directly cause a disease, but raise the risk for e.g. cancer. Here, a timely diagnosis allows to delay or avoid the outbreak of the disease. Analysis of mitochondrial mutations is informative for a variety of applications from dis-

*to whom correspondence should be addressed

ease genetics to forensic identification. Affymetrix's GeneChip® Human Mitochondrial Resequencing Array 2.0 interrogating the entire 16kb mitochondrial genome on a single array has been used for the detection and diagnosis of various diseases [M$^+$07, v$^+$06]. In differential diagnostics of infectious diseases, resequencing allows to precisely identify the infectious pathogen [L$^+$06, M$^+$06, W$^+$06]. In another application, resequencing arrays have been used to identify possible antibiotic resistances [D$^+$05]. NimbleGen's CGS platform [A$^+$05] has been applied to similar purposes, especially on a whole microbial genome scale [J$^+$08]. Although the length of the sequence that can be (re)sequenced with one array is limited, the same technology is applied for whole-genome SNP (Single Nucleotide Polymorphism) analysis studies using sets of several arrays [H$^+$05, C$^+$07].

Resequencing microarrays work in a sequencing by hybridization scheme [BS88, LF94]. Oligonucleotide probes, typically of length 25, are synthesized using an array tiling strategy with eight unique probes per target base position. Each oligo probe is varied at the central position to incorporate each possible nucleotide - A, G, C, or T - allowing for the detection of both known and novel SNPs.

Given a successfully hybridized and scanned resequencing microarray, it is necessary to derive the nucleotide sequence from the spot intensities. This process is known as base calling. A naïve base calling scheme would simply call that base corresponding to the highest intensity at the specific position. More elaborate algorithms have been developed. The *Adaptive Background genotype Calling Scheme* (ABACUS) algorithm [C$^+$01], employs a series of data integrity checks to filter out sites of poor quality and uses a likelihood-based method for base calling. Model-P uses a physical model based on the sequence of the oligo probe and the target to obtain feature intensities for different potential genotypes [ZK05]. Clark *et al.* have proposed a model-based algorithm using intensities and neighborhood-related features [C$^+$07].

All base calling algorithms have in common that a number of ambiguous calls remain, so that manual inspection of data is required. No-call ratios as low as 5% [Aff06] leave several hundred or more bases per experiment to inspect by the user in order not to miss an important mutation. Manual inspection and subsequent editing of such large datasets is generally cumbersome and time consuming. Furthermore, GSEQ (GeneChip® Sequence Analysis Software, Affymetrix), the only currently available software applications for Affymetrix resequencing arrays, lacks important visualization features, base-editing ability and has restrictive operating system requirements.

Any software for the analysis of resequencing microarrays should satisfy a few criteria. First, it should allow efficient and fast processing of the arrays. In particular base calling should be automatic and leave the least possible number of ambiguous base calls for subsequent manual inspection. Second, user-friendly interaction as well as swift navigation through sequence and intensity data is necessary to find and identify the impact of mutations. Finally, the software should provide an overview and position specific visualization of intensity as well as sequence data, which is important for the inspection of critical positions and for manual base calls.

In this paper we present ResqMi, a new open source software and framework for the analysis of resequencing microarray data. In ResqMi we have implemented an efficient base

calling algorithm that produces easily interpretable base calls. Furthermore, we present an additional algorithm to enhance the call rates. Applications of the algorithms to three different resequencing experiments shows that the model-based approach is much faster than ABACUS, and, in combination with the second calling algorithm it produces higher call rates in most cases. ResqMi features visualization of intensity and sequence data and facilities to revise problematic calls. ResqMi has a user-friendly GUI and can easily be expanded with further functionality via a plugin interface.

2 Methods

We focus on resequencing data derived using the Affymetrix GeneChip® Sequence Analysis platform [Aff04, M$^+$04]. It allows resequencing of genomic DNA using custom made oligonucleotide arrays. Oligonucleotides of length 25 are used with the interrogation base at position 13. The sequence analyzed with CustomSeq arrays is split into fragments (with lengths ranging from 6 to several 10,000 positions). Each fragment represents for example an exon or a region of particular interest (e.g. known spots of variation). Fragments are not necessarily genomically adjacent. Therefore, when investigating windows around particular positions, only positions within the same fragment are considered. Our base calling algorithm uses a model-based approach as described in [C$^+$07]. The underlying idea is that bases are called according to a combination of position-wise intensity comparisons as well as a region-wise conformance assessment (see figure 1 for an overview).

Let $I^s_{x,i}$ be the intensity of base $x \in \{a, c, g, t\}$ in tiling position i on the sense strand and let $I^a_{x,i}$ be the intensity of base x in tiling position i on the antisense strand. Let R_i be equal to the base at the ith position of the reference DNA. Let R^c_i be its respective complementary base.

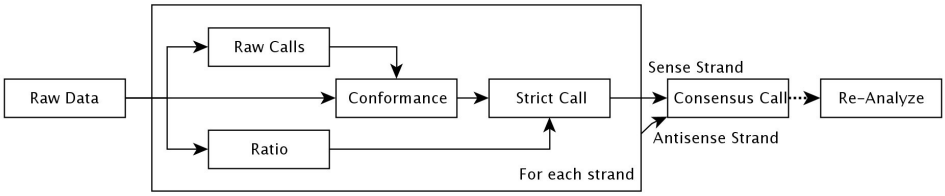


Figure 1: Overview of base calling algorithms in ResqMi. The figure shows the major steps in the model-based base calling algorithm. The dashed arrow refers to optional steps that the user can perform after base calling.

1. Raw Call: For each position i on the sense strand set $B^s_i = \arg\max_x \left\{I^s_{x,i}\right\}$. Similarly, for each position i on the antisense strand set $B^a_i = \arg\max_x \left\{I^a_{x,i}\right\}$.

2. Conformance: C^s_i is defined as the fraction of raw base calls on the sense strand which are identical to the bases of the reference DNA within a sliding window. If $B^s_i = R_i$, the window ranges from $i - 10 \ldots i + 10$, else the range is set to

$i-20 \ldots i-10$ and $i+10 \ldots i+20$. Likewise, calculate C_i^a for the antisense strand.

3. Ratio: At position i let P_i^s be the highest intensity and let Q_i^s be the second highest intensity on the sense strand. Then we define $\Delta_i^s = \frac{P_i^s}{Q_i^s}$. For the antisense strand, calculate Δ_i^a likewise.

4. Strict Call: Let μ be a threshold parameter for the conformance, and let ν be a threshold parameter for the ratio. A strict call $S_i^s = B_i^s$ is made if $C_i^s > \mu$ and $\Delta_i^s > \nu$. If either C_i or Δ_i is below the respective threshold, no reliable base call is possible and $S_i^s = \mathtt{n}$ is called. Furthermore,$S_i^s = \mathtt{n}$ is set if $B_i^s \neq R_i$ and if there is an alternative call within a region $i-5 \ldots i+5$ with a higher intensity than at position i. Analogously, strict calls S_i^a are produced for the antisense strand.

5. Consensus Call: In this last step the strict calls of the sense and antisense strand are compared. If S_i^s is complementary to S_i^a and if there is no alternative call within $i-5 \ldots i+5$ with a higher intensity, the respective base is returned. If the strict calls in both strands differ, we have implemented two possibilities for the call. The strict consensus call returns $\mathtt{n}$. The relaxed consensus call sets the resulting base to its IUPAC code. For example, if $S_i^s = \mathtt{a}$ and $S_i^a = \mathtt{c}$, the resulting relaxed consensus call is $\mathtt{r}$.

The rationale for this algorithm is based on the following natural approach: A call is made, unless the position has poor quality. A position has poor quality if the signal is ambiguous, i.e. either the difference between the highest and second highest intensity at the respective position is very small (low Δ) or the conformance of base calls with the reference sequence within a region around the position is low. If at a specific position the corresponding call differs from the reference base, the intensities at the adjacent positions are reduced, because they each have a mismatch position. Therefore, any signal at these positions may be due to unspecific hybridization [Hac99]. While this tends to lower the conformance around non-reference calls, at least the flanking regions are required to have a high conformance. For the same reason, in the strict and consensus call step, no brighter alternative calls are allowed within an interval around a position, for the brightest alternative is more likely to be actually based on a reliable signal. Using these two main criteria for quality, the meaning and impact of the two threshold parameters μ and ν are intuitive and the results are easily interpretable. If μ and ν are high, only regions with large consistency with the reference sequence and large Δ will be unambiguously called.

All current calling algorithms fail to call bases that in fact can be unambiguously assigned to a specific base when manually inspecting the site. These are bases that are clearly homozygous. In this case on both strands the brightest and the next-brightest intensity have a reasonable distance above a certain threshold, and the highest signal of the sense strand and the highest signal of the antisense strand refer to complementary bases. Therefore we have devised a simple scheme, called Re-Analyze, to resolve such calls automatically. Re-Analyze is applied only to positions that were previously called as $\mathtt{n}$. Formally, let P^s and P^a be the highest, Q^s and Q^a the next highest intensities in the sense and antisense strands, respectively. If $Q^s < \nu P^s$ and $Q^a < \nu P^a$ and P^s and P^a correspond to complementary

bases, call this base, else the other original call is left unchanged. The parameter ν is, as above, a user-defined threshold for the ratio of the intensities. Re-Analyze can be seen as a relaxation of the above base calling algorithm since it omits conformance and other neighborhood-based quality measures.

3 Software

Here, we present ResqMi, short for "*Reseq*uencing using *Mi*croarrays". ResqMi works on Affymetrix's GeneChip® Sequence Analysis platform for resequencing data, including the GeneChip® Human Mitochondrial Resequencing Array [M$^+$04]. ResqMi is implemented in C++ using the QT toolkit for the graphical user interface and the Affymetrix Fusion SDK (http://www.affymetrix.com) for data parsing. ResqMi features a graphical user interface with a similar design as Affymetrix GSEQ software in order to assure users instant usability of the software. The interface is designed to allow visual inspection and easy manual data editing. It offers a project-based organization of the data, keeping all necessary files in a data tree for easy and well-arranged access. ResqMi can work on raw intensity data (CEL file format) and processed sequence data (CHP file format), either provided by external applications or produced by base calling performed within ResqMi.

Sequence data can be viewed and edited in the Resequencing window (see figure 2). This is the central window of ResqMi, giving a fragment-wise view of the processed data of one or more arrays. In general, we focused on quick navigation and concise overviews: a header view summarizes the base composition of the reference sequence and an overview of the called bases, highlights no-calls as well as heterozygous and homozygous non-reference calls. The called sequences are displayed aligned to the reference, also highlighting non-reference calls. Editing sequences can easily be done in place, just by typing or using a context menu. Only valid IUPAC nucleotide symbols can be entered. Search functions, specifically for non-reference calls, further enhance navigating in the sequences. All other windows are connected with the Resequencing window to adjust to the currently selected position. For swift finding of interesting locations, such as specific sequence features or known polymorphic sites, a bookmark system has been implemented that allows the user to jump between locations. An in-detail tabular view of the calls and quality scores is implemented in the Resequencing table. Here, editing the called base is also possible. Reports can be generated from sequence data, that includes an overview of the `n` calls and the type and position of non-reference calls. For visual inspection of intensity data, the CEL intensity window shows a table of the intensity data (for all eight probes for each position) and a lineplot of the currently selected base and its two nearest neighbors. This view of the data allows to identify possible non-reference calls or regions of low or saturated intensities. When working on many hybridized arrays, the full CEL intensity window may be too large. A shrunk window showing the intensity plot without the values is available for this case. There are several ways of obtaining sequences in ResqMi. CEL files can be processed in Affymetrix GSEQ and the resulting CHP and CEL files are imported in ResqMi. CEL files can also be directly analyzed using ResqMi. For this purpose, ResqMi features its own implementations of the calling algorithm described above. Additionally,

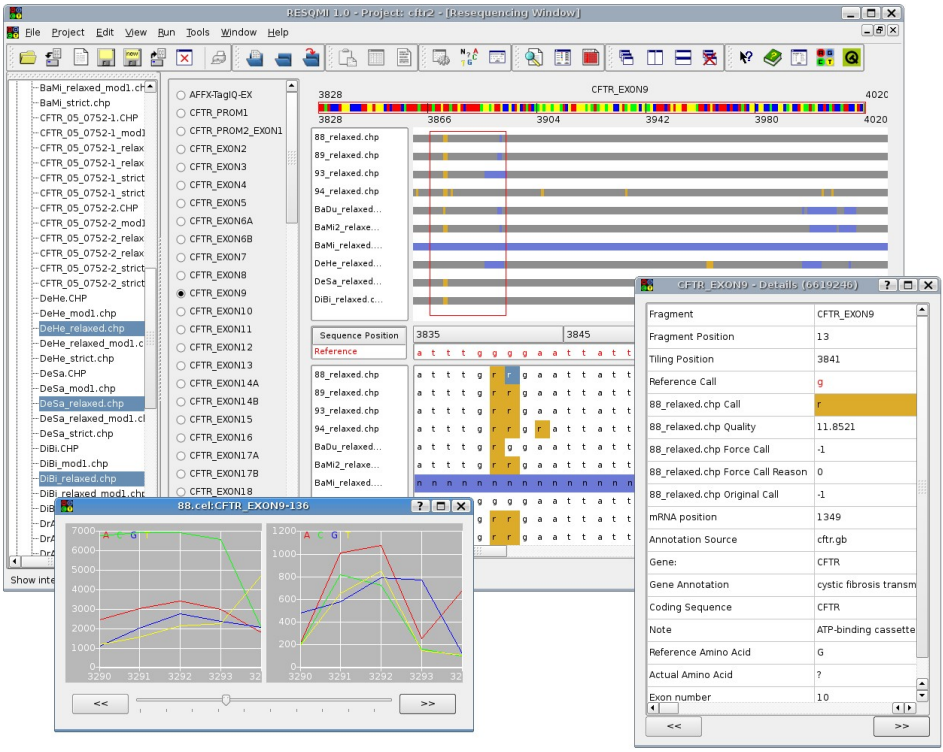


Figure 2: Central Resequencing window of ResqMi, including intensity view (bottom left) and position details exemplified here for the case of a mutation in the CFTR gene.

Affymetrix GSEQ can be executed from within ResqMi to produce calls from intensity data.

ResqMi offers a plugin interface, which allows an easy extension of its functionality. Currently, we have implemented the model-based calling algorithm and Re-Analyze as ResqMi plugins. Parameters and results can be set and inspected in a GUI, showing all necessary information to readily interpret the results. The base-calling algorithm yields CHP files such that GSEQ and other Affymetrix software can read them.

Re-Analyze can be applied to any CHP file in the data tree. Especially when processing files that contain many no-calls (`n`'s), Re-Analyze is helpful to resolve the straightforward cases, such as bases that are clearly homozygous.

When detecting a mutation, it is necessary to estimate its impact. A mutation can be, depending on its position, either synonymous (the resulting amino acid stays the same), or non-synonymous, leading to a different amino acid or to a major effect on the gene product, including missing start codon, premature stop of translation, or new or missing splice sites. If a mapping of resequencing fragments to the genomic or mRNA sequence of a gene is available, ResqMi can give detailed information on the position, whether it

is within the coding sequence, intron, exon or if this position has been identified as SNP position. A basic helper application called ResqMap is included in ResqMi that produces the required mappings from the array-specific library file (the so-called CDF file) to one or more GenBank data files.

ResqMi is open source software released under the GPL. Binaries for different platforms and the source code are available at http://www-ps.informatik.uni-tuebingen.de/resqmi.

4 Results

We used three data sets of three different resequencing microarrays (see table 1 for details). The data set encompassed a custom-made resequencing array of the disease-related human CFTR gene (unpublished data), the human mitochondrium and the Coronavirus causing SARS (Severe Acute Respiratory Syndrome). The sequence lengths per array ranged from 9,511 to 37,756 base pairs. Altogether, 75 arrays were used. For CFTR and Mito, fully analyzed sequence data produced with the current Affymetrix software was available. The

Table 1: Key features of the data sets used. Note that each array contains several fragments for sites of known mutations of the main target.

Name	Target	Bases	Fragments	Experiments
CFTR[a]	CFTR	9511	84	17
Mito[b]	Human Mitochondrium	37756	480	14
SARS[c]	SARS Coronavirus	30588	3	44

[a]unpublished data
[b]http://www.affymetrix.com/support/downloads/demo_data/demo_data_mito.zip
[c]Array Express, accession numbers E-MEXP-510 and E-MEXP-511.

data was imported into ResqMi and the model based calling algorithm was applied with different settings to evaluate the algorithm parameters.

We tested every combination of conformance parameter $\mu \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$ and ratio parameter $\nu \in \{1.01, 1.05, 1.1, 1.25, 1.33\}$, both for strict and relaxed consensus calls. For each parameter combination and array we computed the ratio of no-calls (see figure 3).

In all datasets, we observe that both parameters have some impact. Higher values of ν generally lead to higher `n` counts. The effect is larger in the CFTR and SARS datasets than in the Mito dataset. Increasing the value of the conformance cutoff μ generally only has a small effect, except for the Mito dataset. However, $\mu = 0.9$ leads to considerably more `n`-calls in all datasets. The results heavily depend on the data. Especially the SARS and CFTR datasets contain arrays which have relatively low `n`-call rates as well as arrays with virtually no called bases, leading to heavy outliers (see table 2). Relaxing the requirements for the conformance would not be useful to produce calls on some arrays, since mean values for conformance as low as 0.25 are observed. The consensus method, i.e. whether

strict or relaxed consensus is used, also affects the calling rate. For strict calls, we observe a higher rate of n-calls and fewer discrepancies (see figure 3 and table 2), while fewer n-calls and more (ambiguous) alternative calls are made when using relaxed consensus calls. This is especially pronounced in the Mito data set where we observed a decrease of the n-call rate from 35.6% to 31.5% with an increase from 1.4% to 5.5% for the mean discrepancy rate. The impact of the parameters on the number of called bases differing from the reference sequence is similar. For low values of μ and ν, the rate of discrepancies is higher than for more restrictive ones. The ratio cutoff, however has a far lower impact on the rate as the conformance cutoff. Naturally, the relaxed consensus method leads to more discrepant calls than the strict consensus method, especially at lower values for μ. Note that generally calls made using low values of μ and ν might be unreliable. Therefore, choosing higher values can increase accuracy, albeit at a lower call rate.

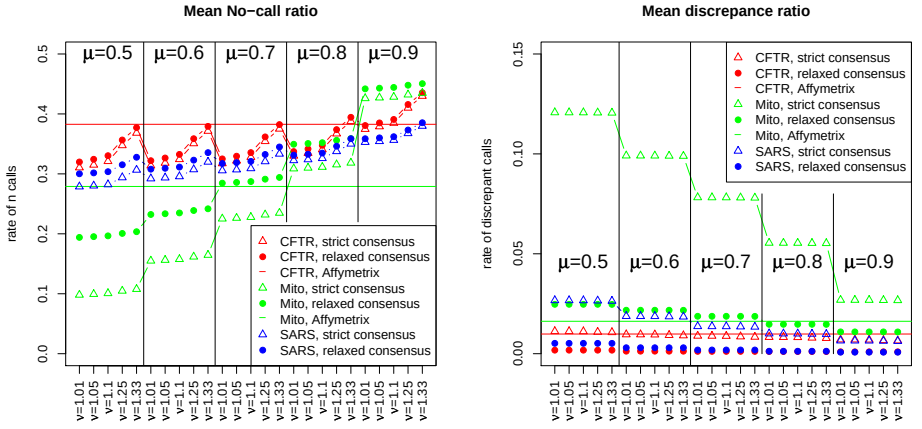


Figure 3: Mean No-call-ratio (left) and discrepancies of reference sequence ratio (right) in three datasets using strict consensus calls (triangles) and relaxed consensus calls (dots) for different values of μ and ν. For comparison, the two horizontal lines show no-call-ratio of the Affymetrix algorithm on the same data.

We observe that the results of the model-based approach are comparable to the calling rate of the Affymetrix algorithm. In the CFTR dataset, for all parameters the no-call ratio is lower or roughly equal to the Affymetrix results, both for strict and relaxed calls. In the Mito dataset, for $\mu \leq 0.7$, the n-call rate is below or equal to the Affymetrix algorithm, for relaxed and strict consensus respectively.

Generally, there is a good concordance between the called bases and the reference sequence. Cohen's Kappa statistic (K) [Coh60], summarizing this concordance, is ≥ 0.5 for most parameter sets. The Affymetrix algorithm yields comparable values for Kappa (data not shown).

For further analysis of the algorithm, we used parameters $\mu = 0.8$ and $\nu = 1.25$ (see table 2). These values should ensure reliable calls at a moderate no-call ratio. At these parameters, the K is ≥ 0.54 for all datasets. We found that the model-based calling

algorithm generally reduces the correlation between n-calls and low intensities, compared to the Affymetrix algorithm. Relaxed base calls generally lead to lower correlations than strict base calls, implying that the algorithm is less prone to produce n-calls for positions with lower intensities (data not shown). Applying Re-Analyze to all datasets, yielded further improvements of the calling rate (see table 3). By empirical testing it was found that $\nu \geq 1.2$ is a reasonable value, reducing the n-rate by up to 9.35%.

Table 2: Mean percentage of n-calls (parenthesis shows range), percentage of discrepancies and Kappa statistic (K) for all datasets using parameters $\mu = 0.8$, $\nu = 1.25$. Note that K is equal for both consensus methods.

	Strict consensus call		Relaxed consensus call		
Name	%n	% disc.	%n	% disc.	K
CFTR	37.3 (18.5. . .100)	0.1	36.7 (17.9. . . 100)	0.8	0.56
Mito	35.6 (32.3. . .38.4)	1.4	31.5 (27.2. . . 34.7)	5.5	0.54
SARS	34.6 (1.55. . .97.8)	0.1	33.7 (1.49. . . 97.8)	0.9	0.61

Table 3: Rates of n-calls after application of Re-Analyze and mean percentage of n-calls resolved by Re-Analyze, for Affymetrix calls, strict and relaxed consensus calls ($\mu = 0.8$, $\nu = 1.25$).

Dataset	Affymetrix % resolved	Strict consensus % resolved	Relaxed consensus % resolved
CFTR	8.81	9.24	9.35
Mito	8.69	7.73	7.67
SARS	-	3.71	3.59

5 Discussion

ResqMi is the first freely available open source and multi-platform software for the analysis of resequencing microarray data. In order to allow a fast and flexible evaluation of intensity and sequence data, ResqMi features a graphical user interface and allows easy data handling. ResqMi offers several views of the data, from large-scale overviews to spot-oriented views of the intensity data necessary to make most of the data available. With ResqMi, the actual impact of a mutation can easily be identified. Since all calling algorithms fail to call bases for many sequence positions, manual inspection features like the ones offered by ResqMi are pressingly needed. Altogether, we think that ResqMi is a valuable tool for working with resequencing data. A whole analysis of a resequencing dataset including import, base calling, running Re-Analyze and generation of reports takes less than 5 minutes depending on the dataset. So far our software only processes arrays

produced with the Affymetrix GeneChip® technology. However, to our knowledge most resequencing arrays published were produced by Affymetrix.

The currently largest resequencing array is Affymetrix's CustomSeq® 300k array that allow the analysis of up to 300,000 bases of double stranded sequence (600,000 bases total) on a single array. Since all current base calling algorithms leave a part of the sequence uncalled, several thousand bases must be manually inspected, even in the rare case of extremely low no-call rates of 5% or less. Combining the model-based base calling algorithm of Clark et al. with a subsequent application of Re-Analyze we have shown that up to 10% of the n's can be resolved. In a resequencing setting, independently produced sequences are usually not available. Therefore, a strict precision estimation is impossible. However, we assessed the concordance of the called bases with the reference sequence using the Kappa statistic and found reasonably good results.

The default base calling algorithm in ResqMi follows the typical approach of evaluating and processing resequencing data. During manual inspection bases are called when its signal distance to the second-highest and the reliability of the neighborhood is large. Therefore, the parameters μ and ν are easy to understand. Furthermore, the application to several data sets indicate that the impact of parameter changes is generally linear and predictable. Although the model-based algorithm is simple, we found that on most data sets the performance in terms of call ratio is comparable or better in comparison to the ABACUS algorithm as implemented in Affymetrix's GSEQ software. We recommend as default parameters for μ and ν to use 0.8 and 1.25, respectively, since these values produced high calling rates with low risk of false discrepancy calls. Lowering the cutoffs below these values will increase the calling rate, but will as shown increase the rate of discrepancies, which may be unreliable. Due to this tradeoff, such parameter sets may only be useful when analyzing especially noisy data.

Subsequent application of Re-Analyze helps to automatically lower the fraction of uncalled positions that are clearly homozygous loci. Heterozygous positions indicated by conflicting calls for the sense and antisense strand as well as a possibly small intensity ratio of the highest and second highest signal remain uncalled, and therefore Re-Analyze will not produce false calls.

As a plugin-based architecture, adding new methods for visualizing, analyzing and editing is easy and will further improve ResqMi. Future directions include adding new functionality to ResqMi as technology evolves. Although the calling algorithm performs well on some datasets, the algorithm should further be improved in order to cope with extremely noisy data. Further testing with other data sets should improve our understanding of how to set parameters and how to devise improvements of the algorithm. Ambiguous signals can also be the result of weak hybridization affinity, saturated signals or cross-hybridization. An improved base calling algorithm should therefore also take the hybridization condition of the individual probes in consideration [Hac99].

Generally, further research in this field is necessary in order to generate more complete and reliable calls from a wide range of resequencing data.

References

[A+05] Thomas J Albert et al. Mutation discovery in bacterial genomes: metronidazole resistance in Helicobacter pylori. *Nature Methods*, 2(12):951–953, 2005.

[Aff04] Affymetrix, Inc. GeneChip® CustomSeq Resequencing Array Program. Technical report, Affymetrix, Inc, 2004.

[Aff06] Affymetrix, Inc. GeneChip® CustomSeq Resequencing Array Base Calling Algorithm Version 2.0: Performance in Homozygous and Heterozygous SNP Detection. Technical report, Affymetrix, Inc, 2006.

[BS88] William Bains and Geoff C. Smith. A Novel Method for Nucleic Acid Sequence Determination. *J Theor Biol*, 135:303–307, 1988.

[C+01] David J Cutler et al. High-Throughput Variation Detection and Genotyping Using Microarrays. *Genome Research*, 11:1913–1925, 2001.

[C+07] Richard M. Clark et al. Common Sequence Polymorphisms Shaping Genetic Diversity in Arabidopsis thaliana. *Science*, 317:338–342, 2007.

[Coh60] Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20:37–46, 1960.

[D+05] Louis Davignon et al. Use of Resequencing Oligonucleotide Microarrays for Identification of Streptococcus pyrogens and Associated Antibiotic Resistance Determinants. *Journal of Clinical Microbiology*, 43(11):5690–5695, 2005.

[H+05] David A. Hinds et al. Whole-Genome Patterns of Common DNA Variation in Three Human Populations. *Science*, 18:1072–1079, 2005.

[Hac99] Joseph G. Hacia. Resequencing and mutational analysis using oligonucleotide microarrays. *Nature Genetics*, 21:42–47, 1999.

[J+08] Crystal Jaing et al. A Functional Gene Array for Detection of Bacterial Virulence Elements. *PLOS one*, 3(5):e2163, 2008.

[L+06] Baochuan Lin et al. Broad-spectrum respiratory tract pathogen identification using resequencing DNA microarrays. *Genome Research*, 16:527–535, 2006.

[LF94] Robert J Lipshutz and Stephen PA Fodor. Advanced DNA sequencing technologies. *Current Opinion in Structural Biology*, 4:367–380, 1994.

[M+04] Anirban Maitra et al. The Human MitoChip: A High-Throughput Sequencing Microarray for Mitochondrial Mutation Detection. *Genome Research*, 14:812–819, 2004.

[M+06] Anthony P. Malanoski et al. Automated identification of multiple micro-organisms from resequencing DNA microarrays. *Nucleic Acids Research*, 34(18):5300–5311, 2006.

[M+07] Suhail K. Mithani et al. Mitochondrial Resequencing Arrays Detect Tumor-Specific Mutations in Salivary Rinses of Patients with Head and Neck Cancer. *Cini Cancer Res*, 13:7335–7340, 2007.

[S+06] Irshad M. Sulaiman et al. Evaluation of Affymetrix Severe Acute Respiratory Syndrome Resequencing GeneChips in Characterization of the Genomes of Two Strains of Coronavirus Infecting Humans. *Applied and Environmental Microbiology*, 72:207–211, 2006.

[v+06] Rudy G. E. van Eijsden et al. Chip-based mtDNA mutation screening enables fast and reliable diagnosos of OXPHOS patients. *Genetics in Medicine*, 8(10):620–627, 2006.

[W+06] Zheng Wang et al. Identifying Influenza Viruses with Resequencing Microarrays. *Emerging Infectios Diseases*, 12:638–646, 2006.

[ZK05] Yiping Zhan and David Kulp. Model-P: a basecalling method for resequencing microarrays of diploid samples. *Bioinformatics*, 21(Suppl. 2):ii182–ii189, 2005.

Exploring the Enzyme Neighbourhood to interpret gene expression data

Nicolas Goffard, Tancred Frickey, Nijat Imin, Georg Weiller

ARC Centre of Excellence for Integrative Legume Research
Bioinformatics Laboratory, Genomic Interactions Group
Research School of Biological Sciences, Australian National University
GPO Box 475, Canberra, ACT 2601, Australia
ngoffard@gmail.com
tancred.frickey@anu.edu.au
nijat.imin@anu.edu.au
georg.weiller@anu.edu.au

Abstract: Post-genomic data analysis represents a new challenge to link and interpret the vast amount of raw data obtained with transcriptomic or proteomic techniques in the context of metabolic pathways. We propose a new strategy with the help of a metabolic network graph to extend PathExpress, a web-based tool to interpret gene expression data, without being restricted to predefined pathways. We defined the Enzyme Neighbourhood as groups of linked enzymes, corresponding to a sub-network, to explore the metabolic network in order to identify the most relevant sub-networks affected in gene expression experiments.

1 Introduction

With the development of transcriptomic and proteomic techniques, post-genomic data analysis represents a new challenge for researchers to link the vast amount of raw data to a biological context [Br06]. The interpretation of microarray data is usually performed in two steps. The first step is the identification of genes that are differentially expressed under two or more conditions, using different statistical methods [CC03]. In a second step, the selected genes are compared with a background in order to find enrichment in any functional term. Many ontological tools are now available that support the functional interpretation of gene expression data, through the identification of significantly enriched Gene Ontology categories [As00] among a class of genes of interest [KD05].

Additionally, with the availability of pathway databases such as the Kyoto Encyclopaedia of Genes and Genomes (KEGG) [KG00] or MetaCyc [Ca06], numerous tools have been proposed to visualize and analyse microarray data in the context of known biological networks by including metabolic or regulatory pathway information [Pa03], [PGM04], [Th04], [Ch05], [Ml05], [Ba06], [Wu06], [GW07], [Sa07]. However, the predefined metabolic pathways used in these methods represent an arbitrary segmentation of metabolism.

In contrast, other methods integrate, *a priori,* the knowledge of gene networks in the analysis of gene expression data. Ideker and co-workers presented a procedure for screening a molecular interaction network combined with a statistical measure to identify sub-networks that show significant changes in expression [Id02]. This approach has been included in Cytoscape to identify functional modules, i.e. highly connected network regions with similar responses across multiple experimental conditions [Cl07]. Hanisch and co-workers proposed a co-clustering method based on a distance function that combines information from expression data and biological networks [Ha02]. A Potts spin algorithm was developed to cluster gene expression data by using the nearest neighbour relations of biochemical networks [KE04]. Rapaport and co-workers extracted gene expression patterns of neighbouring genes in the network, involving the attenuation of high-frequency signals with respect to the graph [Ra07]. Another approach consists of the development of techniques for the decomposition of biochemical networks into the smallest functional units based on the network topology using the Petri net theory [Sc02], [SHK06]. It has been shown by Schwartz and co-workers that elementary modes represent true functional units of metabolism and can be used to reveal transcriptional activity [Sc07]. However, these methods are limited by the combinatorial explosion of computing elementary modes in large networks.

We recently presented a web-based tool called PathExpress [GW07] to interpret gene expression results from microarray experiments in the context of biological pathways, available at http://bioinfoserver.rsbs.anu.edu.au/utils/PathExpress/. PathExpress has been developed to identify the most relevant pathways or sub-pathways associated with a subset of genes, e.g., differentially expressed. It is based on a directed graph to model enzymatic reactions, derived from the publicly available KEGG Ligand database of chemical compounds and reactions in biological pathways [GNT98], [Go02]. Two types of nodes are used to represent compounds and reactions that can be mediated by one or more enzymes. To take into account how reactions are linked in pathway, sub-pathways are defined as a chain of reactions linked to each other by a common compound (substrate or product). Thus, PathExpress compares a submitted list of genes to the genes involved in annotated pathways or sub-pathways and identifies the significantly over-represented set of enzymatic reactions in the query using a hypergeometric distribution [Ch01]. This statistical test has been employed by many ontological tools to detect significant enrichments of functional categories within a class of genes of interest [Ri07].

This article presents developments in PathExpress that explore the metabolic network for the interpretation of gene expression data. We created a graph representing the complete metabolic network, which allows us to examine the neighbourhood of a given enzyme by following the chain of connected reactions linked by a common compound. The Enzyme Neighbourhood (EN) represents a group of linked enzymes corresponding to a sub-network. The EN can then be compared to a submitted list of genes with the aim to find ENs in which the submitted genes are significantly over-represented. In a case study, our method was tested with gene expression data of the model legume *Medicago truncatula* to compare the transcriptomes of meristematic and non-meristematic root cells [Ho08].

2 Methods

This approach is based on a directed graph modelling enzymatic reactions as used in the Petri net representation of biological networks [SHK06]. Two types of nodes are used to represent compounds and reactions with reactions represented by one or more enzymes. Directed edges, connecting these nodes, correspond to the consumption or the production of compounds by the reaction. We first built the global metabolic network consisting of 2,198 enzymes and 2,796 compounds involved in 3,706 reactions as specified in the KEGG LIGAND database [GNT98], [Go02]. This database has the advantage of providing a manually curated representation of enzymatic reactions involved in metabolic pathways where most secondary metabolites (very common and highly connected compounds such as water, oxygen, major coenzymes and prosthetic groups) have been removed, thus avoiding invalid metabolic connections and unspecified pathways.

In this network, two reactions are neighbours if a metabolite exists that is the product of one reaction and the substrate for the other. Then, we define the Enzyme Neighbourhood (EN) of depth *d* for an enzyme *e*, as the set of enzymes that can be reached in the graph from *e* by traversing a maximum of *d* compounds, regardless of the direction of the edges (Fig 2.1). The EN of depth 1 for a given enzyme thus corresponds to the set of enzymes directly connected via a compound. The EN of depth 2 includes the enzymes involved in the EN of depth 1 plus the enzymes linked to them. As different paths can connect two enzymes, the shortest distance is considered to define the EN. These ENs correspond to different sub-networks of the global metabolic network.

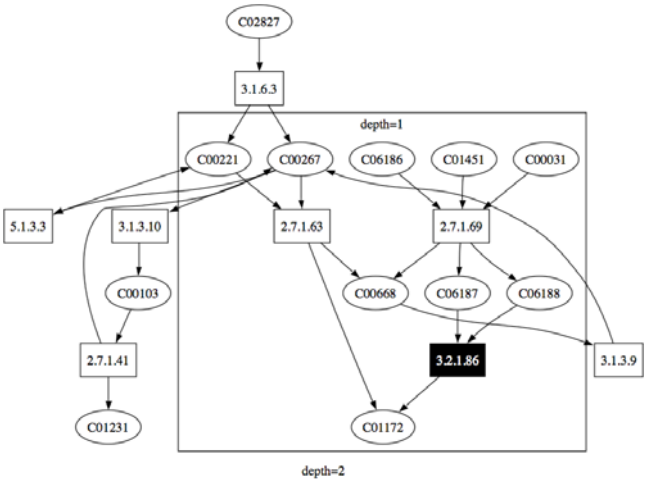


Figure 2.1: Example of an Enzyme Neighbourhood (EN). Compounds (labelled with their KEGG identifier and represented as ellipses) and reactions (labelled with the EC number of the enzymes that mediates it and represented as boxes) are the nodes of the directed graph. The EN of depth 1for the enzyme 'EC 3.2.1.86' contains the enzymes 'EC 3.2.1.86', 'EC 2.7.1.69' and 'EC 2.7.1.63', whereas the EN of depth 2 contains in addition to those includes in the EN of depth 1, the enzymes 'EC 3.1.6.3', 'EC 5.1.3.3', 'EC 3.1.3.9', 'EC 2.7.1.41' and 'EC 3.1.3.10'.

To identify the most relevant sub-network associated with a list of submitted enzymes, the EN of each seed (EC number) is determined in the global network, for a given depth, and its composite EC numbers are compared to the submitted list. For each test, a *p*-value representing the probability that the intersection *k* of the list of enzymes of size *n* belonging to the given EN, of size *D*, occurs per chance in the population of *N* enzymes involved in the entire network, is calculated using the hypergeometric distribution [Ch01] as described below.

$$p(k,N,D,n) = \frac{\binom{D}{k}\binom{N-D}{n-k}}{\binom{N}{n}}$$

Because multiple hypothesis tests are performed, it is necessary to correct these *p*-values with adjustment methods such as the conservative Bonferroni correction [Bo06], in which the *p*-values are multiplied by the number of comparisons, or the less stringent False Discovery Rate (FDR) approach [BH95], which determines the expected proportion of false positive results among all rejected hypotheses.

The size *D* of the EN depends on its depth *d*, which has to be specified as a parameter in the current implementation. It is typically necessary to examine several ENs with different depths. To optimize this parameter with the size of the submitted list of enzymes, we have computed the average number of enzymes involved in each possible EN for a range of depths (Table 2.1). Using these results, it is possible to adjust the depth parameter to compare groups of enzymes with sub-networks of similar size. For example, to compare a group of 10 enzymes, a depth parameter of 1 (i.e. direct neighbours), corresponding to an average size of 11.7 enzymes in the network, is recommended.

Table 2.1: Average size of the Enzyme Neighbourhood according to the depth parameter

Depth	Average no. of neighbours
1	11.7
2	14.5
3	21.9
4	34.0
5	51.0
6	74.2
7	105.5
8	145.1
9	193.8
10	253.5
20	995.0
30	1397.7
40	1622.1
50	1767.4
100	2106.8

3 Application to gene expression data

We extended the web-based tool PathExpress with this method of exploring the Enzyme Neighbourhood in order to identify the most relevant sub-networks associated with a list of genes (e.g. differentially expressed genes).

3.1 Linking expressed enzymes with metabolic networks

One of the main constraints in methods for the functional interpretation of gene expression data corresponds to the linkage of such data to the metabolic network, as the number of available organisms in pathway databases is limited. To overcome this, we use similarities between probe set sequences of supported genome arrays and protein sequences of known EC numbers, retrieved from the Swiss-Prot database [Ba05], in order to link probe sets to the metabolic network (Table 3.1). Blastx [Al90] is used to find the best match (E-value $\leq 10^{-8}$) for the sequences representing each probe set sequence (i.e. sequences derived from the most 5' to the most 3' probe in the public Unigene cluster) of the genome arrays analyzed. If these entries have been annotated as enzymes, the probe set is assigned to the corresponding EC number, extracted from its definition line. This strategy can be applied to any set of sequences. A complete metabolic graph representing all assignments is produced and all qualifying sub-networks are compared with the data of a submitted genome array. High scoring Enzyme Neighbourhoods are then presented.

Note that probe sets that cannot be assigned to EC numbers are excluded from further analyses, and although this limits the number of usable probe sets, it also eliminates non-enzymatic gene functions that are present in many unrelated metabolic pathways. As the comparisons are based on enzyme composition rather than single probe set assignments, biases that arise from a multiplicity of genes coding for the same enzyme are largely overcome and the functional activities become apparent.

Table 3.1: Available Affymetrix genome arrays and assignment statistics

Affymetrix Genome Array (Organism)	% Sequences assigned	No. of ECs	No. of reactions
ATH1 Genome Array (*A. thaliana*)	22.7	823	1,177
E. coli Genome 2.0 Array (*E. coli*)	22	803	1,217
Drosophila Genome 2.0 Array (*D. melanogaster*)	16.4	724	1,011
Yeast Genome 2.0 Array (*S. cerevisiae*)	25.3	601	918
Yeast Genome 2.0 Array (*S. pombe*)	26.5	566	839
Medicago Genome Array (*M. truncatula*)	17.6	953	1,412
Soybean Genome Array (*G. max*)	17.2	803	1,217
Rice Genome Array (*O. sativa*)	17.6	923	1,363

3.2 Microarray data analysis

Our method was applied to interpret a microarray experiment in the model legume *Medicago truncatula*, comparing the gene expression of meristematic and non-meristematic root tissues [Ho08]. The data have been deposited in NCBI's Gene Expression Omnibus [EDL02] and are accessible through GEO series accession number GSE8115 (http://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE8115). Following normalisation, differentially expressed probe sets were identified by evaluating the log2 ratio between the two conditions associated to a standard *t*-test [Ca00]. All probe sets that differed by more than a two-fold difference with a *t*-test $p \leq 0.05$ were considered to be differentially expressed. Of the 363 transcripts over-expressed in the non-meristem, 119 could be assigned to 62 different enzymatic functions, defined by their EC number and found in the Affymetrix Medicago Genome Array. In order to identify the most relevant sub-networks involved in this group, we compare it, using PathExpress, to all ENs with a depth of 6, using the hypergeometric distribution. The resulting sub-networks were ranked by increasing *p*-values, representing the probability that the intersection of the enzymes differentially expressed in the non-meristem with the given EN occurs by chance.

The most significant EN (*p*-value = 1.4e-4), using the flavonone 3-dioxygenase (EC 1.14.11.9) as seed (black), is given in Figure 3.1. Of the 20 enzymatic reactions present in the depicted sub-network, 9 occur in the submitted list of differentially expressed enzymes (grey and black). Only 12 of the 20 reactions in this EN are part of the classical flavonoid biosynthesis pathway as described in the KEGG database, which is consistent with the role for the flavonoids and their derivatives in the non-meristematic root [Im07]. The remaining 8 reactions connected to this sub-network are part of different pathways (such as propanoate metabolism or liminene and pinene degradation) and would not have been considered by an approach restricted to predefined metabolic pathways.

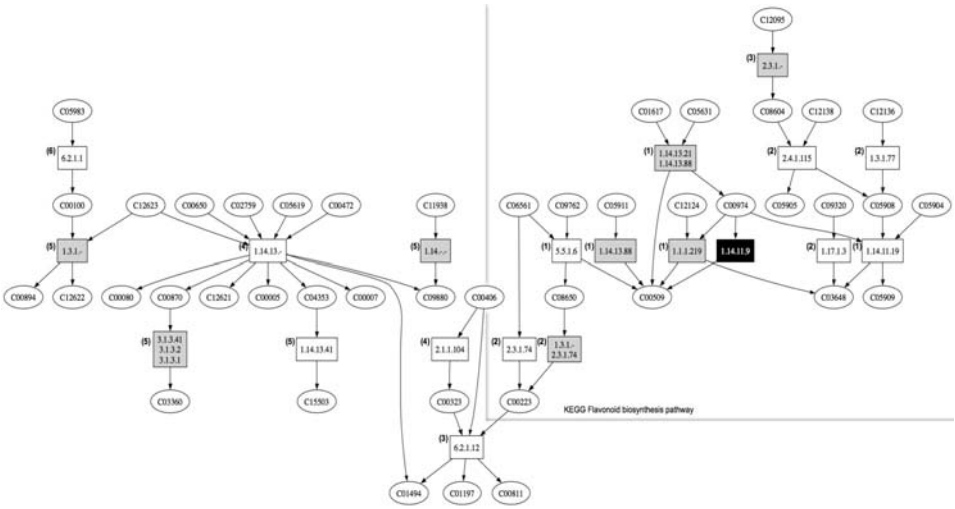


Figure 2.1: Enzyme Neighbourhood of depth 6, identified from a list of differentially expressed genes in *Medicago truncatula*. For each reaction represented, the EN depth is indicated (number in brackets). Genes encoding enzymes for all these reactions have been identified in the Affymetrix Medicago Genome Array. The reaction coloured in black corresponds to the enzyme (EC 1.14.11.9) used to establish this EN. Greyed reactions show that at least one of the corresponding enzymes belongs to the submitted group of enzymes. The set of reactions inside the frame represent part of the classical flavonoid biosynthesis pathway as described in KEGG database.

4 Conclusion

The interpretation of microarray experiments represents a main challenge to characterize biological processes. This paper presents a method to interpret results of gene expression data in the context of metabolic pathways. Our web-based tool PathExpress, in which metabolic pathways are modelled as directed graphs of enzymatic reactions, has been extended to identify Enzyme Neighbourhoods (EN) with statistically significant differential expressions. The EN of a given enzyme is defined as a connected sub-network within the global metabolic network, built from the KEGG database. This method is based on the same statistical approach as used for the identification of gene enrichment in GO terms or metabolic pathways. However, the clustering method differs, as it includes knowledge about the network of gene products without being restricted to predefined pathways. Based on a pre-computed assignment of sequences to EC numbers this approach can be applied to any organism or set of sequences (e.g. custom DNA microarray, proteome array) and hence provides a useful resource for the integration of transcriptomic and proteomic data sets.

References

[Al90] Altschul, S.F. et. al.: Basic local alignment search tool. *J. Mol. Biol.*, 1990; vol. 215(3), pp. 403-410.

[As00] Ashburner, M. et. al.: Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat. Genet.*, 2000; vol. 25, pp. 25-29.

[Ba05] Bairoch, A. et. al.: The Universal Protein Resource (UniProt). *Nucleic Acids Res.*, 2005; vol. 33(Database issue), pp. D154-159.

[Ba06] Baitaluk, M. et. al.: BiologicalNetworks: visualization and analysis tool for systems biology. *Nucleic Acids Res*., 2006; vol. 34(Web Server issue), pp. W466-471.

[BH95] Benjamini, Y.; Hochberg, Y.: Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J. R. Statist. Soc. B.,* 1995; vol. 57(1), pp. 289 - 300.

[Bo36] Bonferroni, C.: Teoria statistica delle classi e calcolo delle probabilità. *Pubblicazioni del Regio Istituto Superiore di Scienze Economiche e Commerciali di Firenze,* 1936; vol. 8, pp. 3-62.

[Br06] Breitling, R.: Biological microarray interpretation: the rules of engagement. *Biochim. Biophys. Acta.*, 2006; vol. 1759(7), pp. 319-327.

[Ca00] Callow, M.J. et. al.: Microarray expression profiling identifies genes with altered expression in HDL-deficient mice. *Genome Res.*, 2000; vol. 10, pp. 2022-2029.

[Ca06] Caspi, R. et. al.: MetaCyc: a multiorganism database of metabolic pathways and enzymes. *Nucleic Acids Res*., 2006; vol. 34(Database issue), pp. D511-516.

[CC03] Cui, X.; Churchill, G.A.: Statistical tests for differential expression in cDNA microarray experiments. *Genome Biol.*, 2003; vol. 4(4), pp. 210.

[Ch01] Cho, R.J. et. al.: Transcriptional regulation and function during the human cell cycle. *Nat. Genet.*, 2001; vol. 27(1), pp. 48-54.

[Ch05] Chung, H.J. et. al.: ArrayXPath II: mapping and visualizing microarray gene-expression data with biomedical ontologies and integrated biological pathway resources using Scalable Vector Graphics. *Nucleic Acids Res*., 2005; vol. 33(Web Server issue), pp. W621-626.

[Cl07] Cline, M.S. et. al.: Integration of biological networks and gene expression data using Cytoscape. *Nat. Protoc.*, 2007; vol. 2(10), pp. 2366-2382.

[EDL02] Edgar, R.; Domrachev, M.; Lash, AE.: Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. *Nucleic Acids Res., 2002; vol*. 30(1), pp. 207-210.

[Go02] Goto, S. et. al.: LIGAND: database of chemical compounds and reactions in biological pathways. *Nucleic Acids Res*., 2002; vol. 30(1), pp.402-404.

[GNT98] Goto, S.; Nishioka, T.; Kanehisa, M.: LIGAND: chemical database for enzyme reactions. *Bioinformatics*, 1998; vol. 14(7), pp. 591-599.

[GW07] Goffard, N.; Weiller, G.: PathExpress: a web-based tool to identify relevant pathways in gene expression data. *Nucleic Acids Res*., 2007; vol. 35(Web Server issue), pp. W176-181.

[Ha02] Hanisch, D. et. al.: Co-clustering of biological networks and gene expression data. *Bioinformatics*, 2002; vol. 18 Suppl 1, pp. S145-154.

[Ho08] Holmes, P. et. al.: Transcriptional profiling of Medicago truncatula meristematic root cells. *BMC Plant Biol*., 2008; vol. 8(1), pp. 21.

[Id02] Ideker, T. et. al.: Discovering regulatory and signalling circuits in molecular interaction networks. *Bioinformatics*, 2002; vol. 8 Suppl 1, pp. S233-240.

[Im07] Imin, N. et. al.: Factors involved in root formation in Medicago truncatula. *J. Exp. Bot.*, 2007; vol. 58(3), pp. 439-451.

[KD05] Khatri, P.; Drăghici, S.: Ontological analysis of gene expression data: current tools, limitations, and open problems. *Bioinformatics*, 2005; vol. 21(18), pp. 3587-3595.

[KE04] König, R.; Eils, R.: Gene expression analysis on biochemical networks using the Potts spin model. *Bioinformatics*, 2004; vol. 20(10), pp. 1500-1505.

[KG00] Kanehisa, M.; Goto, S.: KEGG: Kyoto encyclopedia of genes and genomes. *Nucleic Acids Res.*, 2000; vol. 28, pp. 27-30.

[Ml05] Mlecnik, B. et. al.: PathwayExplorer: web service for visualizing high-throughput expression data on biological pathways. *Nucleic Acids Res.*, 2005; vol. 33(Web Server issue), pp. W633-637.

[Pa03] Pan, D. et. al.: PathMAPA: a tool for displaying gene expression and performing statistical tests on metabolic pathways at multiple levels for Arabidopsis. *BMC Bioinformatics*, 2003; vol. 4, pp. 56.

[PGM04] Pandey, R.; Guru, R.; Mount, D.: Pathway Miner: extracting gene association networks from molecular pathways for predicting the biological significance of gene expression microarray data. *Bioinformatics*, 2004; vol. 20, pp. 2156-2158.

[Ra07] Rapaport, F. et. al.: Classification of microarray data using gene networks. *BMC Bioinformatics*, 2007; vol. 8, pp. 35.

[Ri07] Rivals, I. et. al.: Enrichment or depletion of a GO category within a class of genes: which test? *Bioinformatics*, 2007; vol. 23(4), pp. 401-407.

[Sa07] Salomonis, N. et. al.: GenMAPP 2: new features and resources for pathway analysis. *BMC Bioinformatics*, 2007; vol. 8, pp. 217.

[Sc02] Schuster, S. et. al.: Exploring the pathway structure of metabolism: decomposition into subnetworks and application to Mycoplasma pneumoniae. *Bioinformatics*, 2002; vol. 18(2), pp. 351-361.

[Sc07] Schwartz, J.M. et. al.: Observing metabolic functions at the genome scale. *Genome Biol.*, 2007; vol. 8(6), pp. R123.

[SHK06] Sackmann, A.; Heiner, M.; Koch, I.: Application of Petri net based analysis techniques to signal transduction pathways. *BMC Bioinformatics*, 2006; vol. 7, pp. 482.

[Th04] Thimm, O. et. al.: MAPMAN: a user-driven tool to display genomics data sets onto diagrams of metabolic pathways and other biological processes. *Plant J.*, 2004; vol. 37(6), pp. 914-939.

[Wu06] Wu, J. et. al.: KOBAS server: a web-based platform for automated annotation and pathway identification. *Nucleic Acids Res.*, 2006; vol. 34(Web Server issue), pp. W720-724.

Identifying the topology of protein complexes from affinity purification assays

Caroline C. Friedel* and Ralf Zimmer
Institut für Informatik, Ludwig-Maximilians-Universität München,
Amalienstraße 17, 80333 München, Germany
Caroline.Friedel@bio.ifi.lmu.de

Abstract: Recent advances in high-throughput technologies have made it possible to investigate not only individual protein interactions but the association of these proteins in complexes. So far the focus has been on the prediction of complexes as sets of proteins from the experimental results while the modular substructure and the physical interactions within protein complexes have been mostly ignored. In this article, we present an approach for identifying the direct physical interactions and the subcomponent structure of protein complexes predicted from affinity purification assays. Our algorithm calculates the union of all maximum spanning trees from scoring networks for each protein complex to extract relevant interactions. In a subsequent step this network is extended to interactions which are not accounted for by alternative indirect paths. We show that the interactions identified with this approach are more accurate in predicting experimentally derived physical interactions than baseline approaches and resolve more satisfactorily the subcomponent structure of the complexes. The usefulness of our approach is illustrated on the RNA polymerases for which the modular substructure can be successfully reconstructed with our method.

1 Introduction

Cellular processes of all sorts are shaped by proteins associated in complexes. Thus, the identification of such complexes and the interactions within the complexes have become a major experimental focus. While direct, physical interactions can be identified by the yeast two-hybrid (Y2H) approach [FS89], affinity purification methods followed by mass spectrometry, such as tandem affinity purification (TAP) [RSR$^+$99], can also identify indirect interactions via other proteins in complexes. Recently, the TAP systems was applied by Gavin *et al.* [G$^+$06] and Krogan *et al.* [K$^+$06] to identify protein complexes in the yeast *Saccharomyces cerevisiae* on a genome-scale .

In the TAP system, epitope tagged proteins (baits) are expressed and purified in consecutive affinity columns [RSR$^+$99]. Proteins interacting directly or indirectly with the bait, so-called preys, are then co-purified with the bait and identified by mass spectrometry. Ideally, the purification of one bait would yield the complete protein complex the bait is involved in. However, due to large false positive and negative rates in the experiments,

*Corresponding author.

sophisticated methods are necessary to predict the actual complexes from the purification results.

The first predictions methods were developed by the groups of Gavin *et al.* [G$^+$06] and Krogan *et al.* [K$^+$06] themselves. Since the resulting complexes showed only relatively little agreement, advanced methods have been developed recently [PVE$^+$07, C$^+$07, HLM07, FKZ08] which improved predictive performance significantly. Here, most approaches use a two-step approach by first calculating interaction scores and then predicting the complexes from those scores. However, the focus so far has been on predicting sets of proteins associated in complexes and not the substructure of the complexes or the physical interactions within these.

A few methods have been developed which analyse the substructure of protein complexes. Aloy *et al.* [A$^+$04] used interactions from 3D structures and electron microscopy to at least partially resolve interactions between subunits of 54 experimentally derived complexes. The method of Hollunder *et al.* [HBW05, HBW07, HFB$^+$07] identifies substructures in protein complexes which occur more frequently in different complexes than expected at random. As a consequence, this approach can only identify substructures in protein complexes which occur in more than one complex. Gavin *et al.* [G$^+$06] distinguished between core elements and modules or attachments in their protein complex predictions but did not predict direct interactions.

Scholtens *et al.* [SVG05] and Bernard *et al.* [BVH07] modeled the physical topology of protein complexes using both affinity purification and Y2H results. However, Scholtens *et al.* used this only as an intermediate step in predicting protein complexes and did not evaluate the actual interactions they predicted. Bernard *et al.* showed that accurate predictions can be obtained with their approach but did not evaluate to what degree their results depend on the Y2H interactions used additionally.

In this article, we investigate if the topology of protein complexes can be predicted from the affinity purification results alone. Here, the topology of a protein complex describes both the direct physical interactions within a complex (the complex scaffold) but also its modular substructure, i.e. the subdivision of the complex into smaller components. Since most methods for predicting protein complexes from affinity purification results calculate interaction scores as an intermediate step, we developed a method to extract interactions relevant for the complex scaffold from these densely connected scoring networks.

Our algorithm calculates the union of all maximum spanning trees from the interaction scores for each complex. The maximum spanning trees are then extended heuristically by interactions which are not accounted for by alternative indirect interactions. We applied our method to confidence scores and protein complexes calculated with the Bootstrap method [FKZ08] from the yeast affinity purification experiments of Gavin *et al.* [G$^+$06] and Krogan *et al.* [K$^+$06]. We show that the interactions predicted by our approach are enriched for direct physical interactions determined by Y2H experiments. Furthermore, the distance in the resulting network reflects the functional and localization similarity of the corresponding proteins and the substructure of protein complexes can be resolved in a straightforward way.

2 Methods

In the following, let $C = \{C_1, \ldots, C_n\}$ be a set of protein complexes with C_i a set of proteins and $G = (V, E)$ a weighted network of interaction scores. Here, V is the set of all proteins and E the set of all interactions between them. In the following, we assume that all scores are confidence values in the range of 0 to 1. The function $w : E \rightarrow [0, 1]$ defines the weight, i.e. the confidence score, of each edge. Interactions not contained in the network are given a weight of 0. If the scoring method calculates general scores from $-\infty$ (or 0) to ∞, edge weights are scaled to $[0, 1]$.

Furthermore, we assume that each complex is connected in the network of actual physical interactions. This means that each protein can be reached from every other protein in the same complex by an indirect path of physical (direct) interactions. This network of direct interactions is denoted as the scaffold of the complex in the following. We perform predictions separately for each complex and, consequently, two interactions with the same weight may be predicted as direct in one complex and indirect via other proteins in another one depending both on the association strength within a complex and the existence of alternative paths in this complex.

2.1 Maximum spanning trees

For each complex, we start with a fully or almost fully connected scoring network for interactions between proteins in this complex. In this network, we want to identify a hierarchical subcomponent structure and, thus, not only the largest subcomplexes but also subcomponents of these subcomplexes. This hierarchical structure can be identified with hierarchical clustering algorithms.

The most commonly used variants of hierarchical clustering are average linkage and single linkage clustering. Average linkage uses the average score between two clusters to define their similarity which makes it difficult to assign the actual physical interactions. Single linkage uses the maximum score and, accordingly, the physical interactions can be defined in a straightforward way as the interactions providing the link between two clusters.

Single linkage effectively computes the maximum spanning tree of the network if the resulting dendrogram is unrooted. A spanning tree is a tree which connects all vertices in the network. The maximum spanning tree (MST) is the spanning tree which maximizes the sum of its edge weights and can be calculated efficiently in $O(|E|+|V| \log |V|)$ [CLRS00].

Because of this relationship between hierarchical clustering and MSTs, our algorithm for predicting the scaffold of a complex is based on calculating the MST of the corresponding network. If all interaction weights within the complex are distinct, the MST is unique. As this is generally not the case in scoring networks, many MSTs can exist. As a consequence, we calculate the set of direct interactions in the complex scaffold as the union of all possible MSTs.

To calculate all interactions contained in at least one MST, we do not have to compute all

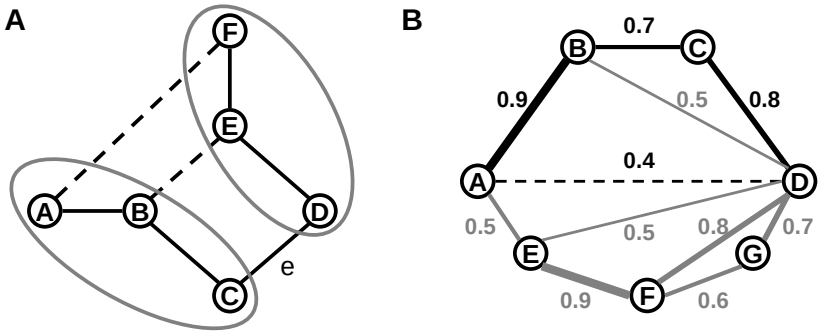


Figure 1: Figure **A** outlines how interactions contained in at least one MST are identified. The solid lines show the MST T calculated first. By removing edge e, a cut into the two sets $\{A, B, C\}$ and $\{D, E, F\}$ is created (grey ellipses). Dashed lines indicate edges crossing that cut with the same weight as e. By replacing e with any of these edges another MST T' is created. Thus, all of these edges are contained in at least one MST and added to the predicted scaffold. Figure **B** illustrates how MSTs are extended. For each possible edge (A, D), we find the optimal shortest path between the two nodes. In this case, this is $A \rightarrow B \rightarrow C \rightarrow D$ (black) which has a weight of $0.9 \cdot 0.7 \cdot 0.8 = 0.504$. Since the weight of edge (A, D) is smaller than this, the edge is discarded. If the weight of (A, D) were larger than 0.504, it would be added to the scaffold network.

possible MSTs, but can find the relevant interactions from one arbitrary MST (see Figure 1). Deleting each edge e in turn from this MST yields a cut *Cut*(T, e) – a partitioning into two sets – of the proteins in the complex. All edges crossing that cut, i.e. connecting proteins not in the same set, with the same weight as e are contained in at least one MST. With this algorithm, all edges in the union of all MSTs can be identified.

Thus, the algorithm for predicting the scaffold consists of two steps. First, an MST is calculated either with the Kruskal or Prim algorithm [Kru56, Pri57]. Second, all other edges contained in an MST are identified with the approach described above.

2.2 Extended MSTs

Although the combination of all MSTs is no longer a tree, the resulting networks are extremely sparse and many protein interactions may still be missing. As a consequence, we add a post-processing step in which we identify interactions which are not yet accounted for by an indirect interaction via other proteins in the MST scaffold. For this purpose, we compare an interaction (u, v) in the original network to the best indirect interaction between u and v in the current scaffold network. If the edge weight is at least as high as a factor α times the weight of the best indirect interaction, the interaction is added to the MST network. The resulting network is denoted as $eMST_\alpha$ and generally, α is set to 1.

For calculating the best indirect interaction we use the fact that all edge weights are confidence values in $[0, 1]$ and, thus, can be interpreted as probabilities. The weight of an

indirect interaction is the probability of the optimal path between the corresponding proteins in the current scaffold (without the edge (u, v)). The probability is calculated as the product of the edge probabilities on this path and the optimal path is defined as the path with the highest probability. If we transform edge weights by taking the absolute values of the logarithms, the path with maximum probability is the path with the smallest sum of transformed edge weights. This optimal path between a pair of nodes can then be calculated using Dijkstra's algorithm for shortest paths [CLRS00].

To identify interactions which cannot be explained by a sequence of sufficiently strong indirect interactions, we process candidate interactions in the order of non-increasing edge weights. For each interaction e, we calculate the optimal alternative path P between the corresponding proteins in the current scaffold. The interaction e is added to the scaffold if $w(e) \geq \alpha w(P)$ and the scaffold is updated whenever a new interaction is identified. In the following, we show that this algorithm is correct for $\alpha \leq 1$. This means that there is no edge e in the final scaffold such that an alternative path P exists in the network with $w(e) < \alpha w(P)$.

Proof by contradiction: Assume, there exists such a path P for an edge e. Since the weight of each edge is ≤ 1, we have for each edge $f \in P$ that $w(P) \leq w(f)$. Thus, $w(e) < \alpha w(f) \, \forall f \in P$ and $w(e) < w(f) \, \forall f \in P$ if $\alpha \leq 1$. Thus, all edges on this path have been processed before e and this path was already contained in the network at the time e is added. This is a contradiction to the construction of the scaffold network.

2.3 Baseline prediction algorithms

We compare our algorithm against two baseline predictors. The complete approach predicts all interactions within the complex as direct, physical interactions. The connected approach calculates the network G_τ for each complex where $\forall e \in E_\tau : w(e) \geq \tau$ and τ the largest value such that G_τ is connected.

3 Results

The MST and extended MST approaches were applied to interaction scores and complex predictions calculated from the combined results of the genome-scale TAP experiments of Gavin *et al.* [G$^+$06] and Krogan *et al.* [K$^+$06] in yeast. Here, we used confidence scores and protein complexes predicted with the unsupervised Bootstrap approach we presented recently [FKZ08]. These confidence scores are more accurate than any other scoring method. Furthermore, the medium (BT-409) and high confidence (BT-217) Bootstrap complexes are of the same quality as the best supervised predictions and manually curated protein complexes, respectively.

All bootstrap confidence scores are between 0 and 1 and the original network contains 62,876 interactions. By restricting this to interactions within BT-409 complexes (the complete approach), we obtained 9,918 interactions (15.8% of the original set). The connected

approach yielded 5,404 interactions (8.6%), the MST approach 1,658 interactions (2.6%) and the extended MST approach (with $\alpha = 1$) 3,085 interactions (4.9%).

3.1 Reference interactions

To compile a reference set of direct interactions we extracted all yeast protein-protein interactions from the DIP database [SMS$^+$04] determined with the Y2H method. We chose Y2H interactions to make sure that only direct physical interactions are contained in the reference set. Furthermore, we used the genome-scale Y2H results for yeast from the studies of Uetz et al. [U$^+$00] and Ito et al. [I$^+$01]. For the Ito dataset, both the high confidence and complete set were evaluated.

Since large Y2H interaction sets are also available for *Drosophila* [G$^+$03], *C. elegans* [L$^+$04] and human [S$^+$05, R$^+$05], we used orthology assignments from the Inparanoid database [BSOS08] to map these interactions onto yeast. Interactions were mapped if both interaction partners had orthologs in yeast. This resulted in 575 predicted interactions from *Drosophila* to yeast, 170 from *C. elegans* to yeast (70 from the core set defined by Li et al.) and 220 predicted interactions from human to yeast.

Table 1 shows a comparison of the Y2H interaction networks against the BT-409 and BT-217 complexes and manually curated complexes from the MIPS database [M$^+$04]. The first row for each combination indicates the enrichment of Y2H interactions within complexes. Enrichment is calculated as $p_C/p_{\overline{C}}$ where

$$p_C = \frac{|E_C| \cap |E_{Y2H}|}{|E_C|} \quad \text{and} \quad p_{\overline{C}} = \frac{|E_{\overline{C}}| \cap |E_{Y2H}|}{|E_{\overline{C}}|}. \tag{1}$$

Here, E_C is the set of interactions within complexes, $E_{\overline{C}}$ the set of interactions between proteins in different complexes and E_{Y2H} the set of Y2H interactions. The second row of Table 1 specifies the fraction of Y2H interactions contained within complexes.

Complex set	**Y2H interaction network**					
	DIP	**Uetz**	**Ito**	**Ito core**	**All yeast**	**Yeast + Pred.**
MIPS	53.0 [0.06]	106.9 [0.07]	33.9 [0.03]	195.6 [0.09]	52.1 [0.05]	41.4 [0.05]
BT-409	64.5 [0.07]	152.4 [0.14]	53.1 [0.05]	192.7 [0.18]	64.2 [0.07]	59.8 [0.07]
BT-217	75.1 [0.05]	150.9 [0.1]	65.1 [0.03]	205.2 [0.12]	75.2 [0.05]	66.8 [0.05]

Table 1: This table shows for each Y2H network the enrichment (see equation 1) of Y2H interactions within the MIPS, BT-409 and BT-217 complexes. The second row for each combination of network and complex set specifies the fraction of Y2H interactions within protein complexes.

As can be seen, Y2H interactions are significantly enriched within protein complexes and the enrichment values appear to reflect at least partly the confidence of the corresponding Y2H set. The Ito core interactions have much higher enrichment values than the less confident complete Ito set. When adding the less confident predicted interactions to the complete yeast network a less distinctive but still considerable decrease in enrichment can be observed. Interestingly, the enrichment is significantly higher in the Bootstrap complexes than in the MIPS complexes. The average bootstrap score of Y2H interactions in complexes (0.76) is significantly higher than for interactions in complexes not confirmed by Y2H (0.44). Unfortunately, the fraction of interactions in the Y2H network which actually connect proteins in the same complex is very small.

3.2 Assessing the predictive accuracy of complex scaffolds

We evaluated the predictive accuracy of the presented methods using *receiver operating characteristic* (ROC) curves [Faw06]. For this purpose, true positive rates are plotted against false positive rates with decreasing thresholds for predicting a direct interaction. In this case, true positive rate is the fraction of Y2H interactions within the BT-409 complexes recovered by the prediction methods. False positive rate is the fraction of interactions within the BT-409 complexes predicted to be in the scaffold but not contained in the Y2H network.

Figure 2 **A** shows the ROC curve for the complete, connected, MST and extended MST predictions compared against the complete set of yeast Y2H interactions. Similar results can be observed for all Y2H sets. As can be clearly seen, significant improvements in predictive accuracy can be obtained with the MST approach. At a maximum true positive

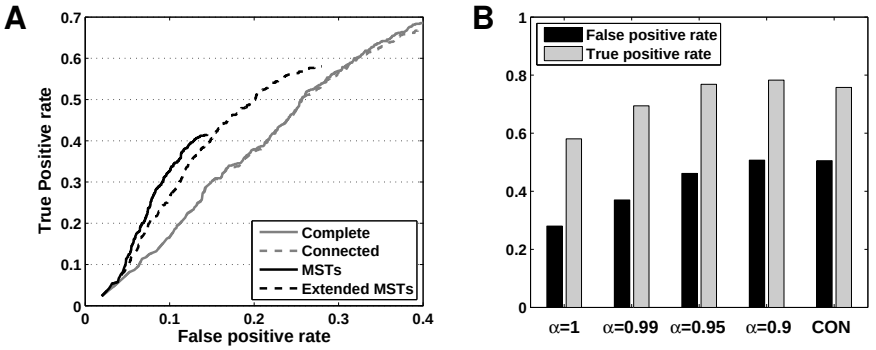


Figure 2: ROC curve (**A**) for the direct interactions predicted by the complete, connected, MST and extended MST (for $\alpha = 1$) approach compared to all yeast Y2H interactions within the BT-409 complexes. Here, the curves for the complete and connected approach are almost identical in this range as 90% of the top scoring interactions of the complete network are also contained in the connected network. The two networks differ mostly in the low scoring and low quality interactions contained additionally in the complete network. Figure **B** illustrates true positive and false positive rates for decreasing values of α and the connected networks (CON).

rate of 41.6%, only 14.5% false positives are predicted. At the same true positive rate, about 22% false positives are predicted by the complete and connected predictions.

However, the higher specificity of the MST approach results in a significantly lower sensitivity. Thus, less than half of the Y2H interactions recovered by the baseline predictions are recovered by the MST approach. By extending the MSTs, the fraction of true positives identified can be increased significantly. Although the false positive rate consequently increases as well, the overall performance of the extended MSTs is nevertheless significantly better than observed for the baseline predictions.

Figure 2 **B** illustrates the true and false positive rates for decreasing values of α used for extending the MSTs. The more conditions are relaxed for extending the networks, the more interactions are added. As a consequence, more true interactions are recovered but also more wrong predictions are made. Nevertheless, at the same false positive rate the extended MSTs can recover more true positives than the connected networks.

3.3 Separation of substructures within complexes

By predicting the topology of protein complexes, we aim to identify substructures within complexes. Proteins which are closely involved with each other should end up very close to each other in the network, i.e. separated by only few interactions. Proteins more distantly related, on the other hand, should be separated by many interactions within the network. To measure the distance of two proteins in the network we calculate the number of interactions on the shortest (unweighted) path between them.

Figure 3 **A** illustrates the correlation between the distance of two proteins and the confi-

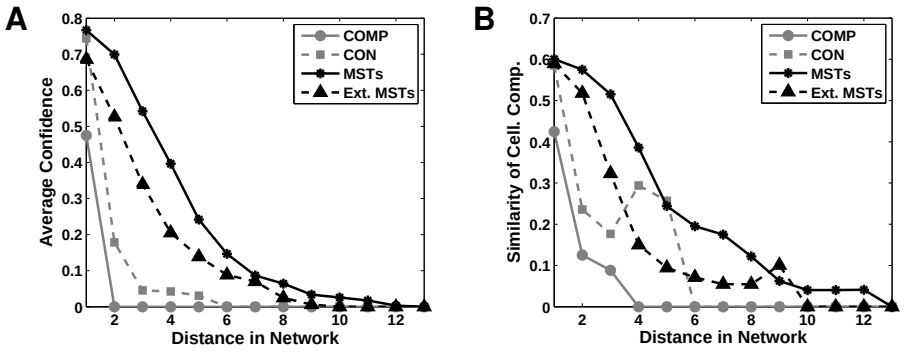


Figure 3: Figure **A** compares the distance between a protein pair in the complete (COMP), connected (CON), MST and extended MST networks against the confidence of this pairwise interaction in the complete network. Averages are taken over all protein pairs with the same distance. Figure **B** compares the distance in the network against the fraction of GO cellular compartment annotations the two protein have in common.

dence of the corresponding interaction in the original bootstrap network. As expected, the more interactions the predicted network contains the shorter are the distances in the network. Thus, largest distances are observed in the MST network and shortest distances in the complete network where most proteins are directly connected. Furthermore, the higher the confidence of an interaction between two proteins is, the smaller is the distance in the resulting scaffold network. Due to the larger distances in the MST and extended MST networks, the rate of decrease is significantly smaller for these networks which allows for a better resolution of the network structure.

We calculated for each pair of proteins the fraction of Gene Ontology (GO) [A$^+$00] annotations they have in common and correlated this with their distance. We used this simple measure instead of more complicated methods as semantic similarity [SDRL06] because semantic similarity within complexes is generally very high [FKZ08]. In this case, the simple overlap measure allows for a more fine-grained analysis of protein function and localization within complexes.

Figure 3**B** shows the results for the cellular component ontology of the GO. Similar results were observed for the biological process and molecular function ontologies. As with interaction confidences, we observe that the similarity of the cellular component assignments tends to decrease with the distance between the corresponding proteins. Furthermore, the rate of decrease is lowest for the sparse MST network. In the extended MSTs, this rate is significantly higher but still by far not as high as in the connected and complete networks.

This indicates that proteins involved in different subcomponents of a complex are separated from each other by many interactions in the predicted scaffolds, whereas proteins involved in the same subcomponents are close to each other. Surprisingly, co-localization scores increase again at a distance of 4 for the connected network and at a distance of 9 for the extended MST network. This is due to the small number of protein pairs with this distance in the corresponding networks. Thus, outliers affect the average co-localization more strongly.

3.4 Analysis of the DNA-directed RNA polymerase complex

To illustrate the value of the predicted interaction scaffolds for the identification of substructures in complexes, we analyzed the DNA-directed RNA polymerase complex. This complex contains 46 proteins in the BT-409 set and effectively consists of three separate RNA polymerase complexes (RNA polymerase I, II and III) which have been clustered into one complex since they have many proteins in common. The crystal structure of polymerase II is known, whereas only little structural information is available for polymerases I and III [CAB$^+$08].

Figure 4 shows the complex connections for the RNA polymerase in the connected and extended MST network. The complex was visualized using the organic layout function of Cytoscape [SMO$^+$03] which clusters closely connected proteins together. In the complete bootstrap network, no substructure can be observed but all proteins form a tight cluster. In the connected network (see Figure 4 **A**), we observe at least a separation between the

RNA polymerase III complex and the remaining proteins but polymerase I and II are too tightly connected to identify the substructure. It is only when proteins are colored by their cellular components that we detect that proteins from the same subcomponents are clustered together.

In the extended MST network (see Figure 4 **B**) the subdivision of the complex into polymerase complexes I, II and III can be clearly observed. The polymerase III complex (light grey) is connected by two proteins (RPC19, RPC5) to the polymerase I complex (dark grey). The latter one is then connected to the polymerase II complex (black) by a group of five proteins (RPB5, RPB6, RPB8, RPB10 and RPB12) contained in all three RNA polymerase complexes.

Interestingly, these five proteins are not directly connected to the other polymerase III proteins although they are contained in this complex. If we relax the criterion for extending an MST ($\alpha = 0.99$), the interaction between RPB10 and RPC5, which has also been identified in Y2H screens [LCST93, FBG$^+$99], is added to the scaffold. This might suggest that the interaction of the common proteins to polymerase III is mediated via this interaction. However, if we look at the crystal structure of polymerase II and the model for polymerase III [CAB$^+$08], we find that none of the common proteins are actually in physical contact in the complexes (possibly apart from RPB10 and RPB12).

Going back to the original purification experiments, we find that of the 7 interactions predicted between the common proteins, 6 interactions are bait-prey interactions which have been found to be very reliable [BCRC04] and 3 of those are identified in both directions. Since the proteins do not appear to physically interact, this is probably a consequence of the common occurrence of these proteins in several different complexes.

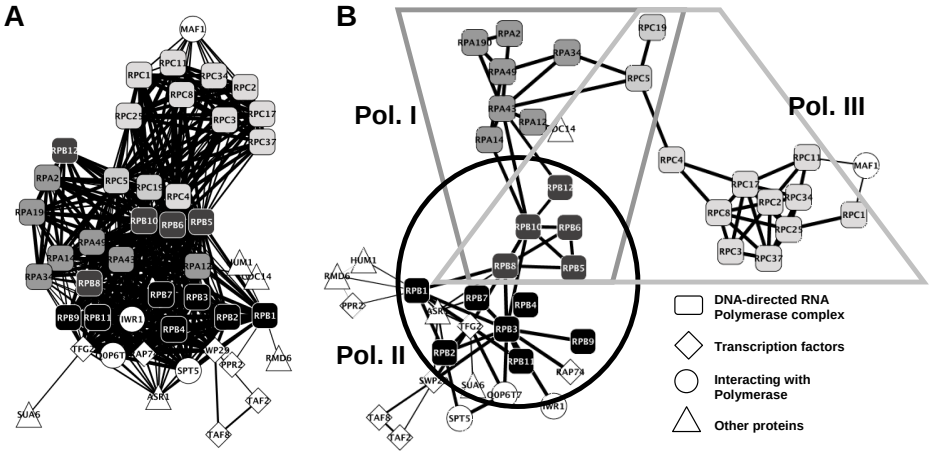


Figure 4: This figure shows the predicted subnetworks for the DNA-directed RNA complex. Figure **A** shows the results for the connected subnetwork and **B** the results for extended MST network. Colors indicate the subcomponents: Polymerase complexes I (dark grey), II (black) and III (light grey). Rectangles denote polymerase proteins and diamonds transcription factors.

Another example which illustrates the problems of affinity purification in distinguishing the actual physical interactions, is the interaction between the RPB3 and RPB9 protein in the polymerase II subcomplex. Although the two proteins are located at different ends of the Polymerase II in the 3D structure, this interaction has a very high confidence score as RPB3 and RPB9 co-purified each other whenever one of these proteins was used as bait (6 times for RPB3 and 5 times for RPB9).

4 Discussion

In this article, we presented an approach for predicting the topology of protein complexes, i.e. the scaffold of direct interactions which spans the complex. First, our method calculates the union of all maximum spanning trees (MSTs) in the interaction score network for a protein complex. In a subsequent step, this network is iteratively extended by interactions which cannot be explained by a path of alternative indirect interactions. The MST approach is applicable to all weighted interaction networks and in particular to interaction scores calculated from affinity purification assays with any of the recently published scoring methods. Confidence scores which are required for extending the MSTs in our algorithm, can be obtained by scaling any type of scores to $[0, 1]$ or using the Bootstrap approach we developed to calculate scores from affinity purification experiments.

Predictive performance of subnetworks calculated from Bootstrap confidence scores was evaluated on experimentally determined direct, physical interactions from Y2H experiments. We showed that predictive accuracy can be increased significantly with our approach compared to baseline predictions. When comparing the individual protein complexes to the Y2H network, we observed that less than half of the complexes both in the predicted complex set and in manually curated complexes contain at least one Y2H interaction, and only 5% to 11% of the complexes are actually non-trivially connected (i.e. they are connected and contain more than two proteins) in the Y2H network. This suggests that many of the direct interactions within complexes have not been identified yet. Here, the interactions predicted by our approach but not found in the Y2H network are promising starting points for experimental verification.

Protein complexes are not simply clumps of proteins but they have an internal substructure in which not all proteins bind closely together. Thus, proteins in the same subcomplex are closely connected by short paths of direct interactions whereas proteins in different subcomponents are separated by many physical interactions in this network. Our results show that both for the network predicted with our approach and for the baseline predictors, the distance between protein pairs is correlated strongly with the corresponding interaction confidence and the similarity of the cellular components these proteins are contained in. However, in the scaffold network predicted by our method, separation of proteins in different subcompartments of a complex is more distinctive and thus, the substructure of the complex can be better resolved.

We illustrated this observation on the complex of DNA-directed RNA polymerases. While the substructure of the complex with three different RNA polymerases can only be partly

observed in the baseline predictions, it is clearly evident in the network predicted with our approach. By relaxing the conditions of our algorithm slightly, the substructure of the complex can be further emphasized and important interactions can be identified. Thus, the algorithm presented in this article is valuable for identifying the scaffold of physical protein interactions within complexes as well as their subcomponent structure.

References

[A+00] M. Ashburner et al. Gene Ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat Genet*, 25(1):25–29, May 2000.

[A+04] Patrick Aloy et al. Structure-based assembly of protein complexes in yeast. *Science*, 303(5666):2026–2029, Mar 2004.

[BCRC04] Joel S Bader, Amitabha Chaudhuri, Jonathan M Rothberg, and John Chant. Gaining confidence in high-throughput protein interaction networks. *Nat Biotechnol*, 22:78–85, Jan 2004.

[BSOS08] Ann-Charlotte Berglund, Erik Sjölund, Gabriel Ostlund, and Erik L L Sonnhammer. InParanoid 6: eukaryotic ortholog clusters with inparalogs. *Nucleic Acids Res*, 36(Database issue):D263–D266, Jan 2008.

[BVH07] Allister Bernard, David S. Vaughn, and Alexander J. Hartemink. Reconstructing the Topology of Protein Complexes. In *Proceedings of the 11th Annual International Conference on Research in Computational Molecular Biology, RECOMB 2007, Oakland, CA, USA, April 21-25*, pages 32–46, 2007.

[C+07] Sean R Collins et al. Toward a comprehensive atlas of the physical interactome of Saccharomyces cerevisiae. *Mol Cell Proteomics*, 6(3):439–450, Mar 2007.

[CAB+08] P. Cramer, K.-J. Armache, S. Baumli, S. Benkert, F. Brueckner, C. Buchen, G.E. Damsma, S. Dengl, S.R. Geiger, A.J. Jasiak, A. Jawhari, S. Jennebach, T. Kamenski, H. Kettenberger, C.-D. Kuhn, E. Lehmann, K. Leike, J.F. Sydow, and A. Vannini. Structure of Eukaryotic RNA Polymerases. *Annual Review of Biophysics*, 37(1):337–352, 2008.

[CLRS00] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms, 2nd edition*. MIT Press, McGraw-Hill Book Company, 2000.

[Faw06] Tom Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, June 2006.

[FBG+99] A. Flores, J. F. Briand, O. Gadal, J. C. Andrau, L. Rubbi, V. Van Mullem, C. Boschiero, M. Goussot, C. Marck, C. Carles, P. Thuriaux, A. Sentenac, and M. Werner. A protein-protein interaction map of yeast RNA polymerase III. *Proc Natl Acad Sci U S A*, 96(14):7815–7820, Jul 1999.

[FKZ08] Caroline C. Friedel, Jan Krumsiek, and Ralf Zimmer. Bootstrapping the Interactome: Unsupervised Identification of Protein Complexes in Yeast. In *Proceedings of the 12th Annual International Conference on Research in Computational Molecular Biology, RECOMB 2008, Singapore, March 30 - April 2*, pages 3–16, 2008.

[FS89] S. Fields and O. Song. A novel genetic system to detect protein-protein interactions. *Nature*, 340(6230):245–246, Jul 1989.

[G+03] L. Giot et al. A protein interaction map of Drosophila melanogaster. *Science*, 302:1727–36, Dec 2003.

[G+06] Anne-Claude Gavin et al. Proteome survey reveals modularity of the yeast cell machinery. *Nature*, 440:631–6, Mar 2006.

[HBW05] Jens Hollunder, Andreas Beyer, and Thomas Wilhelm. Identification and characterization of protein subcomplexes in yeast. *Proteomics*, 5(8):2082–2089, May 2005.

[HBW07] Jens Hollunder, Andreas Beyer, and Thomas Wilhelm. Protein subcomplexes–molecular machines with highly specialized functions. *IEEE Trans Nanobioscience*, 6(1):86–93, Mar 2007.

[HFB+07] Jens Hollunder, Maik Friedel, Andreas Beyer, Christopher T Workman, and Thomas Wilhelm. DASS: efficient discovery and p-value calculation of substructures in unordered data. *Bioinformatics*, 23(1):77–83, Jan 2007.

[HLM07] G. Traver Hart, Insuk Lee, and Edward Marcotte. A high-accuracy consensus map of yeast protein complexes reveals modular nature of gene essentiality. *BMC Bioinformatics*, 8(1):236, Jul 2007.

[I+01] T. Ito et al. A comprehensive two-hybrid analysis to explore the yeast protein interactome. *Proc Natl Acad Sci USA*, 98:4569–74, Apr 2001.

[K+06] Nevan J Krogan et al. Global landscape of protein complexes in the yeast Saccharomyces cerevisiae. *Nature*, 440:637–43, Mar 2006.

[Kru56] J. B Kruskal. On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem. *Proc. Amer. Math. Soc.*, 7:48–50, 1956.

[L+04] Siming Li et al. A map of the interactome network of the metazoan C. elegans. *Science*, 303:540–3, Jan 2004.

[LCST93] D. Lalo, C. Carles, A. Sentenac, and P. Thuriaux. Interactions between three common subunits of yeast RNA polymerases I and III. *Proc Natl Acad Sci U S A*, 90(12):5524–5528, Jun 1993.

[M+04] H. W. Mewes et al. MIPS: analysis and annotation of proteins from whole genomes. *Nucleic Acids Res*, 32(Database issue):D41–D44, Jan 2004.

[Pri57] R. C. Prim. Shortest connection networks and some generalisations. *Bell System Technical Journal*, 36:1389–1401, 1957.

[PVE+07] Shuye Pu, Jim Vlasblom, Andrew Emili, Jack Greenblatt, and Shoshana J Wodak. Identifying functional modules in the physical interactome of Saccharomyces cerevisiae. *Proteomics*, 7(6):944–960, Mar 2007.

[R+05] Jean-François Rual et al. Towards a proteome-scale map of the human protein-protein interaction network. *Nature*, 437:1173–8, Oct 2005.

[RSR+99] G. Rigaut, A. Shevchenko, B. Rutz, M. Wilm, M. Mann, and B. Sraphin. A generic protein purification method for protein complex characterization and proteome exploration. *Nat Biotechnol*, 17(10):1030–1032, Oct 1999.

[S+05] Ulrich Stelzl et al. A human protein-protein interaction network: a resource for annotating the proteome. *Cell*, 122:957–68, Sep 2005.

[SDRL06] Andreas Schlicker, Francisco S Domingues, Jörg Rahnenführer, and Thomas Lengauer. A new measure for functional similarity of gene products based on Gene Ontology. *BMC Bioinformatics*, 7:302, 2006.

[SMO+03] Paul Shannon, Andrew Markiel, Owen Ozier, Nitin S Baliga, Jonathan T Wang, Daniel Ramage, Nada Amin, Benno Schwikowski, and Trey Ideker. Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res*, 13(11):2498–2504, Nov 2003.

[SMS+04] Lukasz Salwinski, Christopher S Miller, Adam J Smith, Frank K Pettit, James U Bowie, and David Eisenberg. The Database of Interacting Proteins: 2004 update. *Nucleic Acids Res*, 32(Database issue):D449–D451, Jan 2004.

[SVG05] Denise Scholtens, Marc Vidal, and Robert Gentleman. Local modeling of global interactome networks. *Bioinformatics*, 21(17):3548–3557, Sep 2005.

[U+00] P. Uetz et al. A comprehensive analysis of protein-protein interactions in Saccharomyces cerevisiae. *Nature*, 403:623–7, Feb 2000.

Evolutionary Construction of Multiple Graph Alignments for the Structural Analysis of Biomolecules

Thomas Fober, Eyke Hüllermeier, Marco Mernberger
FB Mathematik/Informatik, Philipps-Universität Marburg

Abstract: The concept of multiple graph alignment has recently been introduced as a novel method for the structural analysis of biomolecules. Using inexact, approximate graph-matching techniques, this method enables the robust identification of approximately conserved patterns in biologically related structures. In particular, multiple graph alignments enable the characterization of functional protein families independent of sequence or fold homology. This paper first recalls the concept of multiple graph alignment and then addresses the problem of computing optimal alignments from an algorithmic point of view. In this regard, a method from the field of evolutionary algorithms is proposed and empirically compared to a hitherto existing greedy strategy. Empirically, it is shown that the former yields significantly better results than the latter, albeit at the cost of an increased runtime.

1 Introduction

Focusing on the identification of *structural* similarities of biomolecules, this paper presents the concept of *multiple graph alignment* (MGA) as a structural counterpart to sequence alignment. As opposed to homology-based methods, this approach allows one to capture non-homologous molecules with similar functions as well as evolutionary conserved functional domains. Our special interest concerns the analysis of protein structures or, more specifically, protein binding sites, even though graph alignments can also be used for analyzing other types of biomolecules.

The problem of comparing graphs occurs in many applications and, correspondingly, has been studied in different research fields, including pattern recognition [5], network analysis [2] and kernel-based machine learning [6, 4]. These approaches, however, almost exclusively focus on the comparison of two graphs, while our method, in analogy to multiple sequence alignment, seeks to analyze multiple graphs simultaneously. Moreover, most existing approaches target on exact matches between graphs or parts thereof, often resorting to the concept of subgraph isomorphism [8].

This work draws on [10], in which the concept of MGA was first introduced. That paper proposed an algorithm which employs a simple greedy strategy to construct MGAs in an incremental way. Here, we present an alternative method using evolutionary algorithms. As will be shown experimentally, significant improvements in terms of the quality of alignments can thus be achieved, albeit at the cost of an increased runtime.

The paper is organized as follows: Subsequent to a brief introduction to graph-based mo-

deling of protein binding sites in Section 2, we introduce the concept of a multiple graph alignment in Section 3. The problem of computing an MGA is then addressed in Section 4, where an evolutionary algorithm is proposed for this purpose. Section 5 is devoted to the experimental validation of the approach, and Section 6 concludes the paper.

2 Graph-Based Modeling of Protein Binding Sites

In bio- and chemoinformatics, single biomolecules are often modeled at an abstract level in terms of a graph G consisting of a set of (labeled) nodes V and (weighted) edges E. In this paper, our special interest concerns the modeling of protein binding pockets. More specifically, our work builds upon Cavbase [9], a database system for the automated detection, extraction, and storing of protein cavities (hypothetical binding pockets) from experimentally determined protein structures (available through the PDB). In Cavbase, graphs are used as a first approximation to describe binding pockets. The database currently contains $113,718$ hypothetical binding pockets that have been extracted from $23,780$ publicly available protein structures using the LIGSITE algorithm [7].

To model a binding pocket as a graph, the geometrical arrangement of the pocket and its physicochemical properties are first represented by predefined *pseudocenters* – spatial points that represent the center of a particular property. The type and the spatial position of the centers depend on the amino acids that border the binding pocket and expose their functional groups. They are derived from the protein structure using a set of predefined rules [9]. As possible types for pseudocenters, hydrogen-bond donor, acceptor, mixed donor/acceptor, hydrophobic aliphatic, metal ion, pi (accounts for the ability to form π–π interactions) and aromatic properties are considered. Pseudocenters can be regarded as a compressed representation of areas on the cavity surface where certain protein-ligand interactions are experienced. Consequently, a set of pseudocenters is an approximate representation of a spatial distribution of physicochemical properties.

The assigned pseudocenters form the nodes $v \in V$ of the graph representation, and their properties are modeled in terms of node labels $\ell(v) \in \{\texttt{P1}, \texttt{P2} \ldots \texttt{P7}\}$, where `P1` stands for donor, `P2` for acceptor, etc. Two centers are connected by an edge in the graph representation if their Euclidean distance is below a certain threshold and each edge $e \in E$ is labeled with the respective distance $w(e) \in \mathbb{R}$.[1] The edges of the graph thus represent geometrical constraints among points on the protein surface.

3 Multiple Graph Alignment

When comparing protein cavities on a structural level, one has to deal with the same mutations that also occur on the sequence level. Corresponding mutations, in conjunction with conformational variability, strongly affect the spatial structure of a binding site as well as

[1] An interaction distance of 11.0 Å is typically enough to capture the geometry of a binding site, and ignoring larger distances strongly simplifies the graph representation and hence accelerates the fitness calculation.

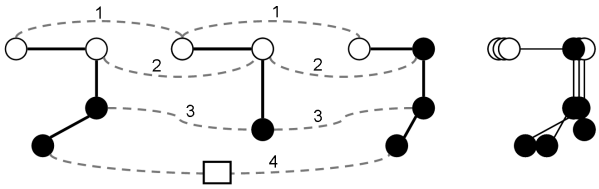


Figure 1: Simple illustration of MGA by an approximate match of three graphs with two types of labels (black and white). Mutual assignments of nodes are indicated by dashed lines. Note that the second assignment involves a mismatch, since the node in the third graph is black. Likewise, the fourth assignment involves a dummy (indicated by a box), since a node is missing in the second graph. The rightmost picture is a graphical overlay of the three structures.

its physicochemical properties and, therefore, its graph descriptor. This is even more an issue when it comes to the comparison of proteins that might share a common function but lack a close hereditary relationship. Thus, one cannot expect that the graph descriptors for two functionally related binding pockets match exactly. Our approach therefore includes the following types of edit operations to account for differences between a graph $G_1(V_1, E_1)$ and another graph $G_2(V_2, E_2)$. **Insertion or deletion** of a node $v_1 \in V_1$: A pseudocenter can be deleted or introduced due to a mutation in sequence space. Alternatively, an insertion or deletion in the graph descriptor can result from a conformational difference that affects the exposure of a functional group toward the binding pocket. **Label mismatch**, i.e., a change of the label $\ell(v_1)$ of a node $v_1 \in V_1$: The assigned physicochemical property of a pseudocenter can change if a mutation replaces a certain functional group by another type of group at the same position. **Node mismatch**, i.e., a change of the weight $w(e_1)$ of an edge $e_1 \in E_1$: The distance between two pseudocenters can change due to conformational differences.

By assigning a cost value to each of these edit operations, it becomes possible to define an edit distance for a pair of graph descriptors. The edit distance of two graphs G_1, G_2 is defined as the cost of a cost-minimal sequence of edit operations that transforms graph G_1 into G_2. As in sequence analysis, this allows for defining the concept of an alignment of two (or more) graphs. The latter, however, also requires the possibility to use dummy nodes $\perp$ that serve as placeholders for deleted nodes. They correspond to the gaps in sequence alignment (cf. Fig. 1).

Let $\mathcal{G} = \{G_1(V_1, E_1) \dots G_m(V_m, E_m)\}$ be a set of node-labeled and edge-weighted graphs. Then

$$\mathcal{A} \subseteq (V_1 \cup \{\perp\}) \times \cdots \times (V_m \cup \{\perp\})$$

is an alignment of the graphs in $\mathcal{G}$ if and only if the following two properties hold: (i) for all $i = 1 \dots m$ and for each $v \in V_i$ there exists exactly one $a = (a_1 \dots a_m) \in \mathcal{A}$ such that $v = a_i$ (i.e., each node of each graph occurs exactly once in the alignment). (ii) For each $a = (a_1 \dots a_m) \in \mathcal{A}$ there exists at least one $1 \leq i \leq m$ such that $a_i \neq \perp$ (i.e., each tuple of the alignment contains at least one non-dummy node).

Each $a \in \mathcal{A}$ corresponds to a vector of mutually assigned nodes from the graphs $G_1 \dots G_n$.

Note that, by matching nodes, a mutual assignment of edges is determined in an implicit way. To assess the quality of a given alignment, a scoring function is is used that corresponds to the above-mentioned edit distance, as each graph alignment defines a set of edit operations that have to be performed to transform one of the aligned graphs into another entry of the alignment. Our scoring function follows a sum-of-pairs scheme, i.e., the score s of a multiple alignment $\mathcal{A} = (a^1 \dots a^m)$ is defined by the sum of scores of all induced pairwise alignments:

$$s(\mathcal{A}) = \sum_{i=1}^{n} \mathrm{ns}(a^i) + \sum_{1 \le i < j \le n} \mathrm{es}(a^i, a^j), \tag{1}$$

where the *node score* (ns) is given by

$$\mathrm{ns}\begin{pmatrix} a_1^i \\ \vdots \\ a_m^i \end{pmatrix} = \sum_{1 \le j < k \le m} \begin{cases} \mathrm{ns}_m & \ell(a_j^i) = \ell(a_k^i) \\ \mathrm{ns}_{mm} & \ell(a_j^i) \neq \ell(a_k^i) \\ \mathrm{ns}_{dummy} & a_j^i = \perp, a_k^i \neq \perp \\ \mathrm{ns}_{dummy} & a_j^i \neq \perp, a_k^i = \perp \end{cases}$$

Comparing two edges is somewhat more difficult than comparing two nodes, as one cannot expect to observe edges of exactly the same lengths. We consider two edges as a match if their respective lengths, a and b, differ by at most a given threshold ϵ, and as a mismatch otherwise. The *edge score* (es) is then given by

$$\mathrm{es}\left(\begin{pmatrix} a_1^i \\ \vdots \\ a_m^i \end{pmatrix}, \begin{pmatrix} a_1^j \\ \vdots \\ a_m^j \end{pmatrix}\right) = \sum_{1 \le k < l \le m} \begin{cases} \mathrm{es}_{mm} & (a_k^i, a_k^j) \in E_k,\ (a_l^i, a_l^j) \notin E_l \\ \mathrm{es}_{mm} & (a_k^i, a_k^j) \notin E_k,\ (a_l^i, a_l^j) \in E_l \\ \mathrm{es}_m & d_{kl}^{ij} \le \epsilon \\ \mathrm{es}_{mm} & d_{kl}^{ij} > \epsilon \end{cases}$$

where $d_{kl}^{ij} = \|w(a_k^i, a_k^j) - w(a_l^i, a_l^j)\|$. The parameters (i.e., ns_m, ns_{mm}, ns_{dummy}, es_m, es_{mm}) are constants used to reward or penalize matches, mismatches and dummies, respectively. Throughout our experiments in Section 5, we used the parameters recommended in [10]: $\mathrm{ns}_m = 1$, $\mathrm{ns}_{mm} = -5$, $\mathrm{ns}_d = -2.5$, $\mathrm{es}_m = 0.2$, $\mathrm{es}_{mm} = -0.1$, $\epsilon = 0.2$.

The problem of calculating an optimal MGA, that is, an alignment with maximal score for a given set of graphs, is provably NP-complete. In [10], simple and effective heuristics for the MGA problem have been devised that were found to be useful for the problem instances that were examined. The main idea of these methods is to reduce the multiple alignment problem to the problem of pairwise alignment (i.e., calculating an optimal graph alignment for only two graphs) in a first step. Resorting to the idea of star-alignment, which is well-known in sequence analysis, these pairwise alignments are subsequently merged into a multiple alignment.

In this paper, we elaborate on the use of evolutionary algorithms as an alternative approach. On the one hand, evolutionary optimization is of course more expensive from a computational point of view. On the other hand, the hope is that this approach will be able to improve the solution quality, i.e., to produce alignments that are better than those obtained by the simple greedy strategy.

4 An Evolutionary Algorithm for Multiple Graph Alignment

An *evolution strategy* is a special type of evolutionary algorithm (EA) that seeks to optimize a *fitness function*, which in our case is given by the sum-of-pairs score (1). To this end, it simulates the evolution process by repeatedly executing the following loop [3]: (i) Initially, a population consisting of μ individuals, each representing a candidate solution, is generated at random; μ specifies the *population size* per generation. (ii) In each generation, $\lambda = \nu \cdot \mu$ offspring individuals are created; the parameter ν is called *selective pressure*. To generate a single offspring, the mating-selection operator chooses ρ parent individuals at random and submits them to the *recombination* operator. This operator generates an offspring by exchanging the genetic information of these individuals. The new individual is further modified by the *mutation* operator. (iii) The offsprings are evaluated and added to the parent population. Among the individuals in this temporary population T, the *selection* operator chooses the best μ candidates, which then form the population of the next generation. (iv) The whole procedure is repeated until a stopping criterion is met.

4.1 Representation of Individuals

Regarding the representation of individuals, note that in our case candidate solutions correspond to MGAs. Given a fixed numbering of the nodes of graph G_i from 1 to $|V_i|$ (not to be confused with the labeling), an MGA can be represented in a unique way by a two-dimensional matrix, where the rows correspond to the graphs and the columns to the aligned nodes (possibly a dummy, indicated by the number 0) of these graphs.

In the course of optimizing an MGA, the graphs can become larger due to the insertion of dummy nodes. For the matrix representation, this means that the number of columns is in principle not known and can only be upper-bounded by $n_1 + \ldots + n_m$, where $n_i = |V_i|$. This, however, will usually be too large a number and may come along with an excessive increase of the search space. From an optimization point of view, a small number of columns is hence preferable. On the other hand, by fixing a too small length of the alignment, flexibility is lost and the optimal solution is possibly excluded.

To avoid these problems, we make use of an *adaptive* representation: Starting with a single extra column filled with dummies, more such columns can be added if required or, when becoming obsolete, again be removed (see below). Thus, our matrix scheme is initialized with m rows and $n + 1$ columns, where $n = \max\{n_1, n_2 \ldots n_m\}$. For each graph G_i, a permutation of its nodes is then inserted, with dummies replacing the index positions $j > |V_i|$. As an aside, we note that dummy columns are of course excluded from scoring, i.e., the insertion or deletion of dummy columns has no influence on the fitness.

4.2 Evolutionary Operators

Among the proper selection operators for evolution strategies, the deterministic plus-selection, which selects the μ best individuals from the union of the μ parents and the λ offsprings, is most convenient for our purpose. In fact, since the search space of an MGA problem is extremely large, it would be very unfortunate to loose a current best solution. This excludes other selection techniques such as fitness-proportional or simulated annealing selection.

As we use a non-standard representation of individuals, namely a matrix scheme, the commonly used recombination and mutation operators are not applicable and have to be adapted correspondingly. Our recombination operator randomly selects ρ parent individuals from the current population (according to a uniform distribution). Then, $\rho - 1$ random numbers r_i, $i = 1 \ldots \rho - 1$ are generated, where $1 \leq r_1 < r_2 < \ldots < r_{\rho-1} < m$, and an offspring individual is constructed by combining the sub-matrices consisting, respectively, of the rows $\{r_{i-1} + 1 \ldots r_i\}$ from the i-th parent individual (where $r_0 = 0$ and $r_\rho = m$ by definition). Simply stitching together complete sub-matrices is not possible, however, since the nodes are not ordered in a uniform way. Therefore, the ordering of the first sub-matrix is used as a reference, i.e., the elements of the first row serve as pivot elements. General experience has shown that recombination increases the speed of convergence, and this was also confirmed by our experiments (see Section 5).

The mutation operator selects one row and two columns at random and swaps the entries in the corresponding cells. To enable large mutation steps, we have tried to repeat this procedure multiple times for each individual. As the optimal number of repetitions was unknown in the design phase of the algorithm, it was specified as a strategy component adjusted by a self-adaptation mechanism [3]. However, our experiments indicated that a simple mutation operator performing only single swaps solves the problem most effectively.

To adapt the length of an MGA (number of columns in the matrix scheme), it is checked in randomly chosen intervals whether further dummy columns are needed or existing ones have become unnecessary. Three cases can occur: (i) There exists exactly one dummy column, which means that the current length is still optimal. (ii) There is more than one dummy column: Apparently, a number of dummy columns are obsolete and can be removed, retaining only a single one. (iii) There is no dummy column left: The dummy column has been "consumed" by mapping dummies to real nodes. Therefore, a new dummy column has to be inserted.

4.3 Combining Evolutionary Optimization and Pairwise Decomposition

The search space of an MGA problem grows exponentially with the number of graphs, which is of course problematic from an optimization point of view. One established strategy to reduce complexity is to decompose a multiple alignment problem into several pairwise problems and to merge the solutions of these presumably more simple problems into

a complete solution. This strategy has already been exploited in the greedy approach, where the merging step has been realized by means of the star-alignment algorithm [10]. In star-alignment, a center structure is first determined, and this structure is aligned with each of the other $m-1$ structures. The $m-1$ pairwise alignments thus obtained are then merged by using the nodes of the center as pivot elements. As the quality of an MGA derived in this way critically depends on the choice of a suitable center structure, one often tries every structure as a center and takes the best result. In this case, all possible pairwise alignments are needed, which means that our evolutionary algorithm must be called $\frac{1}{2}(m^2-m)$ times.

As star-alignment is again a purely heuristic aggregation procedure, the gain in efficiency is likely to come along with a decrease in solution quality, compared with the original EA algorithm. This is not necessarily the case, however. In fact, a decomposition essentially produces two opposite effects, a positive one due to a simplification of the problem and, thereby, a reduction of the search space, and a negative one due to a potentially suboptimal aggregation of the partial solutions. For a concrete problem, it is not clear in advance which among these two effects will prevail. Roughly speaking, it is well possible that constructing good pairwise alignments and aggregating them in an ad-hoc way is better than getting astray in a huge search space of multiple alignments.

5 Experimental Results

In a first step, we adjusted the following exogenous parameters of our EA using the *sequential parameter optimization toolbox* (SPOT) [1] in combination with suitable synthetic data: μ, the population size; ν, the selective pressure; ρ, the recombination parameter; τ, the probability to check for dummy columns; `selfadaption`, which can assume values $\{\texttt{on}, \texttt{off}\}$, and enables or disables the automatic step size control; `initial step size`, which defines the initial step size for the mutation; if the automatic step size control is disabled, this parameter is ignored and a constant step size of 1 is used for the mutation.

After optimizing the parameters on diverse datasets, the following parameter configuration turned out to be well-suited for our problem class: $\mu = 4$, $\nu = 15$, `selfadaption` $=$ `off`, $\rho = 4$, $\tau = 0.35$. As can be seen, a small value for the population size (only large enough to enable recombination) is enough, probably due to the fact that local optima do not cause a severe problem. On the other hand, as the search space is extremely large, a high selective pressure is necessary to create offsprings with improved fitness. The self-adaptation mechanism is disabled and, hence, the mutation rate is set to one (only two cells are swapped by mutation). This appears reasonable, as most swaps do not yield an improvement and instead may even produce a deterioration, especially during the final phase of the optimization. Thus, an improvement obtained by swapping two cells is likely to be annulled by a second swap in the same individual. Finally, our experiments suggest that a recombination is very useful and should therefore be enabled. The probability τ is set to a relatively high value due to avoiding long times of stagnation because of an insufficient alignment length.

5.1 Mining Protein Binding Pockets

We examined the performance of our algorithms on a data set consisting of 74 structures derived from the Cavbase database. Each structure represents a protein cavity belonging to the protein family of thermolysin, bacterial proteases frequently used in structural protein analysis and annotated with the E.C. number 3.4.24.27 in the ENZYME classification database. The data set is suited for our purpose, as all cavities belong to the same enzyme family and, therefore, evolutionary related, highly conserved substructures ought to be present. On the other hand, with cavities (hypothetical binding pockets) ranging from about 30 to 90 pseudocenters and not all of them being real binding pockets, the data set is also diverse enough to present a real challenge for graph matching techniques.

We produced 100 graph alignments of size 2, 4, 8, 16 and 32, respectively, for randomly chosen structures, using the greedy heuristic (Greedy), our evolutionary algorithm with optimized parametrization (EA), and in combination with a star-alignment procedure (EA*). As a measure of comparison, we derived the relative improvement of the score (1),

$$\frac{s(\mathcal{A}') - s(\mathcal{A})}{\min\{|s(\mathcal{A}')|, |s(\mathcal{A})|\}} \ , \tag{2}$$

where $\mathcal{A}'$ and $\mathcal{A}$ denote, respectively, the alignment produced by EA (or EA*) and Greedy. This measure is positive if the EA (EA*) solution yields a higher score than the Greedy solution; e.g., a relative improvement of 1 would mean an increase in score by a factor of 2 (note that $s(\mathcal{A}) < 0$ is possible).

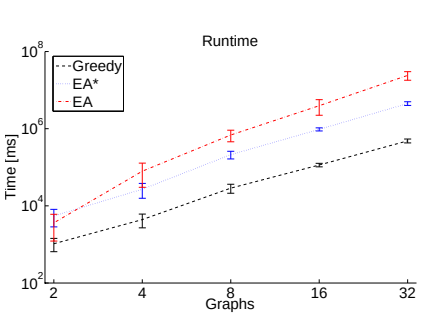
The results are summarized in Fig. 2. As can be seen, the EA solutions are never worse and often significantly better than the Greedy solutions. In terms of runtime,[2] it is clear that Greedy is still more efficient. Yet, a good compromise between solution quality and efficiency is achieved by EA*, as the runtime is much better than for EA, especially for a larger number of graphs.

5.2 Influence on Similarity Retrieval

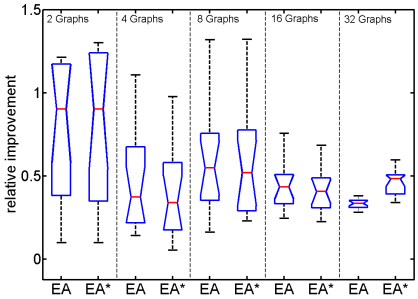
Pairwise similarity scores are often used to rank the objects stored in a database with respect to a given query object. For this purpose, the *absolute* similarity degrees are less important than the *relative* ones. Consequently, one may ask whether our EA, in addition to finding alignments with higher score, does actually yield rankings that differ from those produced by the Greedy algorithm. This is not self-evident since, for example, a constant improvement by a factor c, the same for each pairwise alignment, would not have any influence on a ranking.

Therefore, we compared 26 protein cavities belonging to the ClpP proteasome complex of *E. coli* with a set of 964 other cavities using EA and Greedy, respectively. Thus, we generated 2 sets of 25064 pairwise alignments and ranked the alignments according to

[2] Intel Core 2 Duo 2.4 GHz, 2 GB memory, Windows XP SP 2 operating system.



(a) Runtimes



(b) Relative improvements (boxplot)

Figure 2: Results of the first experiment: (a) Runtimes in milliseconds (mean and standard deviation) of greedy heuristic, EA using star alignment decomposition (EA*) and pure EA. (b) Relative improvements as defined in (2).

their score. We subsequently compared the generated rankings by computing the overlap of top-k ranks for both algorithms. This is done by calculating the intersection I of the top-k lists from the EA and the Greedy ranking. The results in terms of $k \mapsto f(k) = \frac{1}{k}|I|$ mappings are shown in Fig. 3. As one can see, the rankings produced by both algorithms significantly differ with respect to their top ranks. As indicated by a value $f(k) = 0$, most rankings (one ClpP cavity compared to 964 others) do not share any cavity in their top positions. In fact, there are only three rankings that share a few cavities in their top-10 lists. Although some curves appear to start increasing rather soon, one has to keep in mind that the real interest is most often focused on the top-positions only.

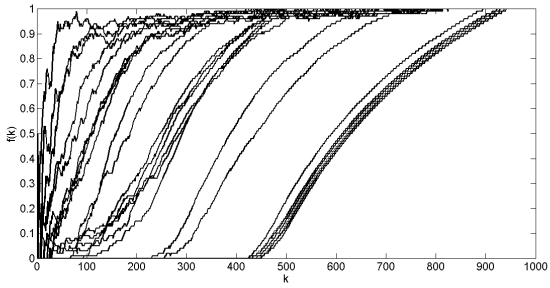


Figure 3: Top$_k$-Cuts showing the overlap in the top k ranks ($k = 1 \ldots 964$) for 26 protein cavities each compared to 964 other cavities.

6 Conclusions

Multiple graph alignment (MGA) has recently been introduced as a novel method for analyzing biomolecules on a structural level. Using robust, noise-tolerant graph matching techniques, MGA is able to discover approximately conserved patterns in a set of graph-descriptors representing a family of evolutionary related biological structures. As the computation of optimal alignments is a computationally complex problem, this paper has proposed an evolutionary algorithm (EA) as an alternative to an existing greedy strategy.[3]

Our experiments have shown the high potential of this approach and give rise to the following conclusions: The EA is computationally more complex but significantly outperforms the greedy strategy in terms of MGA scores. The alignments produced by the EA are better in the sense that conserved substructures are discovered more reliably. Besides, the improved similarity computation also leads to better performance in similarity retrieval. Finally, a reasonable compromise between solution quality and runtime is achieved by a combination of evolutionary optimization with binary decomposition techniques.

Literatur

[1] Bartz-Beielstein, T.: Experimental Research in Evolutionary Computation—The New Experimentalism. Springer-Verlag. 2006.

[2] J. Berg and M. Lassig.: Local graph alignment and motif search in biological networks. *Proceedings of the National Academy of Sciences*, 101(41):14689, 2004.

[3] Beyer, H.-G. and Schwefel, H.-P.: Evolution strategies – A comprehensive introduction. *Natural Computing*, 1(1):3–52. 2002.

[4] Borgwardt, K. M., and Kriegel, H.-P.: Shortest-path kernels on graphs. *Proc. Intl. Conf. Data Mining*, 74 - 81. 2005.

[5] Conte, D., Foggia, P., Sansone, C. and Vento, M.: Thirty Years of Graph Matching in Pattern Recognition. *Int. J. of Pattern Recognition and Artificial Intelligence*, 18(3):265–298. 2004.

[6] Gärtner, T.: A survey of kernels for structured data. *SIGKKD Explorations*, 5(1): 49 - 58. 2003.

[7] Hendlich, M., Rippmann, F. and Barnickel, G.: LIGSITE: Automatic and efficient detection of potential small molecule-binding sites in proteins. *Journal of Molecular Graphics and Modelling*, 15:359–363. 1997.

[8] Raymond, J., and Willett, P.: Maximum common subgraph isomorphism algorithms for the matching of chemical structures. *Journal of Computer-Aided Molecular Design*, 16: 521–533. 2002.

[9] Schmitt, S., Kuhn, D. and Klebe, G.: A New Method to Detect Related Function Among Proteins Independent of Sequence and Fold Homology. *J. Mol. Biol.*, 323(2):387–406. 2002.

[10] Weskamp, N., Hüllermeier, E., Kuhn, D. and Klebe, G.: Multiple Graph Alignment for the Structural Analysis of Protein Active Sites. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 4(2):310–320, 2007.

[3] An implementation of this algorithm along with a user's guide can be downloaded at `http://www.uni-marburg.de/fb12/kebi/research/software`.

A Propagation-based Algorithm for Inferring Gene-Disease Associations

Oron Vanunu* Roded Sharan*

Abstract: A fundamental challenge in human health is the identification of disease-causing genes. Recently, several studies have tackled this challenge via a two-step approach: first, a linkage interval is inferred from population studies; second, a computational approach is used to prioritize genes within this interval. State-of-the-art methods for the latter task are based on the observation that genes causing the same or similar diseases tend to lie close to one another in a network of protein-protein or functional interactions. However, most of these approaches use only local network information in the inference process. Here we provide a global, network-based method for prioritizing disease genes. The method is based on formulating constraints on the prioritization function that relate to its smoothness over the network and usage of prior information. A propagation-based method is used to compute a function satisfying the constraints. We test our method on gene-disease association data in a cross-validation setting, and compare it to extant prioritization approaches. We show that our method provides the best overall performance, ranking the true causal gene first for 29% of the 1,369 diseases with a known gene in the OMIM knowledgebase.

1 Introduction

Associating genes with diseases is a fundamental challenge in human health with applications to understanding disease mechanisms, diagnosis and therapy. Linkage studies are often used to infer genomic intervals that are associated with a disease of interest. Prioritizing genes within these intervals is a formidable challenge and computational approaches are becoming the method of choice for such problems. Prioritization methods are based on comparing a candidate gene to other genes that were implicated with the same or a similar disease. Recently, several methods were suggested that use physical network information for the prioritization task, and these were shown to outperform other approaches to the problem. The basic paradigm underlying these methods is that genes causing the same or a similar disease tend to lie close to one another in a protein-protein interaction (PPI) network.

Previous approaches to prioritizing disease-causing genes can be roughly classified according to whether prior knowledge on some of the genes (or genomic intervals) underlying a disease of interest is assumed or not. Approaches in the first category are based on computing the similarity between a given gene and the known disease genes. Such a similarity can be based on sequence [G+06], functional annotation [PIBAN07], protein-protein

*School of Computer Science, Tel Aviv University, Tel Aviv 69978. Email: {oronv,roded}@post.tau.ac.il.

interactions [O$^+$06, F$^+$06] and more. The reader is referred to [OB07] for a comprehensive review of these methods.

Approaches in the second category, which is the focus of the current work, are often based on a measure of phenotypic similarity (see, e.g., [vD$^+$06, L$^+$07]) between the disease of interest and other diseases for which causal genes are known. Lage et al. [L$^+$07] scores a candidate protein w.r.t. a disease of interest based on the involvement of its direct network neighbors in a similar disease. Kohler et al. [K$^+$08] group diseases into families. For a given disease, they employ a random walk from known genes in its family to prioritize candidate genes. Finally, Wu et al. [W$^+$08] score a candidate gene g for a certain disease d based on the correlation between the vector of similarities of d to diseases with known causal genes, and the vector of closeness in a protein network of g and those known disease genes.

In this work we suggest a global, network-based approach for predicting disease-causing genes. Our method falls under the second category and is able to exploit information on known genes for the disease of interest or for other similar diseases. The method receives as input a disease-disease similarity measure and a network of protein-protein interactions. It uses a propagation-based algorithm to infer a strength-of-association function that is smooth over the network (i.e., adjacent nodes are assigned similar values) and exploits the prior information (on causal genes for the same disease or similar ones).

Methodologically, we make a three-fold contribution: (i) we suggest a transformation from textual-based disease similarity values to confidence values that are learned automatically from data and better captures the probability that similar diseases share genes that lie close to one another in the network; (ii) we provide a propagation-based method for gene prioritization that takes into account, in a global manner, confidence values for disease similarity and a PPI network in which interactions are weighted by their reliability and the degrees of their end points. (iii) we re-implement three state-of-the-art methods and perform a comprehensive comparison between those methods and ours on the same input data.

On the practical side, we apply our method to analyze disease-gene association data from the Online Mendelian Inheritance in Man (OMIM) [H$^+$02] knowledgebase. We test, in a cross-validation setting, two possible applications of our method: (i) prioritizing genes for diseases with at least two known genes; (ii) prioritizing genes for all diseases (with at least one known gene). We compare the performance of our method to two state-of-the-art, recently published methods [K$^+$08, W$^+$08], as well as to a simple shortest-path based prioritization method. In all our tests the propagation-based method outperforms the other methods by a significant margin.

2 Our Algorithmic Approach

Preliminaries. The input to a gene prioritization problem consists of a set A of gene-disease associations; a query disease q; and a protein-protein interaction network $G = (V, E, w)$, where V is the set of proteins, E is the set of interactions and w is a weight

function denoting the reliability of each interaction. The goal is to prioritize all the proteins in V (that are not known to be associated with q) w.r.t. q.

For a node $v \in V$, denote its direct neighborhood in G by $N(v)$. Let $F : V \rightarrow \Re$ represent a prioritization function, i.e., $F(v)$ reflects the relevance of v to q. Let $Y : V \rightarrow [0, 1]$ represent a prior knowledge function, which assigns positive values to proteins that are known to be related to q, and zero otherwise.

Intuitively, we wish to compute a function F that is both smooth over the network, i.e., adjacent nodes are assigned with similar values, and also respects the prior knowledge, i.e., nodes for which prior information exists should have similar values of F and Y. These requirements often conflict with each other, e.g., when two adjacent nodes have very different Y values. Formally, we express the requirements on F as a combination of these two conditions:

$$F(v) = \alpha[\sum_{u \in N(v)} F(u)w'(v, u)] + (1 - \alpha)Y(v) \tag{1}$$

where w' is a normalized form of w, such that $\sum_{u \in N(v)} w'(v, u) \leq 1$ for every node $v \in V$. Here, the first term expresses the smoothness condition, while the second term expresses the prior information constraint. The parameter $\alpha \in (0, 1)$ weighs the relative importance of these constraints w.r.t. one another.

Computing the prioritization function. The requirements on F can be expressed in linear form as follows:

$$F = \alpha W'F + (1 - \alpha)Y \Leftrightarrow F = (I - \alpha W')^{-1}(1 - \alpha)Y \tag{2}$$

where W' is a $|V| \times |V|$ matrix whose values are given by w', and F and Y are viewed here as vectors of size $|V|$. Since W' is normalized, its eigenvalues are in $[0, 1]$. Since $\alpha \in (0, 1)$, the eigenvalues of $(I - \alpha W')$ are in $(0, 1]$; in particular, all its eigenvalues are positive and, hence, $(I - \alpha W')^{-1}$ exists.

While the above linear system can be solved exactly, for large networks an iterative propagation-based algorithm works faster and is guaranteed to converge to the system's solution. Specifically, we use the algorithm of Zhou et al. [Z+03] which at iteration t computes

$$F^t := \alpha W'F^{t-1} + (1 - \alpha)Y$$

where $F^0 := 0$. This iterative algorithm can be best understood as simulating a process where nodes for which prior information exists pump information to their neighbors. In addition, every node propagates the information received in the previous iteration to its neighbors. In practice, as a final iteration we apply the propagation step with $\alpha = 1$ to smooth the obtained prioritization function F.

We chose to normalize the weight of an edge by the degrees of its end-points, since the latter relate to the probability of observing an edge between the same end-points in a random network with the same node degrees. Formally, define a diagonal matrix D such that $D(i, i)$ is the sum of row i of W. We set $W' = D^{-1/2}WD^{-1/2}$ which yields a symmetric matrix with row sums ≤ 1, where $W'_{ij} = W_{ij}/\sqrt{D(i, i)D(j, j)}$.

Incorporating disease similarity information. As observed by several authors [L$^+$07, OB07], similar diseases are often caused by proteins in the same complex or signalling pathway; therefore, such proteins tend to lie close to one another in the network. This empirical observation motivated us to use disease similarity information to determine the prior information vector Y.

We used the similarity metric computed by van Driel et al. [vD$^+$06], which spans $5,080$ diseases in the OMIM [H$^+$02] knowledgebase. Each disease entry in OMIM was scanned for terms taken from the anatomy (A) and the disease (C) sections of the medical subject headings vocabulary (MeSH). A disease was then represented by a binary vector specifying the terms associated with it. Similarity between diseases was computed by taking the cosine of the angle between the corresponding vectors.

van Driel et al. also tested the predictive power of different ranges of similarity values by calculating the correlation between the similarity of two diseases and the functional relatedness of their causative genes. According to their analysis, similarity values in the range $[0, 0.3]$ are not informative, while for similarities in the range $[0.6, 1]$ the associated genes show significant functional similarity. These empirical findings motivated us to represent our confidence that two diseases are related using a logistic function $L(x) = \frac{1}{1+e^{(cx+d)}}$. We constrained $L(0)$ to be close to zero (0.0001) which determines d (as $\log(9999)$), and tuned the parameter c using cross validation (see Parameter Tuning Section below). We used L to compute the prior knowledge Y in the following way: for a query disease q and a protein v associated with a disease d, we set $Y(v) := L(s)$, where s is the similarity between q and d. If v is associated with more than one disease, we set s to be the maximal similarity between q and any of those diseases.

3 Experimental Setup

We extracted $1,600$ known disease-protein associations from GeneCards[R$^+$97] spanning $1,369$ diseases and $1,043$ proteins. We considered only disease-protein relations that included proteins from the network and such that the relations are known to be causative to avoid associations made by circumstantial evidence.

We constructed a human PPI network with $9,998$ proteins and $41,072$ interactions that were assembled from three large scale experiments [R$^+$05, S$^+$05b, E$^+$07] and the Human Protein Reference Database (HPRD) [P$^+$04]. The interactions were assigned confidence scores based on the experimental evidence available for each interaction using a logistic regression model adapted from [S$^+$05a]. We used the obtained scores to construct the adjacency matrix W.

To simulate the case of prioritizing proteins encoded by genes inside a linkage interval, we followed [K$^+$08] and artificially constructed for each protein associated with a disease an interval of size 100 around it. We used the F values obtained from the output of the algorithm to prioritize proteins residing in that interval.

Comparison to other methods. In order to perform a comprehensive comparison of our approach to extant ones on the same input data, we re-implemented two state-of-the-art approaches for gene prioritization: the random-walk based method of [K$^+$08] and the CIPHER [W$^+$08] algorithm. In addition we implemented a simple shortest-path based approach for the problem. We describe the implementation details below. We note that we could not compare our method to that of Lage et al. [L$^+$07], as code or input data for the latter method were not readily available.

The random-walk based approach requires disease grouping information. To allow it to run on the more comprehensive disease similarity data we had, we generalized the approach to use these similarities (transformed by the logistic function L) as initial probabilities for the random walk. The parameter r of the method, which controls the probability for a restart, as well as our transformation parameter c, were optimized using cross-validation (as in the Parameter Tuning Section below). Note that Kohler et al. suggested a second, diffusion-kernel based approach, which was shown to be inferior to the random walk one, hence we did not include it in our comparison. Also note that our propagation-based method reduces to a random walk under appropriate transformations of the edge weights and prior information.

The CIPHER method [W$^+$08] is based on computing protein closeness in a PPI network. Two variants of the algorithm were suggested: CIPHER-DN, which considers only direct neighbors in the closeness computation, and CIPHER-SP, which is based on a shortest path computation. The former was shown to outperform the latter, and hence we implemented this variant (CIPHER-DN) only.

In addition, we implemented a simple shortest-path (SP) based approach, in which a candidate protein is scored according to the most probable path to a disease-related protein. Formally, define the probability of a path connecting a candidate protein to a causal protein v, as the product of the normalized weights w' of the edges along the path and $Y(v)$. The score of a candidate protein is then the score of its best path.

Performance evaluation. To evaluate the performance of the different methods we tested, we used a leave-one-out cross validation procedure. In each cross-validation trial, we removed a single disease-protein association $< d, p >$ from the data, and in addition all other associations involving protein p. An algorithm was evaluated by its success in reconstructing the hidden association, i.e. by the rank it assigned to protein p when querying disease d. The reason we hid all associations of p was to avoid "easy" cases in which p is also associated with other diseases that are very similar to d.

We evaluated the performance of an algorithm in terms of overall precision vs. recall when varying the rank threshold $1 \leq k \leq 100$. *Precision* is the fraction of gene-disease associations that ranked within the top $k\%$ at some trials and are true associations. In other words, it is the number of trials in which a hidden association was recovered as one of the top $k\%$ scoring ones, over the total number of trials times $k\%$ of the interval size. *Recall* is the fraction of trials in which the hidden association was recovered as one of the top $k\%$ scoring ones.

In addition, we used two other measures for quality evaluation. The first, is the *enrichment*

measure [L$^+$07] which is defined as follows: If the correct gene is ranked in the top $m\%$ in $n\%$ of the trials then there is a n/m-fold enrichment. For example, if the algorithm ranks the correct gene in the top 10% in 50% of the cases, a 5-fold enrichment is achieved, while random prioritization of the genes is expected to rank the correct gene in the top 10% only in 10% of the cases, yielding a 1-fold enrichment. The second, is the *average rank* of the correct gene throughout the cross-validation trials. Note that when $m = 1$, recall, precision and enrichment measures are all equal.

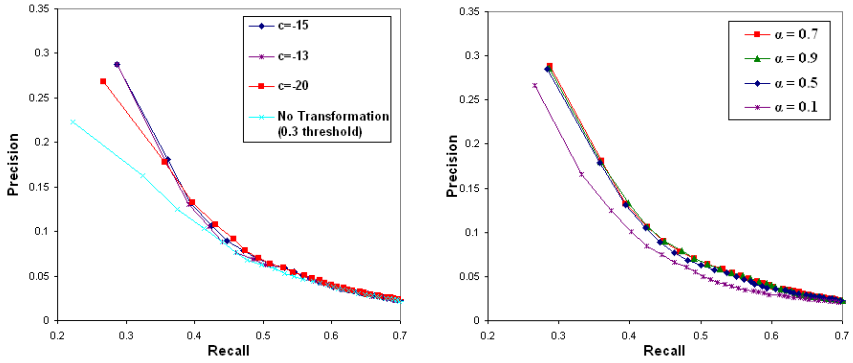
4 Results

We implemented our propagation algorithm and tested its performance in recovering known disease-gene association both on 150 diseases for which more than one causal gene is known, and the entire collection of 1,369 diseases. We report these results and compare our algorithm to previous state-of-the-art algorithms for the prioritization problem.

Parameter tuning. Our algorithm has three parameters that should be tuned: (i) c – the parameter controlling the logistic regression transformation; (ii) α – controlling the relative importance of prior information in the association assignment; and (iii) the number of propagation iterations employed. We used the cross validation framework to test the effect of these parameters on the performance of the algorithm. The precision-recall plots for the general disease case are depicted in Figure 1. By Figure 1(a) the optimal regression coefficient is $c = -15$, implying that similarity values below 0.3 are assigned with very low probability(< 0.002), in accordance with the analysis of [vD$^+$06]. The algorithm is not sensitive to the actual choice of α as long as it is above 0.5 (Figure 1(b)). Finally, the algorithm shows fast convergence, achieving optimal results after only ten iterations (data not shown). Similar results were obtained in the tuning process for diseases with more than one known gene.

Diseases with more than one known gene. Our first set of tests focused on 150 diseases for which more than one causal gene is known. For such diseases we first checked whether our algorithm gains in performance when incorporating information on similar diseases, compared to when using information on the disease of interest d alone. For the latter case we set $Y(v) := 1$ if protein v is associated with d and $Y(v) := 0$ otherwise. As evident from Figure 2 the disease similarity information improves the quality of predictions.

Next, we compared the performance of our algorithm to those of the random-walk and CIPHER methods, as well as to our SP variant. The results are depicted in Figure 3 and summarized in Table 1. Our algorithm achieved the best performance, ranking the correct gene as the top-scoring one in 50.9% of the cases. Interestingly, SP was the second-best performer with 43.7% correct top-1 predictions, while the method of [K$^+$08] and CIPHER attained lower success rates of 40.9% and 37.5%, respectively.



(a) (b)

Figure 1: Effect of parameters on our algorithm's performance, as measured in cross-validation on the set of 1,369 diseases with a known gene. (a) Precision vs. recall plots for different c values, as well as for a simple identity transformation in which values below 0.3 are ignored. (b) Performance comparison for different α values.

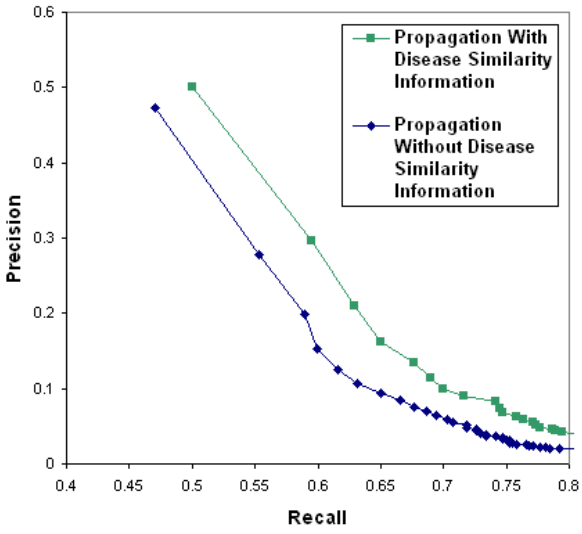


Figure 2: The effect of incorporating disease similarity information on prioritizing genes for 150 diseases with more than one known gene.

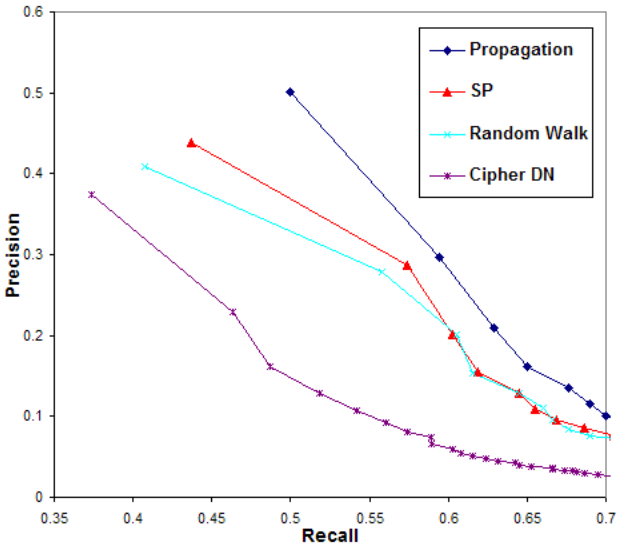


Figure 3: Performance comparison on 150 diseases with more than one known gene.

General diseases. Our second set of tests concerned all 1,369 diseases with a known gene in the OMIM database. The results of applying the different methods are depicted in Figure 4 and summarized in Table 1. Again, our algorithm achieved the best performance, ranking the correct gene as the top-scoring one in 28.7% of the cases. SP, CIPHER and random-walk methods all achieved inferior results with 26.8%, 22.6% and 21.7% success rates, respectively.

5 Acknowledgements

We thank Tomer Shlomi for his help in preprocessing the data and insightful suggestions. We also thank Nir Yosef for providing us with the human PPI network. This research was supported by a German-Israel Foundation grant.

References

[E+07] R. Ewing et al. Large-scale mapping of human protein-protein interactions by mass spectrometry. *Mol Syst Biol*, 3, 2007. 10.1038/msb4100134.

[F+06] L. Franke et al. Reconstruction of a functional human gene network, with an application for prioritizing positional candidate genes. *Am J Hum Genet*, 78(6):1011–1025, June 2006.

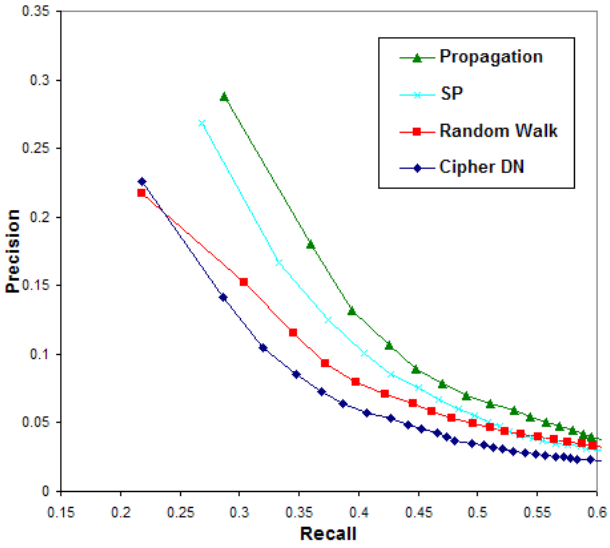


Figure 4: Performance comparison on 1,369 diseases with a known gene.

[G+06] R. A. George et al. Analysis of protein sequence and interaction data for candidate disease gene prediction. *Nucleic Acids Res*, 34(19):e130, 2006.

[H+02] A. Hamosh et al. Online Mendelian Inheritance in Man (OMIM), a knowledgebase of human genes and genetic disorders. *Nucl. Acids Res.*, 30(1):52–55, January 2002.

[K+08] S. Kohler et al. Walking the Interactome for Prioritization of Candidate Disease Genes. *American journal of human genetics*, 82(4):949–958, 2008.

[L+07] K. Lage et al. A human phenome-interactome network of protein complexes implicated in genetic disorders. *Nat Biotech*, 25(3):309–316, 2007.

[O+06] M. Oti et al. Predicting disease genes using protein-protein interactions. *J Med Genet*, 43(8):691–698, August 2006.

[OB07] M. Oti and MG. Brunner. The modular nature of genetic diseases. *Clinical Genetics*, 71(1):1–11, January 2007.

[P+04] S. Peri et al. Human protein reference database as a discovery resource for proteomics. *Nucleic Acids Res*, 32(Database issue):D497–501, 2004.

[PIBAN07] C. Perez-Iratxeta, P. Bork, and M. A. Andrade-Navarro. Update of the G2D tool for prioritization of gene candidates to inherited diseases. *Nucleic Acids Res*, 35(Web Server issue):W212–6, 2007.

[R+97] M. Rebhan et al. GeneCards: integrating information about genes, proteins and diseases. *Trends in Genetics*, 13:163, 1997.

[R+05] JF. Rual et al. Towards a proteome-scale map of the human protein-protein interaction network. *Nature*, 437(7062):1173–1178, 2005.

General diseases		
Method	Enrichment	Avg Rank
Propagation	**28.7**	**24.2**
SP	26.8	25
Kohler et al.	21.7	25.7
Wu et al.	22.6	29.5
Diseases with more than one known gene		
Method	Enrichment	Avg Rank
Propagation	**50.9**	**14.3**
SP	43.7	15
Kohler et al.	40.9	15.4
Wu et al.	37.5	21.2

Table 1: Summary of performance comparison on two collections of diseases.

[S+05a] R. Sharan et al. Conserved patterns of protein interaction in multiple species. *Proc Natl Acad Sci U S A*, 102(6):1974–1979, February 2005.

[S+05b] U. Stelzl et al. A human protein-protein interaction network: a resource for annotating the proteome. *Cell*, 122(6):957–68, 2005.

[vD+06] M. van Driel et al. A text-mining analysis of the human phenome. *Eur J Hum Genet*, 14(5):535–542, 2006.

[W+08] X. Wu et al. Network-based global inference of human disease genes. *Mol Syst Biol*, 4, May 2008.

[Z+03] D. Zhou et al. Learning with local and global consistency, 2003. In 18th Annual Conf. on Neural Information Processing Systems.

Designing Binding Pockets on Protein Surfaces using the A* Algorithm

Susanne Eyrisch, Volkhard Helms

Center for Bioinformatics
Saarland University
P.O. Box 151150
D-66041 Saarbruecken, Germany

eyrisch@bioinformatik.uni-saarland.de

volkhard.helms@ bioinformatik.uni-saarland.de

Abstract: The in-silico design of ligands binding to the protein surface instead of deep binding pockets is still a great challenge. Often no appropriate binding pockets are available in the apo experimental structures and standard virtual screening techniques will fail. Here, we present two new algorithms for designing tailored ligand binding pockets on the protein surface that account for protein backbone and side chain flexibility. At first, the protein surface is scanned for potential pocket positions using a program named PocketScanner. This program minimizes the protein energetically in the presence of generic pocket spheres representing the new binding pockets whose positions remain fixed. The side chains of the relaxed protein conformations are then further refined by a second program named PocketBuilder. PocketBuilder identifies all residues within a given radius of the pocket positions and searches for the best combination of side chain rotamers using the A* algorithm. Given multiple protein conformations as input, PocketBuilder identifies those that lead to the best results, namely protein conformations of low energy that possess binding pockets with desired properties. The approach was tested on the proteins BCL-X_L, IL-2, and MDM2 which are involved in protein-protein interactions and hence represent challenging drug targets. Although the native ligand binding pocket was not or only partly open in the apo crystal or NMR structures, PocketScanner and PocketBuilder successfully generated conformations with pockets into which a known inhibitor could be docked in a native-like orientation for two out of the three test systems. For BCL-X_L, the docking scores were even similar to those obtained in re-docking experiments to the inhibitor bound crystal structure.

1 Introduction

After realizing that most diseases arise from aberrant molecular interactions, it has become an important goal to identify these interactions and to modulate them, for example through the binding of additional ligands, so that the native biological processes are reestablished or unwanted interactions are inhibited. In particular, the development of computational

tools supporting the design process of such modulators has become a very interesting research area. It involves, for example, the in-silico design of ligands that should bind to concave regions on the surface of the target protein. Given the three-dimensional target structure, one may predict the binding modes of a potential ligand by scanning the protein surface for favorable binding pockets. To this end, several computational tools have been developed that use geometric (e.g. PASS [BS00], SURFNET [Las95], PocketFinder [ATA05], LigSite [HRB97]) or energetic (e.g. GRID [Goo85], MCSS [CMK93], QSiteFinder [LJ05], CS-Map [LLY+07]) criteria for detecting such clefts or hot spots. An example for pocket detection tools using only geometric criteria is the PASS (Putative Active Sites with Spheres) algorithm that identifies empty volumes on the protein surface by their burial extent. An example for the tools using energetic criteria for identifying putative hot spots is MCSS that generates positions and orientations of functional groups in the field of a flexible protein. In the case of enzymes, such binding pockets often correspond to active sites that deeply extend into the protein interior and are relatively easy to identify. It is much harder to detect pockets located at flat protein surfaces that often require structural rearrangements to open and therefore may not be fully accessible in the protein conformation used. The lack of clearly shaped binding pockets at protein-protein interfaces is one of the reasons why the structure-based drug design of small molecule protein-protein interaction inhibitors (SMPPIIs) remains a great challenge [AW04]. Until today, most published SMPPIIs for this class of drug targets were identified by experimental screening methods [WM07].

We have previously presented a pocket detection protocol that provides a starting point for in silico drug design for cases in which no potential binding pocket could be identified so that standard screening methods would fail [EH07]. For the three protein systems MDM2, BCL-X_L, and interleukin-2 (IL-2), we found that large pockets not detectable in the crystal structures of the free proteins opened frequently on the protein surfaces during standard molecular dynamics (MD) simulations of 10 nanoseconds length at room temperature. The identified transient pockets represent potential binding sites for new inhibitors. At the native binding site, pockets of similar size as with a known inhibitor bound could indeed be observed for all three systems. Docking known inhibitors with AutoDock 3 [MGH+98] into these transient pockets resulted in docking results with less than 2 Å root mean square deviation (RMSD) from the crystal structures. In a subsequent study, we could show that when the water solvent was replaced by methanol the transient pockets opening in the MD simulations tended to be larger and less polar (unpublished results). Moreover, the docking results improved significantly for two of the three systems. However, a limiting factor of this pocket detection protocol is the high computational demand of MD simulations on biomolecular systems and it would be desirable to achieve the opening of surface pockets by a more efficient protocol. Fortunately, in many drug design applications, the approximate location of the binding site is already known. Hence, it is sufficient to sample only the corresponding part of the protein surface. This local instead of global search allows for a more accurate and directed sampling of low-energy protein conformations with accessible pockets. These protein conformations can then be used to optimize the interaction between the protein and the ligand or for virtual screening. We will show below that the problem of finding appropriate protein conformations can be solved efficiently using an informed graph search algorithm like the A* algorithm [HNR68] that uses knowledge about

the structure of the search space incorporated in heuristic functions to guide the search towards optimal solutions. During this search, a graph is built up in which each node represents a partial solution. Given an initial node representing the initial state, the algorithm searches the path to a given goal node, representing the goal state. The generated nodes are maintained in a priority queue. The priority of a partial solution *x* is given by

$$f(x) = g(x) + h(x) \tag{1}$$

where *g(x)* is the cost of this partial solution so far, i.e. from the start node to *x* and *h(x)* is the heuristic estimate of the minimal cost to reach the goal node from *x*. If the heuristic function is admissible (i.e. it never overestimates the cost of reaching the goal node) and consistent (i.e. it fulfills the triangle inequality), it will always find a path with minimal cost from a given start node to a given goal node if such a goal node exists. Leach applied the A* search to the flexible docking and the side chain placement problems [Lea94]. After placing an anchor region of the ligand into the binding site, he generated all possible ligand conformations. For each conformation that made no unfavorable interactions with the protein backbone and all rotameric states of a residue, the optimal combination of side chain rotamers was determined by an A* search. The initial node represented the structure without assigned rotamers for the residues at the binding site, while the goal nodes represented the optimal docking solutions, i.e. all residues had assigned rotamers.
In this work, we incorporated ideas from PASS, MCSS, and Leach's application of the A*-search into two new algorithms for the efficient generation of energetically favorable protein conformations with accessible binding pockets at defined locations on the surface of the BCL-X_L, IL-2, and MDM2 proteins.

2 Methods and Materials

Our method uses two programs for the construction of putative binding pockets: PocketScanner and PocketBuilder. PocketScanner scans a user-defined region of the protein surface for potential pocket positions and generates protein conformations in which the backbone has adapted to these pocket positions. PocketBuilder uses these intermediate conformations for calculating a final set of conformations that best fulfil the search criteria, namely the desired trade-off between a protein conformation with low-energy side chain rotamers and a pocket of defined volume. Both programs were implemented in C++ using the BALL library [KL00] and the CHARMM EEF1 [LK99] force field that was used to compute all energies given below. This force field treats the solvent as an implicit continuum, and including such effects is crucial for designing pockets on protein surfaces. Binding pockets are represented by generic pocket spheres that were added to the force field. In the current setup, they only interact with the protein atoms via van-der-Waals interactions (with a radius of 1, 2, or 3 Å and a well depth of 0.05 kcal/mol). The pocket volumes and polarities were calculated as described in [EH07].

2.1 Structure Preparation

The unbound (apo) and inhibitor-bound protein structures of three test systems were taken from the Protein Data Bank [BWF+00] (PBD entries 1R2D and 1YSI for BCL-X_L, 1M47 and 1PY2 for IL-2, and 1Z1M and 1T4E for MDM2). All hetero atoms (including the ligand) were manually removed. As residues 28-81 are missing in 1R2D, the two parts of the protein were modeled as two distinct chains. The missing residues in 1M47 were modelled as loops of the lowest AMBER/GBSA potential energy generated by the program RAPPER [dBDBB03]. The structure of apo MDM2 is represented by 24 NMR models that differ mainly in the loop regions. Since no model is defined as most representative, the first model was chosen. The apo structures were superimposed on the inhibitor-bound structures based on the C_α-coordinates using the VMD program [HDS96].

2.2 The PocketScanner Algorithm

PocketScanner creates a grid around a given center with suitable dimensions and edge length and scans the protein surface for potential positions of pockets with a given radius. The z-axis of this grid is the solvent vector defined by the initial pocket position and the center of gravity of the 10 nearest solvent exposed atoms. The generic pocket sphere representing the pocket center is then placed on each grid point and its burial count (number of protein atoms within 8 Å) is calculated. Only those positions with a burial count above a given threshold (default: 65) are accepted, otherwise the resulting cavity may be too flat. As this criterion allows for pocket positions that are deeply buried inside the protein, we additionally require that the minimal distance to any solvent exposed atom must be smaller than 2 Å. The protein is then energy minimized in the presence of the generic pocket sphere using 500 steps of L-BFGS or until the RMS gradient is smaller than 0.01 kcal $\text{mol}^{-1}\text{Å}^{-1}$. During this energy minimization, the position of the generic pocket sphere is fixed, so that the protein relaxes its conformation. If the burial count is still high enough after the energy minimization, this protein conformation in combination with this pocket position is written to an output file and can be used as a starting conformation for PocketBuilder.

2.3 The PocketBuilder Algorithm

For calculating multiple protein conformations with putative binding pockets, PocketBuilder needs the following input data and parameters: starting conformations with putative pocket positions (either generated by PocketScanner or manually selected pocket positions), the radius of these pockets, a search radius for defining the flexible residues (default: 8 Å), a rotamer library, weights for scoring the internal protein energy, w_{energy}, and the van-der-Waals interaction energy with the pockets of the generated conformations, w_{pocket}, and the number of conformations to be generated. The algorithm consists of the initialization stage and the A*-search. The initialization is performed separately for each

starting conformation. It starts with determining all *N* residues (except for Ala and Gly) within a given distance from the generic pocket sphere and defines them as flexible. For the rigid part of the protein including all other residues and the backbone and C_β atoms of the flexible residues, the energy E_{rigid} and the van-der-Waals interaction energy with the pocket (i.e. the generic pocket spheres) $E_{rigid,pocket}$ are calculated. For each of the flexible residues *i* all rotamers *j* defined by the Dunbrack backbone independent rotamer library from 2002 [DC97] (including the original side chain conformation) are tested and their van-der-Waals interaction energy with the pocket $E_{i_j,pocket}$ and the energy change ΔE_{i_j} resulting from including this side chain rotamer in the calculation of E_{rigid} are determined. After calculating $E_{i_j}^{weighted}$ for each rotamer as

$$E_{i_j}^{weighted} = w_{energy} \cdot \Delta E_{i_j} + w_{pocket} \cdot E_{i_j,pocket} \tag{2}$$

the number of allowed rotamers for this residue is reduced by deleting all rotamers *j* with $E_{i_j}^{weighted} \geq 100$ kcal/mol. The pairwise non-bonded interaction energies E_{i_j,k_l} between the remaining rotamers *j* and *l* of each pair of residues *i* and *k* are calculated and stored in a hash table.

After the initialization stage, the algorithm builds up a tree with one subtree per starting conformation. The nodes in this tree represent partial solutions of the search problem, or more precisely conformations in which rotameric states have only been assigned to a part of the flexible residues. The order in which the flexible residues get defined side chain conformations is fixed, so all nodes of the same level in a certain subtree have the same residues already assigned. (The order in which side chains are added has no effect on the final result.) Note that the levels of the leave nodes are identical within a subtree, but may differ within different subtrees depending on the number of flexible residues defined for this starting conformation. The buildup of the tree is controlled by the A* algorithm. The algorithm assigns each node *x* a priority *f(x)* (see equation 1) that evaluates the true costs *g(x)* of this partial conformation so far and the estimated minimal cost *h(x)* for reaching a leaf node, where

$$\begin{aligned} g(x) \quad = \quad & w_{pocket} \cdot E_{rigid,pocket} + w_{energy} \cdot E_{rigid} + \\ & \sum_{i=1}^{x} \left(w_{pocket} \cdot E_{i_r,pocket} + w_{energy} \cdot \left(\Delta E_{i_r} + \sum_{k=1}^{i-1} E_{i_r,k_r} \right) \right) \end{aligned} \tag{3}$$

$$\begin{aligned} h(x) \quad = \quad & \sum_{k=x+1}^{N} \min_{l} \left(w_{energy} \cdot \Delta E_{k_l} + w_{pocket} \cdot E_{k_l,pocket} \right) + \\ & \sum_{k=x+1}^{N} \left(\left(\sum_{i=1}^{x} \min_{l} E_{i_r,k_l} \right) + \left(\sum_{n=x+2}^{N} \min_{l,m} E_{k_l,n_m} \right) \right) \end{aligned} \tag{4}$$

In the summations, *i* runs over all flexible residues with already assigned rotamers *r* (i.e. E_{i_r} indicates that side chain *i* has been locked into rotamer *r*), *k* and *n* run over the remaining ones, and *l* and *m* run over different rotamers of a side chain. In each step, the node

representing the partial conformation that seems most promising (i.e. with lowest *f(x)*) is selected. If this node is not a leaf node, a new node is added for each possible rotamer of the succeeding residue and the priorities of these new partial solutions are determined. Otherwise the corresponding conformation is written to an output file. The algorithm terminates as soon as the total number of output conformations is reached.

2.4 Docking into Designed Pockets

Docking experiments were performed with AutoDock 3.0.5 [MGH+98] as described before [EH07]. The ligands were extracted from the complex crystal structure and rotatable bonds were assigned with AutoTors. The grid maps were calculated with AutoGrid. The grid centers were chosen to coincide with the pocket positions. The default grid spacing of 0.375 Å between the grid points and the default grid dimension of 60 x 60 x 60 points was used and the standard Lamarckian Genetic Algorithm protocol with the default values. 10 independent docking runs were carried out for each PocketBuilder conformation.

3 Results and Discussion

PocketScanner and PocketBuilder were tested using the proteins BCL-X_L, MDM2, and IL-2. IL-2 is an important component of the immune response and BCL-X_L and MDM2 belong to the apoptosis pathway. The binding pockets targeted by the small molecule ligands are not or not fully open in the apo protein structures and thus cannot be used for structure-based drug design. Docking the known inhibitors into these apo structures gave poor results with lowest RMSDs of 2.9 - 3.4 Å as shown in Table 1.

	Re-Docking		Apo-Docking	
System	RMSD [Å]	Score [kcal/mol]	RMSD [Å]	Score [kcal/mol]
BCL-X_L - N3B	0.9	-10.5	3.3	-6.2
IL-2 - FRH	1.1	-10.8	2.9	-6.2
MDM2 - DIZ	1.1	-13.1	3.4	-6.7

Table 1: Best docking results for docking the inhibitor into its bound and the apo structure using AutoDock3

The crystal or NMR structures of the apo proteins were scanned for positions of inducible pockets using PocketScanner. The grid center was placed at the ligand center of mass, the dimension was 11, and the edge length 2 Å. Running PocketScanner took about 1 hour on a single CPU of an Intel Core 2 Duo processor which mainly resulted from the large number of energy minimizations. Out of the 11^3 possible positions, 67 (66) were accepted for BCL-X_L, 25 (18) for IL-2, and 29 (20) for MDM2 when using a pocket radius of 2 Å (3 Å respectively). Note that the pocket positions do not have to be located at the inhibitor binding site. The grid and the accepted positions of BCL-X_L are shown in Fig. 1.

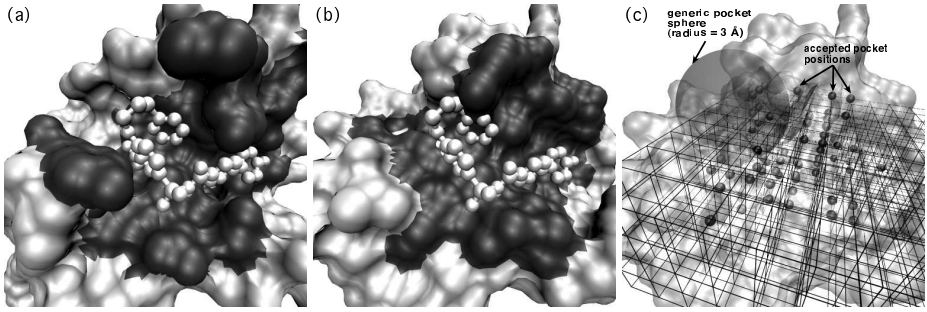


Figure 1: (a) BCL-X_L bound to its small molecule inhibitor N3B (shown as white balls and sticks), (b) BCL-X_L in its apo conformation with the inhibitor (shown to illustrate clashes with protein residues), and (c) with the grid generated by PocketScanner (accepted pocket positions are shown as black spheres) and an example for a generic pocket sphere.

Running PASS revealed that more pockets were detected in the PocketScanner conformations when the larger pocket radius was used. An overview of the properties of the pockets generated using PocketScanner is shown in Table 2. One would expect that the mean pocket volume would increase when using a larger pocket radius, but this is not the case for MDM2. A larger pocket radius may also cause a flat cavity and thus a pocket of reduced volume.

System	Pocket Radius [Å]	Detected Pockets [%]	Ø Pocket Volume [$Å^3$]	Ø Pocket Polarity
BCL-X_L	2	43	381.3 ± 82.3	0.33 ± 0.03
	3	86	394.9 ± 109.7	0.29 ± 0.04
IL-2	2	52	311.8 ± 59.1	0.31 ± 0.04
	3	78	328.8 ± 58.9	0.29 ± 0.03
MDM2	2	31	376.8 ± 91.7	0.33 ± 0.02
	3	75	315.7 ± 98.6	0.31 ± 0.03

Table 2: Properties of the pockets detected in the conformations generated using PocketScanner

The conformations and the corresponding pocket positions generated by PocketScanner with a pocket radius of 2 and 3 Å were then used as starting conformations for PocketBuilder. As the weighting of the internal protein energy and the protein-pocket interaction energy crucially influences the A* search, we calculated 500 final conformations using three different weightings for the two pocket radii: (1) internal protein energy and protein-pocket interaction energy weighted equally (0.5 and 0.5), (2) a strong emphasis on the pocket (0.1 and 0.9), and (3) a dominance of the pocket (0.01 and 0.99). The bottleneck for the run time of PocketBuilder is the initialization stage. This stage takes 6-10 minutes per starting conformation depending on the number of flexible residues (here, 8-18 flexible residues) and the number of accepted rotamers. For this purpose we added a greedy preselection of the starting conformations: For each conformation, the weighted sum of the

internal protein energy and protein-pocket interaction energy is calculated and the 20 starting conformations with lowest score are retained. We are aware that this preselection may delete conformations that would later on score better with altered side chain rotamers, but running the algorithm with too many starting conformations is nearly infeasible. The A* search took between 40 minutes and 4 hours depending on the number of possible nodes in the search tree and on how similar the scores of these nodes are. The ratio between the number of possible nodes and the number of generated nodes gives a measure for the efficiency of the algorithm. As listed in Table 3, the number of possible conformations increases with augmenting w_{pocket}. At the same time, the algorithm generally finds the 500 leaf nodes with lowest score more efficiently, suggesting that the interaction energy between the protein and the pocket is more diverse in the generated nodes than the internal protein energy. This is not surprising as the absolute value of the internal protein energy is about 4 orders of magnitude larger than the interaction energy with the pocket. No trend is apparent for the influence of the weighting and the pocket radius on the mean pocket volume and polarity. These mean volumes even seem to suggest that PocketBuilder reduces the volume of most pockets to snugly fit around the generic pocket spheres.

System	Pocket Radius [Å]	w_{pocket}	# Conformations	Efficiency	Ø Pocket Volume [Å^3]	Ø Pocket Polarity
BCL-X$_L$	2	0.5	$1.0 \cdot 10^{12}$	$8.3 \cdot 10^{6}$	715.3 ± 21.9	0.36
	2	0.9	$1.9 \cdot 10^{12}$	$1.7 \cdot 10^{7}$	343.6 ± 31.7	0.27 ± 0.01
	2	0.99	$3.4 \cdot 10^{12}$	$1.6 \cdot 10^{9}$	337.4 ± 37.2	0.27 ± 0.01
	3	0.5	$1.7 \cdot 10^{11}$	$2.4 \cdot 10^{6}$	282.6 ± 34.2	0.30 ± 0.01
	3	0.9	$5.6 \cdot 10^{11}$	$7.1 \cdot 10^{6}$	276.1 ± 55.0	0.31 ± 0.01
	3	0.99	$4.5 \cdot 10^{14}$	$1.2 \cdot 10^{9}$	485.2 ± 92.7	0.37 ± 0.01
IL-2	2	0.5	$2.0 \cdot 10^{15}$	$1.0 \cdot 10^{11}$	291.7 ± 3.8	0.27
	2	0.9	$2.7 \cdot 10^{16}$	$9.3 \cdot 10^{11}$	290.3 ± 4.8	0.27
	2	0.99	$1.9 \cdot 10^{18}$	$4.1 \cdot 10^{13}$	359.6 ± 36.3	0.33 ± 0.01
	3	0.5	$1.2 \cdot 10^{14}$	$5.9 \cdot 10^{9}$	450.9 ± 80.2	0.31 ± 0.01
	3	0.9	$2.2 \cdot 10^{15}$	$6.9 \cdot 10^{10}$	507.4 ± 90.7	0.30 ± 0.01
	3	0.99	$4.6 \cdot 10^{16}$	$1.2 \cdot 10^{12}$	344.4 ± 23.3	0.33 ± 0.01
MDM2	2	0.5	$1.5 \cdot 10^{14}$	$1.4 \cdot 10^{10}$	314.0 ± 56.8	0.31 ± 0.02
	2	0.9	$1.4 \cdot 10^{15}$	$1.4 \cdot 10^{10}$	420.0 ± 50.2	0.33 ± 0.02
	2	0.99	$2.1 \cdot 10^{16}$	$2.4 \cdot 10^{7}$	277.9 ± 19.6	0.32 ± 0.01
	3	0.5	$2.6 \cdot 10^{12}$	$7.0 \cdot 10^{8}$	233.8 ± 26.3	0.32 ± 0.01
	3	0.9	$8.8 \cdot 10^{13}$	$7.6 \cdot 10^{9}$	235.3 ± 27.1	0.32 ± 0.01
	3	0.99	$2.0 \cdot 10^{15}$	$1.4 \cdot 10^{10}$	339.1 ± 89.1	0.31 ± 0.02

Table 3: Influence of the pocket radius and the weighting on the performance of PocketBuilder and the properties of the induced pockets

The main goal of this study is to design pockets on the protein surface that are suitable for ligand binding. Therefore, the known inhibitors were now docked into the generated conformations. The main questions are: (1) Can docking into the designed pockets reproduce

the native ligand binding mode? (2) Which weighting and pocket radius requires the lowest number of generated conformations? Table 4 lists the best scored docking results with RMSD $\leq$ 2 Å (or the docking result with lowest RMSD) for each weighting and pocket radius.

System	Pocket Radius [Å]	w_{pocket}	RMSD [Å]	Score [kcal/mol]	Relative Score Rank [%]	PocketBuilder Conformation
BCL-X_L	2	0.5	1.9	-10.0	42.0	213
	2	0.9	2.0	-10.1	34.4	169
-	2	0.99	2.0	-10.2	33.6	241
	3	0.5	1.7	-10.2	55.4	82
N3B	3	0.9	1.5	-10.4	58.2	376
	3	0.99	2.0	-11.3	6.2	29
IL-2	2	0.5	1.8	-6.5	6.4	75
	2	0.9	1.8	-7.3	1.7	226
-	2	0.99	2.0	-4.3	54.4	428
	3	0.5	2.0	-5.6	44.1	167
FRH	3	0.9	2.0	-6.6	29.6	285
	3	0.99	2.0	-4.4	60.6	430
MDM2	2	0.5	2.6	-7.9	83.1	193
	2	0.9	2.6	-7.8	90.5	225
-	2	0.99	2.9	-9.1	4.9	113
	3	0.5	3.2	-9.7	5.9	436
DIZ	3	0.9	3.1	-8.8	27.4	345
	3	0.99	2.2	-9.1	88.3	41

Table 4: Influence of the pocket radius and the weighting on the docking results (shown are the best scored docking results with RMSD $\leq$ 2 Å or the docking result with lowest RMSD)

Interestingly, for all setups the native ligand binding mode was found for BCL-X_L and IL-2. This indicates that PocketBuilder was successful in inducing the opening of native-like binding pockets on the surface of the BCL-X_L and the IL-2 proteins. An example of how PocketScanner and PocketBuilder change the apo conformation is shown in Figure 2. For BCL-X_L, the docking scores were even quite similar to that obtained in the re-docking experiment. The unsatisfying docking scores for IL-2 may be due to the fact that this binding site consists of two subpockets, that lie about 15 Å apart and with this setup, only one of these subpockets can be induced. Here, using more than one generic pocket sphere would most probably improve the docking score. However, the large relative rank of most docking results shown in Table 4 indicates that our setup does not only lead to the generation of pockets similar to those seen in the bound structure, but also to alternative pocket conformations that possess the desired properties as well. Moreover, most setups seem to prefer such alternative pockets because the best suited protein conformation is often generated quite late during the A* search. For MDM2, our method was not able to completely reproduce the native binding mode of the ligand. But when comparing the docking results listed in Table 4 to the results when docking into the apo structure (listed

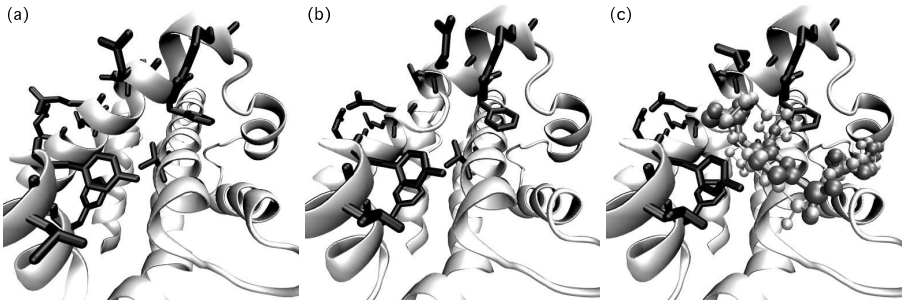


Figure 2: Change of the backbone (white cartoon representation) and the flexible residues (black licorized representation) in a subpocket of BCL-X_L (a) before running PocketScanner, (b) after running PocketScanner (pocket radius = 3 Å), and (c) after running PocketBuilder (w_{pocket} = 0.99). In (c), the ligand N3B is shown in its native conformation (thin white balls and sticks) and in its docked conformation (grey thick balls and sticks).

in Table 1), it becomes apparent that an opening of the native binding was at least partly induced. In such cases using a larger generic pocket sphere may be helpful in making the native binding pocket fully accessible.

4 Conclusion and Outlook

Accounting for protein flexibility is one of the current challenges in the protein-ligand docking field. Here, we have presented a rigorous algorithm to scan the rotameric space of residues on protein surfaces for openings of suitable ligand binding pockets. As shown for the model systems BCL-X_L, IL-2, and MDM2 a systematic scanning of the protein surface around the interface is computationally feasible. For two out of the three systems, the PocketBuilder algorithm was then able to induce pockets of suitable volumes and shapes so that the small molecule ligands may bind in a native-like orientation. By testing the algorithm on a larger number of protein-ligand complexes in the near future, we plan to tune the set of control parameters to further enhance the efficiency of this approach and the ranking of the native binding mode.

References

[ATA05] J. An, M. Totrov, and R. Abagyan. Pocketome via comprehensive identification and classification of ligand binding envelopes. *Mol. Cell Proteomics*, 4:752–761, 2005.

[AW04] M.R. Arkin and J.A. Wells. Small-molecule inhibitors of protein-protein interactions: progressing towards the dream. *Nat. Rev. Drug Discov.*, 3:301–317, 2004.

[BS00] G.P. Brady and P.F. Stouten. Fast prediction and visualization of protein binding pockets with PASS. *J. Comput. Aided Mol. Des.*, 14:383–401, 2000.

[BWF+00] H.M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T.N. Bhat, H. Weissig, I.N. Shindyalov, and P.E. Bourne. The Protein Data Bank. *Nucleic Acids Res.*, 28:235–242, 2000.

[CMK93] A. Caflisch, A. Miranker, and M. Karplus. Multiple copy simultaneous search and construction of ligands in binding sites: application to inhibitors of HIV-1 aspartic proteinase. *J. Med. Chem.*, 36:2142–2167, 1993.

[dBDBB03] P.I.W. de Bakker, M.A. DePristo, D.F. Burke, and T.L. Blundell. Ab initio construction of polypeptide fragments: Accuracy of loop decoy discrimination by an all-atom statistical potential and the AMBER force field with the Generalized Born solvation model. *Proteins*, 51(1):21–40, 2003.

[DC97] R.L. Dunbrack and F.E. Cohen. Bayesian statistical analysis of protein side-chain rotamer preferences. *Protein Sci.*, 6:1661–1681, 1997.

[EH07] S. Eyrisch and V. Helms. Transient pockets on protein surfaces involved in protein-protein interaction. *J. Med. Chem.*, 50:3457–3464, 2007.

[Goo85] P.J. Goodford. A computational procedure for determining energetically favorable binding sites on biologically important macromolecules. *J. Med. Chem.*, 28:849–857, 1985.

[HDS96] W. Humphrey, A. Dalke, and K. Schulten. VMD: visual molecular dynamics. *J. Mol. Graph.*, 14:33–38, 1996.

[HNR68] P.E. Hart, N.J. Nilsson, and B. Raphael. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Trans. on SSC*, 4(2):100–107, 1968.

[HRB97] M. Hendlich, F. Rippmann, and G. Barnickel. LIGSITE: automatic and efficient detection of potential small molecule-binding sites in proteins. *J. Mol. Graph. Model.*, 15:359–363, 1997.

[KL00] O. Kohlbacher and H.P. Lenhof. BALL–rapid software prototyping in computational molecular biology. Biochemicals Algorithms Library. *Bioinformatics*, 16:815–824, 2000.

[Las95] R.A. Laskowski. SURFNET: a program for visualizing molecular surfaces, cavities, and intermolecular interactions. *J. Mol. Graph.*, 13:323–330, 1995.

[Lea94] A.R. Leach. Ligand docking to proteins with discrete side-chain flexibility. *J. Mol. Biol.*, 235:345–356, 1994.

[LJ05] A.T. Laurie and R.M. Jackson. Q-SiteFinder: an energy-based method for the prediction of protein-ligand binding sites. *Bioinformatics*, 21:1908–1916, 2005.

[LK99] T. Lazaridis and M. Karplus. Effective energy function for proteins in solution. *Proteins*, 35:133–152, 1999.

[LLY+07] M.R. Landon, D.R. Lancia, J. Yu, S.C. Thiel, and S. Vajda. Identification of hot spots within druggable binding regions by computational solvent mapping of proteins. *J. Med. Chem.*, 50:1231–1240, 2007.

[MGH+98] G.M. Morris, D.S. Goodsell, R.S. Halliday, R. Huey, W.E. Hart, R.K. Belew, and A.J. Olson. Automated docking using a lamarckian genetic algorithm and an empirical binding free energy function. *J. Comput. Chem.*, 14:1639–1662, 1998.

[WM07] J.A. Wells and C.L. McClendon. Reaching for high-hanging fruit in drug discovery at protein-protein interfaces. *Nature*, 450:1001–1009, 2007.

Protein Structure Alignment through a Contact Topology Profile using SABERTOOTH

F. Teichert[1], U. Bastolla[2], and M. Porto[1]

(1) Institut für Festkörperphysik, Technische Universität Darmstadt,
Hochschulstr. 6-8, 64289 Darmstadt, Germany
(2) Centro de Biología Molecular "Severo Ochoa", (CSIC-UAM),
Cantoblanco, 28049 Madrid, Spain

Abstract: The contact vector (CV) of a protein structure is one of the simplest and most condensed descriptions of protein structure available. It lists the number of contacts each amino acid has with the surrounding structure and has frequently been used e.g. to derive approximative folding energies in protein folding analysis.
The CV, however, is a lossy structure representation, as it does not contain sufficient information to allow for the reconstruction of the full protein structure it was derived from. The loss of information leads to a degeneracy in the sense that a single contact vector is compatible with many different contact matrices, but it has been shown that this degeneracy is nearly fully compensated by the physical constraints protein structure is subject to.
We recently developed the alignment framework 'SABERTOOTH' that is able to generically align connectivity related vectorial structure profiles to compute protein alignments. Here we show that also the CV allows for state-of-the-art alignment quality, just like the elaborated 'Effective Connectivity' profile (EC) that SABERTOOTH currently uses. This simplification leeds to a very simple and elegant approach to structure alignment, which accelerates and generalizes the algorithm we previously proposed.
Furthermore, we conclude from our work that the CV in itself is a useful structure description if its collective properties are called for.

1 Introduction

Alignment of proteins is an every-day remit in many bioinformatics applications and many algorithms exist today that use specialized descriptions of protein structure to solve the problem in a fast and accurate way.
The task, nevertheless, has not been fully solved yet and some improvements are demanded to enhance analyses. Today three different programs are needed for the three different flavours of protein alignment, namely: structural alignment, sequence alignment, and sequence to structure alignment, often referred to as 'threading'. Tailor-made algorithms are available that are specialized for one of these tasks each. Usually, these tools are encumbered with their own often complicated description of protein structure or sequence, respectively. For a user of a software that may result in unforeseeable characteristics and capabilities of the programs, which gets even worse when a combination of two or three

different tools are used in the same project.
A desirable alignment tool would comprise all three kinds of alignments using one single algorithm on converging descriptions of protein structure and sequence that should be straightforward in definition and fast to compute.
As a first step into that direction we recently developed the 'SABERTOOTH' alignment framework [TBP07] that allows for the alignment of connectivity related structural profiles. The resulting profile alignment is highly generic and, hence, allows to input different structural and also sequence derived profiles. In a refinement step, actual coordinate data can be used to improve the alignment, if this information is available.
For the profile alignment we relied on the well understood 'Effective Connectivity' (EC) profile [BOPT08] that constitutes a generalization of the Principal Eigenvector of the contact matrix (PE) but allows for the description of complex multi-domain structures, while it is known that the PE nearly exhausively encodes the structural information of small globular folds to the extent contained in the contact matrix [PBRV04]. Besides of the inherent properties that make the EC favourable to other profiles, it is time consuming to compute since diagonalization of the underlying contact matrix is needed.
Here we assess the capacities of the contact vector (CV) of protein structure in our alignment framework. The CV can be understood as an approximation of the EC (see Fig. 1) that is very easy and fast to compute by listing the numbers of contacts each amino acids has with the structure surrounding it. In fact, the CV has a correlation coefficient of $r(\mathrm{EC}, \mathrm{CV}) = 0.94$ with the EC (for EC and CV based on a heavy-atoms contact matrix with distance cut-off $d_{\mathrm{th}} = 4.5\text{Å}$). A potential disadvantage of the CV is that it

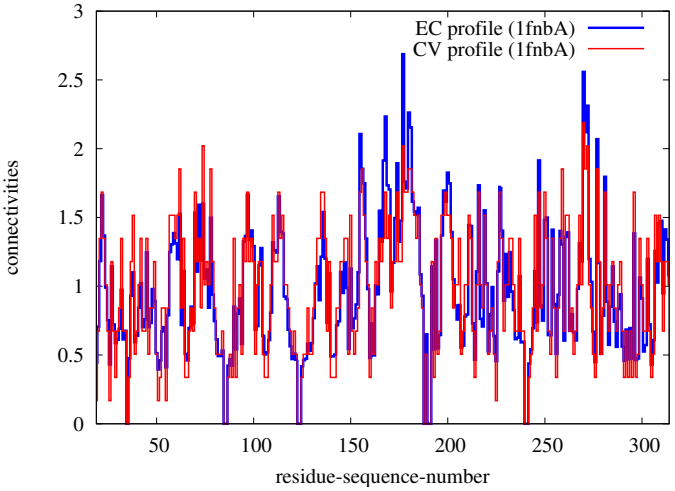


Figure 1: EC and CV profile of the structure with PDB-id '1fnbA' is shown based on a heavy-atoms contact matrix with $d_{\mathrm{th}} = 4.5\text{Å}$. The intriguing correlation between the profiles is obvious.

suffers from degeneracy. If the structure described by a contact matrix is relatively ordered and shows spatial symmetries many contact matrices comply with one and the same contact vector. In [KKVD02] the authors showed that the problem of degeneracy is partly

compensated by the distinct properties of native protein structures, i.e. constraints on the protein's backbone like volume exclusion and chemical propensities. Nevertheless, for our application that means that we have to verify the alignment results not only for accuracy in the alignment of related structures but also for the ability to descriminate true and false positives in a mixed set of related and unrelated structures.
Alongside with the move from the EC to the CV, we introduce a second simplification in the profile by changing from a heavy-atoms based contact matrix to one that is derived from the C_α trace of the protein structure, only. In our tests we found very similar performances of the different profiles independent from the choice of coordinates. The C_α description is favourable especially for cases in which the full coordinate information is not available.
To perform the verification of the alignment routine, we firstly show that the alignment results over a test set of related structures are of comparable quality for CV based and the formerly studied EC based alignments. Analysis of a set of alignments of unrelated structures demonstrates the length dependent statistical behaviour giving insight in possible problems with degeneracy.
As a final test we compare the capacities of the different alignments to sort structures according to the 'Structural Classification of Proteins' (SCOP) [MBHC95].

2 Methods

2.1 Contact Matrix, Effective Connectivity Profile, and Contact Vector

The contact matrix of protein structure is a binary symmetric $(N \times N)$-matrix where N equals the number of amino acids in the protein chain. Two amino acids i and j are assigned *in contact* $C_{ij} = 1$ if their spatial distance lies below a threshold d_{th}, or assigned *not in contact* $C_{ij} = 0$ if their distance exceeds the threshold or contacts would be trivial due to the fact that i and j are close along the protein sequence.
The notion of *distance* between amino acids can be defined in many different ways. For the use of structural analyses pairwise distances of the C_α-atoms of the protein's backbone are commonly used, while for problems that depend more on the energetics of side-chain atoms, the minimum of pairwise heavy-atom distances (i.e. other than hydrogen) are preferred.
In this publication we apply both definitions, the EC is based on heavy-atom distances with a d_{th} of 4.5Å whereas we compute the CV from a C_α contact matrix with $d_{\mathrm{th}} = 11$Å, both with three suppressed trivial diagonals, i.e. $C_{ij} = 0$ when $|i - j| < 3$.
Note that this selection is by no means necessary for our analyses, we found that the EC based on C_α distances and the CV based on heavy-atom distances perform nearly as well (data not shown) but chose the particular ones used here since they provide slighly better results. The main reason that C_α atoms are preferable from a practical point of view is that in some applications only the protein's backbone might be known. Furthermore, moving from the truly real-valued/heavy-atoms based EC profile to the integer-valued/C_α based CV accounts for the robustness of our alignment framework.

The contact vector's components CV_i are simply defined as the sum of all elements in row (or column) i of the contact matrix,

$$\mathrm{CV}_i = \sum_{j=1}^{N} C_{ij} \ .$$

The profile actually used in the alignment framework is normalized by dividing its components by the mean value of all connected (i.e. all non-zero) sites to make the components independent of chain length.
The EC, as we defined it in [BOPT08], is a member of the 'Generalized Effective Connectivity' (GEC) family of protein sequence and structure profiles. Like all members of this family, it shares the properties that (a) it maximizes the quadratic form $Q = \sum_{ij} C_{ij} c_i c_j$, (b) its mean value is fixed to $\langle c \rangle = 1$ to choose a normalization of its components, (c) its mean square component is fixed to $\langle c^2 \rangle = B > 1$. The corresponding B for the EC is set to $B = \langle \mathrm{CV}^2 \rangle / \langle \mathrm{CV} \rangle^2$ with the contact vector CV_i.
The EC profile can as well be expressed as a weighted sum of eigenvectors of the contact matrix C_{ij}, with weights gradually introducing contributions from more vectors from C_{ij}'s eigensystem when structures described get more modular. Consequently, the values of the components of the EC measure the importance of amino acid i for the global connectivity of the protein structure.
We also showed [BOPT08, TP06] that the EC is nearly identical to the Principal Eigenvector of the contact matrix (which is a member of the GEC family itself), for small single-domain structures with low internal modularity. The PE, in turn, allows for the reconstruction of its contact matrix, hence, its structure with an accuracy comparable to typical X-ray experiments making it a representation of protein structure that is equivalant to atomic coordinates [PBRV04].

2.2 The Alignment Framework 'SABERTOOTH'

The alignment framework introduced in [TBP07] translates the task of finding a proper alignment of two protein structures into the recognition of similar connectivity patterns in the vectorial profiles corresponding to the structures. This analogy is grounded on the observation that the structural profile is conserved in protein evolution, like the overall topology of the protein structure that it describes.
In this way, we can use fast and simple comparison algorithms on the condensed profiles, while relevant non-local properties of protein structure are retained. Moreover, the resulting alignment is little dependent on spurious local similarities that could obliterate the recognition of far homologs. However, these local structural details can be reintroduced in a second step, in order to obtain a more precise structural match.
Following this idea, we developed a structural alignment routine that consists of two steps. First, the alignment of the structural profiles is used to recognize global similarities. Second, a refinement step employs the atomic coordinates in order to improve the local structural superimposition.

2.2.1 Alignment Algorithm

The profile alignment was designed similarly to 'traditional' sequence alignment routines like e.g. dot-matrix alignments. We represented every possible alignment of two proteins by a path through an alignment matrix. Possible alignments were defined as the line-up of two amino acid chains, together with an arbitrary number of inserted gaps of arbitrary length.
The optimum alignment path minimizes a cost function based on the profiles' components and a set of parameters that are analogous to traditional 'substitution probabilities' for alignments and 'open/extend' penalties for gaps. However, in contrast to those, the penalties used here are directly dependent on the structures through their explicit dependence on the profile components.
In order to assess the quality of the resulting alignment, we apply the standard MaxSub routine [SERF00] to the set of aligned residue pairs and compute the optimal rigid body rotation and translation that maximize the spatial superimposition of the two proteins. This allows for the calculation of standard similarity scores based on coordinates and for producing spatial views of the alignment.
Through the MaxSub routine and the set of aligned residues, we derive the optimally superimposed set of coordinates, and from that we compute pairwise distances of all combinations of amino acids connecting the two protein chains. This detailed local information can then be exploited in a second alignment step in order to refine the alignment itself, similar in principle to what other structural alignment algorithms do.
The set of amino acids effectively close in space is analyzed and subsequently used to restrict the possible paths through the alignment matrix, so that the second run searches for the optimal alignment only around these identified groups of close pairs. It incorporates close pair groups into the alignment where unambiguously assigned, it picks out the best choice in cases where more than one alternative is present, and it simply minimizes the path cost as before in areas that are not constrained. Obviously, this kind of refinement is only able to improve the input alignment if the initial spatial superposition was already close to optimal.
After the refinement step, a second run of the MaxSub algorithm is used to obtain the optimal spatial superimposition through which we assess quality and significance of the final alignment. Among other scores a Z-score measuring the statistical significance of the alignment is computed from the Percentage of Structural Identity (PSI) by eliminating the inherent length dependency of the latter.
For more details on the alignment algorithm, cost functions, and parameters please refer to [TBP07].

2.3 Alignment Quality Assessment

In [TBP07] we presented an automatic routine to assess the quality of the alignments produced by our algorithm, as well as of alternative ones produced by well established programs. To do so, we measure the quality of alignments by applying SABERTOOTH and reference tools to a test set of 3566 alignments of distantly related protein pairs by

means of different scores including PSI, contact overlap, and sequence similarity. The structures in the test set are derived from the '29SCOPsf' set described in more detail in [LMLRL$^+$05]. The set consists of 525 structures from 29 SCOP [MBHC95] superfamilies (release 1.69) that constitute a representative collection of common structural motifs. All superfamilies are from different folds of the SCOP classification, and cover the four major SCOP classes all alpha, all beta, alpha+beta, and alpha/beta.
In [TBP07] we could show that SABERTOOTH performs state-of-the-art alignments using the heavy-atoms based EC profile.
In this publication we adopt the same alignment quality assessment routine which makes the results presented here directly comparable to those in our previous publication.

3 Results

3.1 Comparison of Alignment Qualities

The alignment results over the test set of distantly related structures are very similar for EC and CV based alignments. The PSI distributions are depicted in the histograms in Fig. 2 along with the differences in PSI for direct comparison.
The EC profile achieves $\langle \mathrm{PSI}_{\mathrm{EC}} \rangle = 68.2$ while the CV based alignment performes slightly better, resulting in $\langle \mathrm{PSI}_{\mathrm{CV}} \rangle = 69.1$.

3.2 Classification Capacities Assessment

Measuring the capacities of an alignment program to reproduce the SCOP classification constitues a challenging benchmark. Accurately computed alignments are the basis for the assignment of a Z-score that assesses the statistical significance of a given alignment independent of chain lengths. This is only possible if alignments of related structures can be clearly distinguished from unrelated ones.
This attribute can be visualized by an algorithm's behaviour when aligning a set of unrelated structures. The resulting PSI of unrelated pairs plotted versus length of the shorter chain should follow a power-law decay for increasing chain lengths. Figure 3 shows that both profiles perform well in this task and, hence, allow for the definition of proper Z-scores. By fitting a power-law for mean PSI and standard deviation we define the Z-score

$$Z = \frac{\mathrm{PSI} - \langle \mathrm{PSI} \rangle}{\sigma_{\mathrm{PSI}}}$$

with

$$\langle \mathrm{PSI}_{\mathrm{EC}} \rangle = 501.9 \cdot \min(N_1, N_2)^{-0.714} \text{ and } \sigma_{\mathrm{PSI}_{\mathrm{EC}}} = 541.4 \cdot \min(N_1, N_2)^{-0.945}$$

$$\langle \mathrm{PSI}_{\mathrm{CV}} \rangle = 493.0 \cdot \min(N_1, N_2)^{-0.711} \text{ and } \sigma_{\mathrm{PSI}_{\mathrm{CV}}} = 555.6 \cdot \min(N_1, N_2)^{-0.947} .$$

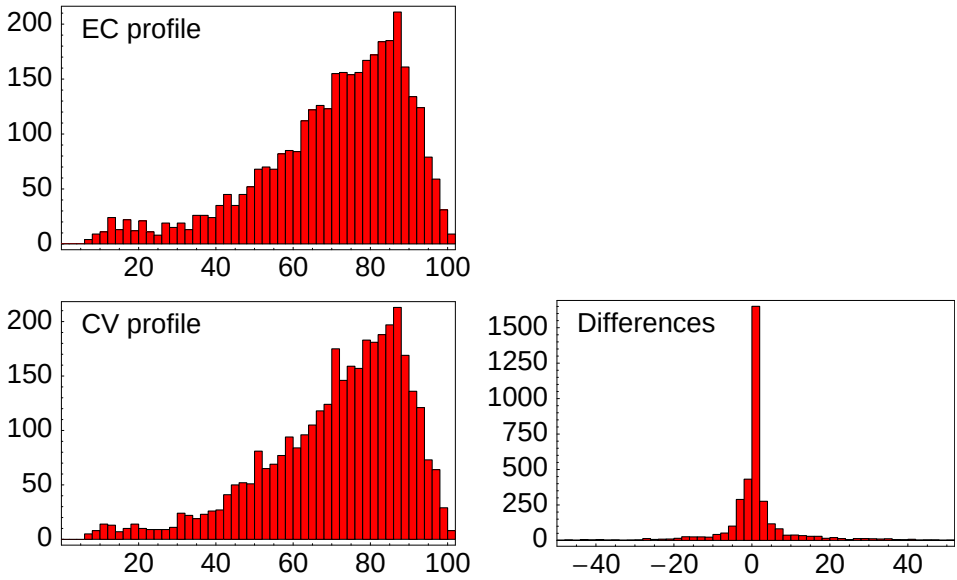


Figure 2: The upper left histogram shows the distribution of PSI values found with EC based alignments as output by SABERTOOTH. The lower left histogram shows the results of the CV based alignments on the same set. The right histogram shows the distribution of the differences $\mathrm{PSI}_{\mathrm{CV}} - \mathrm{PSI}_{\mathrm{EC}}$.

The acutal fold classification capacities are shown in the ROC-plot in Fig. 4. The curve unveils the sensitivity and the generality properties of the Z-score to judge whether the structures in an alignment belong to the same fold in SCOP. The better the classification the larger the area under the curve, i.e. the farther the curve separates from the diagonal line of random guessing.
The set consists of 498 structures that were randomly selected from the 97 largest folds in SCOP (version 1.73) having less than 40% sequence identity. It was assembled by selecting $1/11$ of the structures of all folds with 22 or more members in the ASTRAL40 [CHW$^+$04] database. All-vs-all alignment generates 123753 alignments of protein chains with known SCOP relation.

4 Conclusions

We could show that the very condensed and simple but also lossy representation of protein structure as a contact vector still contains sufficient information to perform structural alignments. Furthermore, the behaviour with unrelated structures is very similar to that of the more sophisticated EC profile. This means that the degeneracy the CV suffers from does not play a major role for this application. This remains true even after reduction of the input data from heavy-atom coordinates to a C_α description.

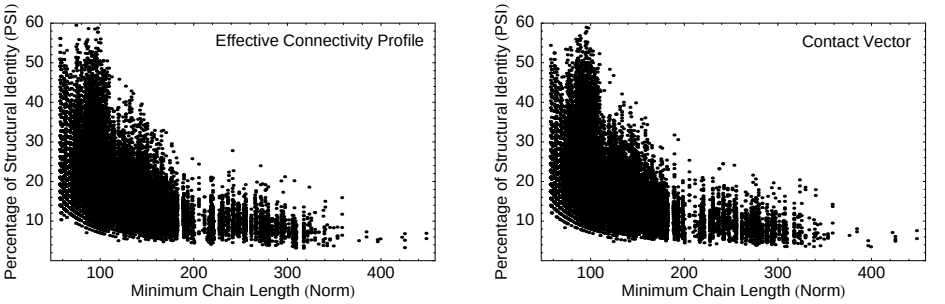


Figure 3: PSI versus minimum chain length for EC (left figure) and CV (right figure). Both plots show the same power-law length dependence for applying SABERTOOTH on a set of unreated structures. Well defined Z-scores can be computed for both distributions.

The slightly superior performance of the CV in our alignment framework, in comparison with the EC, together with its lower computational cost persuaded us to move to the CV as the standard structure representation for our alignment program SABERTOOTH (refer to `http://www.fkp.tu-darmstadt.de/sabertooth/`).
Moreover, from our analyses we conclude that the CV, just as being so simple to compute, might be a better description for analyzing collective properties of protein network topology than one could expect.

5 Funding

FT and MP gratefully acknowledge generous financial support from the Deutsche Forschungsgemeinschaft via project PO 1025/1 and from the Deutscher Akademischer Austauschdienst via project D/06/12858. UB acknowledges financial support from the Spanish Education and Science Ministry through the Ramón y Cajal program and the grant no. BIO2005-05786 and from CSIC through the Acciones Integradas program.

6 Acknowledgements

We are publishing the present paper in remembrance of Angel Ramirez Ortíz (1966-2008), with whom we had the privilege to have inspiring discussions and numerous hints that we gratefully acknowledge, while we deeply miss his friendship and advice.
We thankfully acknowledge Alejandra Leo-Macías for the manually refined 29SCOPsf set.

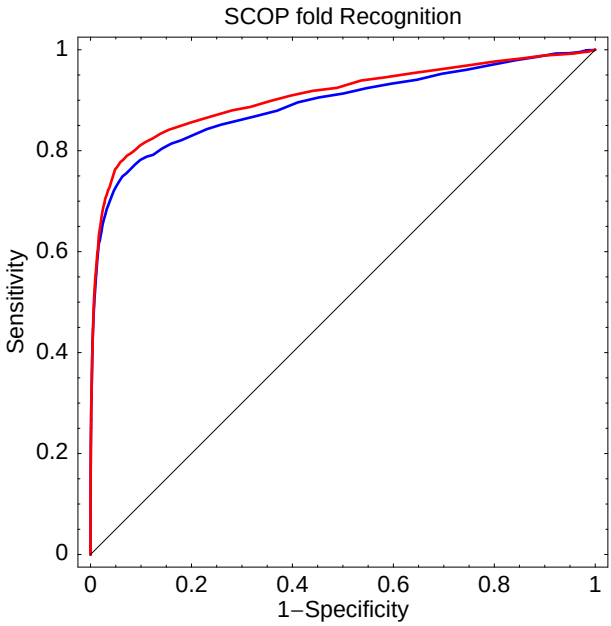


Figure 4: Fold recognition capacities for SABERTOOTH alignments using the EC profile (blue curve) and the CV profile (red curve). It turns out the the CV performs slightly better here than the EC.

References

[BOPT08] Ugo Bastolla, Angel R. Ortíz, Markus Porto, and Florian Teichert. Effective connectivity profile: A structural representation that evidences the relationship between protein structures and sequences. *PROTEINS: Structure, Function, and Bioinformatics*, 2008. (in print; doi: 10.1002/prot.22113).

[CHW+04] J.M. Chandonia, G. Hon, N.S. Walker, L. Lo Conte, P. Koehl, M. Levitt, and S.E. Brenner. The ASTRAL compendium in 2004. *Nucleic Acids Research*, 32:189–192, 2004.

[KKVD02] A. Kabakçioglu, I. Kanter, M. Vendruscolo, and E. Domany. Statistical properties of contact vectors. *Physical Review E*, 65(4):41904, 2002.

[LMLRL+05] Alexandra Leo-Macias, P. Lopez-Romero, D. Lupyan, D. Zerbino, and Angel R. Ortiz. An analysis of core deformations in protein superfamilies. *Biophys J*, 88(2):1291–1299, 2005.

[MBHC95] A. G. Murzin, S. E. Brenner, T. Hubbard, and C. Chothia. SCOP: a structural classification of proteins database for the investigation of sequences and structures. *Journal of Molecular Biology*, 247:536–540, 1995.

[PBRV04] Markus Porto, Ugo Bastolla, H.-Eduardo Roman, and Michele Vendruscolo. Reconstruction of Protein Structures from a Vectorial Representation. *Physical Review Letters*, 92(21):218101, 2004.

[SERF00] Naomi Siew, Arne Elofsson, Leszek Rychlewski, and Daniel Fischer. MaxSub: An automated measure for the assessment of protein structure prediction quality. *Bioinformatics*, 16(9):776–785, 2000.

[TBP07] Florian Teichert, Ugo Bastolla, and Markus and Porto. SABERTOOTH: protein structural alignment based on a vectorial structure representation. *BMC Bioinformatics*, 8:425, 11 2007. Available online at `http://www.fkp.tu-darmstadt.de/sabertooth/`.

[TP06] F. Teichert and M. Porto. Vectorial representation of single-and multi-domain protein folds. *The European Physical Journal B-Condensed Matter and Complex Systems*, 54(1):131–136, 2006.

Registration to a neuroanatomical reference atlas - identifying glomeruli in optical recordings of the honeybee brain.

Martin Strauch[1*] , C. Giovanni Galizia[1]
1: Department for Neurobiology, University of Konstanz, Germany
*: Martin.Strauch@uni-konstanz.de

Abstract: An odorant stimulus given to a bee elicits a characteristic combinatorial pattern of activity in neuronal units called glomeruli. These patterns can be measured by optical imaging, however detecting and identifying the glomeruli is a laborious task and prone to errors. Here, we present an image analysis pipeline for the automatic detection and identification of glomeruli. It involves Independent Component Analysis (ICA) to detect glomeruli in CCD camera data, a filtering step to exclude non-glomerulus objects and a graph-matching approach to find the best projection of the observed brain region onto a reference atlas. We evaluate our method against a manual glomerulus identification performed by a human expert and show that we achieve reliable results. Employing our method, we are now able to screen multiple recordings with the same accuracy, yielding a homogeneous collection of glomerulus identity mappings. These will subsequently be used to extract activity patterns that can be compared between individuals.

1 Introduction

The olfactory sense of insects is a popular model for studies on signal perception, learning and memory formation. It is not only of biological interest, but given an insect's remarkable performance in discrimination, as well as generalisation of chemicals, unveiling the principles behind insect olfaction could also be important for chemoinformatics or classification algorithms in general [LD06, GSSG05].

Briefly, information about odorants is generated by receptor cells, each of which has a certain receptive range, i.e. responds to several chemicals with varying strength. The honeybee *Apis mellifera* has tens of thousands of receptor cells, but only 160 different types of receptors. It is in the antennal lobe (AL), the first instance of olfactory information processing in the insect brain, that the information from all these receptors is assembled, forming a code word that depends on the input odorants, their relative proportion and overall concentration.
The olfactory code is based on glomeruli, lumps of coherently acting neurites in the AL, as basic units of information. Each of them receives input from one of the 160 receptor types (in the honeybee; numbers vary between species). Together, they form a combinatorial code, each input generating a characteristic pattern of glomerulus activity, i.e. code word, which is conserved for all members of a species [GM00, GSRM99].

In calcium-imaging experiments, a calcium-sensitive fluorescent dye is used to report changes in intracellular calcium concentration, which is indicative of neuronal activity. Thus, using a CCD camera to record changes in fluorescent light intensity, we are able to monitor glomerulus activity patterns in response to defined input odorants. For details on the experimental technique please refer to [SG02, GV04].

Any evaluation of glomerular activity patterns across individuals requires an accurate glomerulus identification. As no rigorous method has been proposed for this task as of today (see [SG02, GV04] for previous, manual approaches) , we present here a method for the automatic detection and subsequent identification, i.e. labelling, of glomeruli in the honeybee antennal lobe.

First, we need to detect glomeruli in CCD camera data, i.e. find the pixels which together constitute a glomerulus. We therefore employ Independent Component Analysis [HO00] to detect pixels that change coherently in front of a random background, allowing us to compile a map of the objects observed during the experiment (section 2.1). We then denoise the map and apply a shape-filter based on anatomical criteria to exclude objects that are not glomeruli (section 2.2).

The identification of the glomeruli then amounts to a registration of the glomerulus map to the 3D reference atlas of the honeybee AL [GMM99] where glomeruli are assigned unique labels. Previously, the identification of glomeruli has been a laborious task for human experts, prone to variations in accuracy. A 2D view of the AL as provided by the camera recordings has to be projected onto the 3D atlas, a task that is complicated by experimental noise and biological variability that often leads to changes in glomerulus position in individual bees. Moreover, several of the glomeruli present in a slice from the atlas may not be visible in the actual data, as they do not respond to any of the tested odorants or because they accidentally remained unstained during the experiment.

In order to identify the glomeruli, we transform both the map and the 3D atlas into adjacency graphs with glomeruli as nodes (section 2.3). Using an exact *Branch&Bound* approach for graph matching, we find the projection of the 2D map onto the 3D atlas which comes closest to representing a subgraph isomorphism. Due to the aforementioned biological variability, an exact isomorphism may not always exist. We account for this variability by scoring the graphs according to a penalty matrix for deviations from the atlas geometry.
The matching we perform is related to *subgraph isomorphism*, which is known to be NP-complete [GJ90]. Thus, a complete search that takes all possible combinations of node projections and positional variations into account will be time-consuming. In practice, however, graphs are moderately sized and *a priori* knowledge is often available that helps to efficiently speed up the search process (see section 2.3).

For evaluation, we compare our method to a ”ground truth” on a dataset manually examined by a human expert and find that it achieves reliable results (section 3). Given the deterministic nature of the graph-matching algorithm, we suggest that it is better suited for an integrative analysis of multiple datasets than different manual, often undocumented, algorithms performed by the respective experimentators.

2 Materials and Methods

2.1 Detecting glomeruli with ICA

Independent Component Analysis (ICA) is a method for feature extraction and blind source separation [HO00]. ICA has already found applications in neuroscience, e.g. in the fields of EEG and MEG recording analysis, where it is used to separate the actual signals of interest from artifacts such as muscle or eye activity [VSJ+00].
In the ICA paradigm it is assumed that a number of observers record several independent signals, which, due to the recording situation, can occur to them as mixtures and may be obscured by noise. While in the EEG application the focus was on the different temporal contributions of each component, we are interested in spatial components that change their activity coherently over time. Here, the independent signals correspond to glomeruli: each glomerulus sends a signal, i.e. its state of activation, which is obscured by experimental noise. Additionally, signal superposition may occur in some cases if glomeruli lie on top of each other.
The task for the ICA is to separate signals from noise and thus to detect glomeruli. We interpret all the pixels of the CCD camera recording as observers $\mathbf{x}(t)$, which, at time instant t, perceive n signals $(s_1(t), \ \dots \ , s_n(t))$ that are transformed by vectors $(\mathbf{a}_1, \ \dots \ , \mathbf{a}_n)$, which can, in an abstract sense, be regarded as the parameters of the recording situation, e.g. properties of the dye, the camera etc. The number of glomeruli may be smaller than n, as other objects can also generate signals.

$$\mathbf{x}(t) = \sum_{i=1}^{n} \mathbf{a}_i s_i(t) = \mathbf{A}\mathbf{s}(t) \tag{1}$$

The ICA problem is to estimate the source signals and the coefficients of the so-called mixing matrix **A** based on the (ideal) assumption that the signals are statistically independent and non-gaussian. In practice, however, the strict independence requirement may be relaxed.
In our case, pixels belonging to the same glomerulus will be considered as carrying the same signal $s_i(t)$. Two pixels from different glomeruli will display different behaviour over time and they will be associated with two different signal components $s_i(t)$ and $s_j(t)$.

While exact solutions to the ICA problem are computationally expensive, several efficient algorithms for approximate solutions are available, e.g. the popular *fastICA* algorithm [HO00]. We implemented our approach in Java, employing the open-access platform KNIME (*www.knime.org*) that supports data exchange with R (*www.r-project.org*) . For the ICA we could thus make use of the R-package *fastICA* [MHR07] that implements the mentioned ICA algorithm.

We found that 2000 iterations of *fastICA* were sufficient to obtain stable, reproducible results. We set the number of expected independent components to 50, which is greater than the number of about 20-30 glomeruli that is commonly found in one 2D view of the antennal lobe. This is to account for the fact that other, non-glomerulus structures are often accidentally stained with the fluorescent dye, giving rise to artifact signals that are

separated out by the ICA as further independent components. A subsequent filtering step (see section 2.2) was employed to tell apart glomeruli and other objects.

Each independent component identified by the ICA resulted in an image with the dimensions of the original video data with one glomerulus (or other object) enhanced but not yet neatly cut out (see inlay in *Figure 2b*). For construction of the glomerulus map, we thus separated signal and residual noise in each of these images by regarding only those pixels as signal, whose values were above the upper whisker ($1.5 \times$ IQR above the 3rd quartile) of the box-plot of all pixel values in the image. Overlaying and false-coloring all components we then could construct a glomerulus map of the observed part of the AL (*Figure 2b*).

2.2 Filtering

As previously mentioned, objects other than glomeruli can also appear in the recordings. As this complicates the identification, we thus aimed at cleaning up the map such that only glomeruli remain. For this step, we employed two anatomically motivated criteria, namely object size and circularity. Glomeruli are relatively large, more or less globular objects. In the two space dimensions of calcium-imaging recordings they appear to be roughly circular. Non-glomerulus structures, on the other hand, result e.g. from experimental noise, yielding smaller, scattered objects or may be parts of trachees or the antennal nerve, which are rather elongated structures.
For the filtering we thus demanded a minimum size, i.e. a threshold t_p denoting the number of contiguous pixels. Circularity was measured by drawing a circle around the object's centroid. The radius was set to half the longest diameter of the object. The degree of circularity $degree_c$ was then measured as the ratio

$$degree_c = \frac{\#\text{ pixels (object)}}{\#\text{ pixels (circle)}} \tag{2}$$

For the data analysed in this study we set the default parameters to $t_p = 50$ pixels and the circularity threshold $t_{degree_c} = 0.6$. For objects with more than 3 neighbours we reduced the circularity threshold to $\frac{1}{2}\ t_{degree_c}$ to account for the fact that in dense regions of the recording glomeruli actually lie on top of each other, thus obscuring parts of the circular shape.
Filtering by size also results in de-noising as a side-effect. While the circularity threshold reflects mainly biological proportions, t_p is more dependent on the resolution of the recording and needs to be adjusted for other datasets. As there is variation between individuals, we had to lower both thresholds ($t_p = 40$ and $t_{degree_c} = 0.5$) for some animals in order not to discard glomeruli. To automatise this step, one could define thresholds relative to the average size and circularity of objects in the recording.

2.3 Registration to the atlas with graph-matching

We transformed the previously constructed AL map into a topological graph $S = (V_S, E_S)$ with glomeruli as the set of nodes V_S. Edges E_S between two nodes were drawn if the respective glomeruli touched each other. Edges were annotated with the relative position of the glomeruli's centroids to each other based on the angle between them on a polar grid (see *Figure 1a*). We allowed 8 categories of positional information, i.e. $0°$, $45°$, ... $315°$.

The same positional annotation applied for the atlas graph $G = (V_G, E_G)$ which was based on the topological information contained in the reference atlas of the honeybee AL [GMM99]. Taking into account that the 2D view of the AL represents not a perfect 2D slice but rather the focal plane that may also contain glomeruli from above and below, we accepted also glomeruli touching on the z-axis as neighbouring. For edge annotation, however, we used the same 8 categories of 2D positional information, neglecting the position on the z-axis.
Additionally, a systematical source of variation was taken into account: all members of one of the rosetta-like glomerulus clusters in the atlas, were also regarded as neighbours. Some of these glomeruli appear actually distant in the atlas, but may frequently collapse onto each other in case this is not prevented by solid obstacles such as other glomeruli or structural elements in the AL. These anatomical variations are discussed in [GMM99]. We consequently chose to connect the members of these clusters by edges, but marked them as facultative edges.

Including these rules we ensured that the set of atlas graph edges always remains a superset of the set of map graph edges: $E_S \subseteq E_G$. Having more edges in the atlas graph than expected based on biological knowledge, we could consider every edge from E_S which was also in E_G to be consistent with the atlas topology. Conversely, every edge from E_S which was not in E_G was regarded as a violation of the atlas topology.

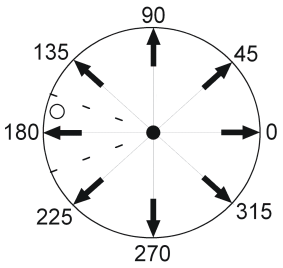
We employed a topological score to measure for a candidate subgraph S' the consistency of its edge set $E_{S'}$ with the atlas edge set E_G. A scoring or penalty matrix (*Figure 1b*) was used as a parameter for the amount of biological variation that should be allowed. Slight deviations from the atlas topology received only slight penalties ($p = 0.25$), whereas gross violations of the atlas topology, such as a mismatch $E_{S'} \ni E' : \ 0° \ , \ E_G \ni E : \ 180°$ received a larger penalty ($p = 1$). Facultative edges were allowed but at an additional cost of $p = 0.25$. The scoring matrix can be regarded as a parameter specifying the variability of glomerulus positions in the honeybee AL. For other species, different parameter settings may apply.

Having defined the scoring matrix, we could search for an isomorphic or close to isomorphic projection of the map subgraph onto the atlas graph, taking a low overall score (the sum of the individual edge scores) as indicative of a good match.
Using a depth-first *Branch&Bound* search strategy, we chose a seed node from the map graph and subsequently assigned all atlas nodes to it, at each step computing all possible assignments of children nodes of the two nodes to each other. All of these assignments were scored and followed on iteratively if the current partial score did not exceed the

overall minimum score for a complete projection and if no gross violation, i.e. $p \geq 1$ occurred. The latter helped to identify obvious orientation mismatches at an early stage.

Using the above search strategy, we were able to efficiently enumerate all valid (with no $p \geq 1$ for any node; consistent node assignments, e.g. no atlas graph node assigned twice) projections. In practice, we kept all valid projections in a search tree and scored all possible paths through the tree, starting at the root, identifying the least scoring path as the desired best approximation to an isomorphism.
The approach described above works without any additional information. However, in order to reduce the number of isomorphic solutions we made use of *a priori* knowledge in the form of a marker glomerulus. Chemical substances such as nonanol can be employed as markers as they elicit a characteristic response, activating one glomerulus - the marker glomerulus - far more than the others in the observed region of the AL. In these cases, we thus knew the correct projection of one of the nodes *a priori*. Hence, we could introduce a constraint to better tell apart solutions with similar scores. Additionally, this provides an excellent seed for the above graph matching approach.

a)



b)

map \ atlas	∉	↑	↓	←	→	↖	↘	↗	↙
∉	0	0	0	0	0	0	0	0	0
↑	1	0	1	0.75	0.75	0.25	1	0.25	1
↓	1	1	0	0.75	0.75	1	0.25	1	0.25
←	1	0.75	0.75	0	1	0.25	1	1	0.25
→	1	0.75	0.75	1	0	1	0.25	0.25	1
↖	1	0.25	1	0.25	1	0	1	0.75	0.75
↘	1	1	0.25	1	0.25	1	0	0.75	0.75
↗	1	0.25	1	1	0.25	0.75	0.75	0	1
↙	1	1	0.25	0.25	1	0.75	0.75	1	0

Figure 1: a) Topological coding: The edge between the black and the white node is annotated with 180° ($\leftarrow$). b) Penalty (scoring) matrix defining the penalty for mismatches between edge annotation in the atlas graph and the map subgraph. If an edge exists in the map (e.g. $\uparrow$), but not in the atlas ($\notin$), this results in a high penalty of $p = 1$. Slight positional variabilities are given lower penalties. As we assume $E_S \subseteq E_G$, atlas edges are allowed to be missing in the map ($p = 0$, see first row).

2.4 Dataset used in this study

The dataset used for the evaluation of our method was taken from [Dit05]. Bee preparation and staining with the calcium-sensitive fluorescent dye fura-dextran were performed as described in [SG02]. Monochromatic light was used for the excitation of the dye. For each odorant measured, two times 40 images (two sequential, quasi-simultaneous images for 340 nm and 380 nm) were taken at a frequency of 5 Hz and a spatial resolution of 2×2 μm per pixel using a TillPhotonics CCD camera and an Olympus BX50WI microscope fitted with a $20\times$ objective lens (TillPhotonics, Germany; Olympus, Germany).
The images contained in the evaluation dataset were constructed using the pixel-wise ratio

of the 340 nm and 380 nm images. The resolution of the images was 160×120 pixels. 16 aliphatic hydrocarbons at 4 different concentrations (10^{-1} to 10^{-4} dilution in mineral oil) were used as odorants. The odorants were presented to the bees from image 11 to 15 of the 40 images recorded.

For glomerulus detection, we concatenated all 40-image recordings that were done in the same animal in order to see as many glomeruli active as possible. Depending on the number of odorant responses measured, this amounted to between 600 and 1200 images per animal. Manual detection and identification of glomeruli was performed as described in [Dit05] using the honeybee AL reference atlas and one marker glomerulus.

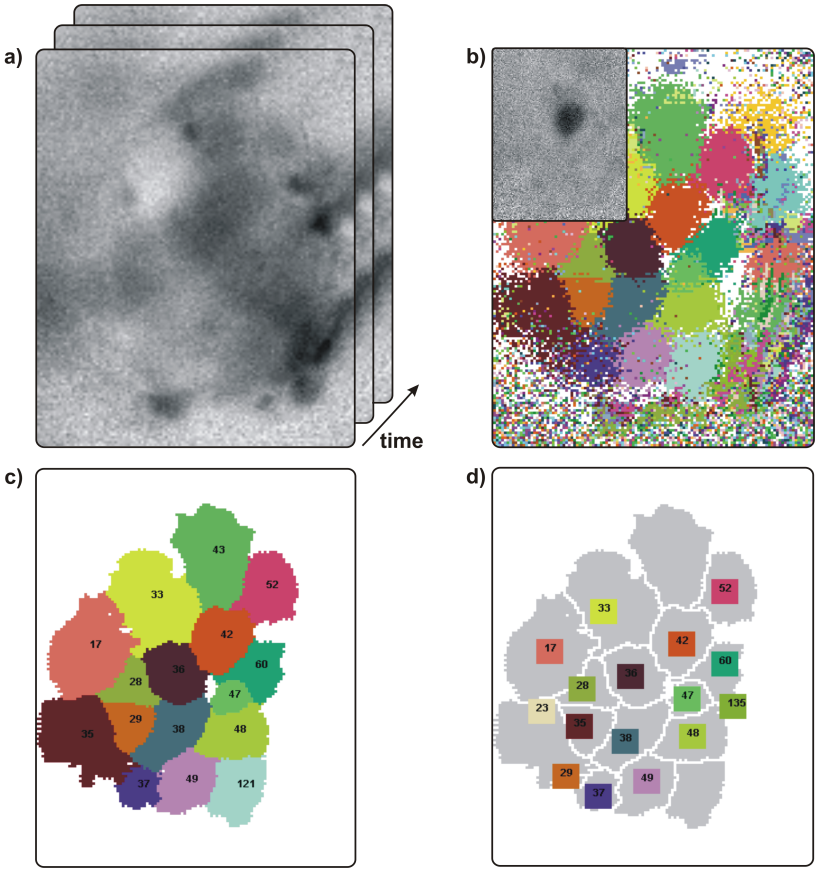


Figure 2: ICA is performed on camera recordings (animal 12) (a) to obtain a map of the AL (b). The inlay in b) shows one of the independent components that were used to construct the map. Through filtering, non-glomerulus objects are discarded. A graph-matching approach identifies the glomeruli, resulting in a labelled AL map (c). For comparison, a human expert labelling is shown (d).

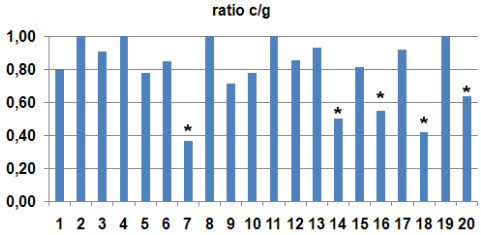


Figure 3: Performance for 20 different animals. The bars represent the ratio $\frac{c}{g}$, where c is the number of glomeruli which received a common label in both the manual and the computed solution, while g stands for the total number of glomeruli.

3 Results

We produced labelled AL maps for a dataset consisting of 20 different animals. An example showing the individual processing steps of our method is given in *Figure 2*. Reproducibility of the maps was generally high. For each animal, 10 runs of the ICA and subsequent filtering were performed to generate 10 maps per animal. The average pairwise pixel overlap between the maps from the same animal was 94%. Deformations of individual glomeruli occurred, the overall map layout, however, was always the same.

To assess the quality of our glomerulus identification, we evaluated the overlap of the computed and the hand-made labelling. The hand-made labellings [Dit05] provided us with coordinates of glomerulus centroids as the human expert saw them (no shape information was recorded). We placed the centroids onto the corresponding maps (see *Figure 2d)* and counted all matches, i.e. positions where both our computed maps and the hand-made labelling agreed on having detected a glomerulus. We defined this as the number of glomeruli g that were visible in the respective AL. Cases like glomeruli 43/121 and 135 which exist only in one of the maps (*Figure 2c/d*) did not influence g.

In order to evaluate the performance of our method we computed the ratio $\frac{c}{g}$, where c is the number of glomerulus labels both maps had in common. Thus, two labellings perfectly in accordance with each other resulted in a ratio of 1.
On average, we achieved a ratio of 0.79 with 5 labellings being 100 percent correct (see *Figure 3*). A typical example is shown in *Figure 2c*, where 14 glomeruli were detected by both the computational and the manual approach. Here, 12 glomeruli received the same label, resulting in a ratio of 0.86. In this example, the source of disagreement between manual and computed solution is the different perception of glomerulus 35 (computed) which was identified as two glomeruli (23 and 29) by the manual approach.

Typically, minor disagreements on the identity of a single glomerulus occured in the outer regions of the maps, where glomeruli have less neighbours and thus less geometrical constraints, giving rise to multiple solutions of equal or similar probability. When ratios were below 0.6, this was the result of follow-on mistakes: if a glomerulus more towards

the middle of the map received a wrong label, adjacent glomeruli were consequently shifted compared to the manual solution, which explains the relatively high number of disagreements in those cases (marked with * in *Figure 3*). These solutions received, however, slightly better scores than the correct solution. Post-hoc human identification clarified that in all of these cases the computed solution was also compliant with the reference atlas. In order to resolve this, one would need a new marker, i.e. another constraint in a different region of the AL to further reduce the number of similarly scoring projections.

Computing time for the graph matching differed depending on the number of glomeruli visible in a particular 2D view of the AL (minimum 5, maximum 20, median 12) and the uniqueness of the correct solution. In case of many equally well scoring projections, all of them had to be followed deep into the search tree, preventing an early bound. On a 4 CPU machine ($4\times$ Intel® Xeon® 2.33 GHz), computing time for the data described in this work was in a range between below one second up to about 13 minutes. The ICA took on average 3-4 minutes on a desktop computer.

4 Conclusion

We have proposed an image analysis method for the detection and identification of glomeruli in the honeybee AL. The average accuracy of its results lies well in the range of human performance, however in some cases the best score according to the given scoring matrix did not correspond to the projection chosen by the human expert. These results leave room for future optimisations of the parameters, i.e. the connectivity of the atlas graph and the scoring matrix. Both reflect biological properties, i.e. anatomy and anatomical variability and could be learned from training data.
Further, we accepted the manual glomerulus identification as a ground truth for evaluation purposes. Although the manual identification was performed very carefully, errors may nevertheless have occurred. The manual identification is usually very reliable close to the marker glomerulus (17 or 33 in this dataset), however accuracy decreases with increasing distance from the fixed point indicated by the marker.
In order to improve the quality of glomerulus identification, both manual and computational, it may thus be necessary to use multiple marker glomeruli to cover the relevant parts of the AL.

Computation time is generally unproblematic, however it could become an issue for real-time application of the method while the experiment is running and for datasets where more glomeruli are visible. Future improvements could also involve extensive preprocessing of the atlas graph, making use of the fact that it does not change between the experiments. This would allow for polynomial time subgraph matching against the preprocessed atlas graph, however at the cost of increasing space requirements [MB99].

In summary, we have devised a method that allows to deterministically identify glomeruli in calcium-imaging recordings of the honeybee AL. By employing a penalty matrix for deviations from the atlas geometry we have introduced a rigorous definition of a good

registration fit to replace hand-made glomerulus labellings. Our graph-matching approach allows to automatically search for the optimal solution based on the score criterion.

With the image analysis pipeline at hand it now becomes possible to screen multiple datasets with the same accuracy. Glomerular activation patterns from different studies can be combined to build a data pool that will subsequently be used as a resource for data mining approaches aimed at understanding information processing in insect brains.

5 Acknowledgements

We thank Mathias Ditzen for offering us his dataset and the results of the manual glomerulus identification [Dit05]. We also acknowledge helpful discussions with Michael Berthold and Julia Rein. This work was supported by the German Ministry for Education and Science (BMBF grant 576/07) and partially supported by the DFG Research Training Group GK-1042 "Explorative Analysis and Visualization of Large Information Spaces".

References

[Dit05] M. Ditzen. *Odor concentration and identity coding in the antennal lobe of the honeybee Apis mellifera*. PhD thesis, Freie Universität Berlin, URL: http://www.diss.fu-berlin.de/2005/211/indexe.html, 2005.

[GJ90] M.R. Garey and D.S. Johnson. *Computers and Intractability; A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York, NY, USA, 1990.

[GM00] C.G. Galizia and R. Menzel. Odour perception in honeybees: coding information in glomerular patterns. *Curr Opin Neurobiol*, 10(4):504–510, Aug 2000.

[GMM99] C.G. Galizia, S.L. McIlwrath, and R. Menzel. A digital three-dimensional atlas of the honeybee antennal lobe based on optical sections acquired by confocal microscopy. *Cell Tissue Res*, 295(3):383–394, Mar 1999.

[GSRM99] C.G. Galizia, S. Sachse, A. Rappert, and R. Menzel. The glomerular code for odor representation is species specific in the honeybee Apis mellifera. *Nat Neurosci*, 2(5):473–478, May 1999.

[GSSG05] F. Guerrieri, M. Schubert, J.C. Sandoz, and M. Giurfa. Perceptual and neural olfactory similarity in honeybees. *PLoS Biol*, 3(4):e60, Apr 2005.

[GV04] C. G. Galizia and R.S. Vetter. *Chapter 13 in "Advances in Insect Sensory Neuroscience" (ed. T.A. Christensen)*, pages 349–392. CRC Press, Boca Raton, 2004.

[HO00] A. Hyvärinen and E. Oja. Independent component analysis: algorithms and applications. *Neural Networks*, 13(4-5):411–430, 2000.

[LD06] X. Liu and R.L. Davis. Insect olfactory memory in time and space. *Curr Opin Neurobiol*, 16(6):679–685, Dec 2006.

[MB99] B.T. Messmer and H. Bunke. A decision tree approach to graph and subgraph isomorphism detection. *Pattern Recognition*, 32(12):1979–1998, Dec 1999.

[MHR07] J.L. Marchini, C. Heaton, and B.D. Ripley. *fastICA: FastICA Algorithms to perform ICA and Projection Pursuit, version 1.1-8*, Oct 2007.

[SG02] S. Sachse and C.G. Galizia. Role of inhibition for temporal and spatial odor representation in olfactory output neurons: a calcium imaging study. *J Neurophysiol*, 87(2):1106–1117, Feb 2002.

[VSJ+00] R. Vigário, J. Särelä, V. Jousmäki, M. Hämäläinen, and E. Oja. Independent component approach to the analysis of EEG and MEG recordings. *IEEE Trans Biomed Eng*, 47(5):589–593, May 2000.

Statistical detection of co-operative transcription factors with similarity adjustment

Utz J. Pape[1,2] and Holger Klein[1] and Martin Vingron[1]

[1] Dept. of Computational Biology, Max Planck Institute for Molecular Genetics, Ihnestr. 73, 14195 Berlin, Germany, (e-mail: utz.pape@molgen.mpg.de)
[2] Dept. of Mathematics and Computer Science, Free University of Berlin, Takustr. 9, 14195 Berlin, Germany.

Abstract: Statistical assessment of cis-regulatory modules (CRMs) is a crucial task in computational biology. Usually, one concludes from exceptional co-occurrences of DNA motifs that the corresponding transcription factors are co-operative. However, similar DNA motifs tend to co-occur in random sequences due to high probability of overlapping occurrences. Therefore, it is important to consider similarity of DNA motifs in the statistical assessment. Based on previous work, we propose to adjust the window size for co-occurrence detection. Using the derived approximation, one obtains different window sizes for different sets of DNA motifs depending on their similarities. This ensures that the probability of co-occurrences in random sequences are equal. Applying the approach to selected similar and dissimilar DNA motifs from human transcription factors shows the necessity of adjustment and confirms the accuracy of the approximation.

Our previously published statistics can only deal with non-overlapping windows. Therefore, we extend the approach and derive Chen-Stein error bounds for the approximation. Comparing the error bounds for similar and dissimilar DNA motifs shows that the approximation for similar DNA motifs yields large bounds. Hence, one has to be careful using overlapping windows. Based on the error bounds, one can pre-compute the approximation errors and select an appropriate overlap-scheme before running the analysis. Software and source code are available at http://mosta.molgen.mpg.de.

1 Introduction

An important goal in computational biology is to decipher the transcriptional regulation of genes. Interaction of nearby transcription factors (TFs) initiate or inhibit transcription of a gene [Fic96, YBD98]. They bind mainly upstream of genes to DNA by recognizing TF-specific sequences which can be summarized to a DNA motif. The set of DNA motif occurrences upstream of a gene is called a *cis* regulatory module (CRM, [BNP$^+$02]). A CRM is a sequence region with dense clusters of DNA motif occurrences as demonstrated experimentally [CMW$^+$03, HGL$^+$04] and computationally [Wag99, LMNP03]. TFs, which combinatorially regulate genes, are called co-operative. Such TFs are assumed to have exceptionally many DNA motif occurrences approximate to each other. Thus, a significant number of co-occurrences of the corresponding DNA motifs can be used to assess the strength of co-operativity.

CRMs can be detected using *ab initio* discovery of new (e.g. [ZW04, GL05]) or based on known DNA motifs. We assume that the DNA motifs are known. Many approaches have been proposed integrating data of different kind for improving CRM prediction [PSC01, YLZQ06]. Since the main characteristic of CRMs is their high local density of DNA motif occurrences, one essential data source is always the DNA sequence annotated with DNA motif occurrences. Here, we focus on DNA motifs represented by position frequency matrices (PFMs) [Sto00]. Other approaches compute the co-operative binding energy of multiple sites of TFs [GS01, FFY$^+$04] using thermo-dynamical models.

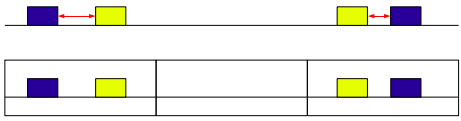


Figure 1: Two different approaches to detect CRMs: Upper panel illustrates approaches which are based on short distances between DNA motif occurrences. Lower panel visualizes detection of CRM considering occurrences in windows.

Based on the PFM representation, [Guh06] classifies the approaches to find CRMs into hidden Markov models [CRB97, FHW01] and occurrence-based approaches. We further divide the occurrence-based approaches into two categories, (i) relying on small distances between DNA motif occurrences [WF98, Wag99] and (ii) based on co-occurrences of DNA motifs in a small window [BNP$^+$02, HL02, FSHW02, KV07]. The method to compute statistical significance is a difficult problem [Kri04] and can be solved by (i) assuming position independence of occurrences [WF98, Wag99, FSHW02] or (ii) employing randomizations [HL02, BHVvR07] or (iii) exact calculation [BCR$^+$07].

The position independence of binding site occurrences is strongly violated for (self-)similar PFMs [Wag99, PRSV08]. The significance calculation based on randomization also encouters problems for similar PFMs, hence, they are usually removed from the analysis [HL02]. In addition, incorporating the complementary strand, introduces further dependencies and worsen the results. The exact calculation [BCR$^+$07] based on a Aho-Corasick automaton has high computational complexity such that solutions for longer PFMs are hard to obtain. Furthermore, the approach does not use the complementary strand.

In [PV08], we propose a fast and accurate approximation for the significance calculation of CRMs circumventing the position independence assumption, incorporating similarity between PFMs, and including the complementary strand. There, we define a CRM to be a sequence region, which we called window, of defined length where all DNA motifs of a given set have at least one occurrence. This is called the co-occurrence event. To get statistically significant CRMs, the length of the window has to be small such that the co-occurrence event is unlikely to happen by chance. We compute the probability of a CRM which is the probability of the co-occurrence event in a random sequence given a window length. Considering the overlap probabilities between the occurrences of the TF binding sites, we capture the (self-)similarities of the PFMs and most of the dependencies introduced by the complementary strand.

In this article, we extend the approach such that one can compute the length of the window for a specific set of DNA motifs by defining the probability of the co-occurrence event as parameter. We focus on pairs of DNA motifs. Intuitively, the results show that for similar PFMs the length of the window is smaller than for dissimilar PFMs given the same probability. Due to this computation, one can adjust the window size based on the similarity of the PFMs. Hence, by using different window sizes for sets of PFMs sharing different amounts of similarity between their PFMs, one can obtain equal co-occurrence probabilities for all sets. Therefore, follow-up analyses do not have to consider the similarity between PFMs anymore. Otherwise, similar PFMs would yield more co-occurrence events than dissimilar PFMs just due to their similarity. This would generally bias statistics based on the number of co-occurrence events. Hence, window size adjustment by considering the similarity of PFMs is necessary.

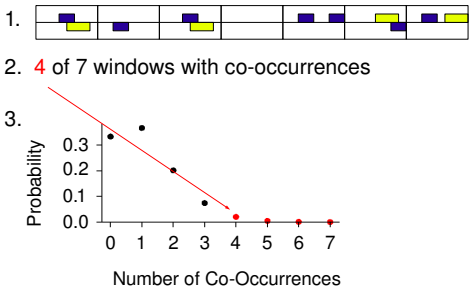


Figure 2: Proposed algorithm to compute co-operativity of a pair of TFs: First, divide sequence into windows. Second, count windows containing at least one hit of each TF. Compute corresponding count distribution under random sequence model to obtain p-value for co-operativity.

Furthermore, one is interested in whether specific TFs are generally involved in the same CRMs. We call this co-operativity of TFs. In [PV08], we also show how to compute the significance of co-operativity. The sequence is divided into equal-sized non-overlapping windows covering the whole sequence. We compute a p-value for the number of observed CRMs (windows with the co-occurrence event) since we can derive the count distribution of CRMs. In case of non-overlapping windows the count distribution is exact besides the approximations in the calculation of the co-occurrence event. The accuracy of the approximation is shown by comparison with a simulation study [PV08]. In contrast, overlapping windows introduce further dependencies. Therefore, we show in this article how to compute error bounds using the Chen-Stein method. Applying these error bounds to selected sets of PFMs show that similar PFMs retrieve high approximation errors due to stronger dependencies between overlapping windows.

In the next section, we derive formulas for the window length and explicitly state the Chen-Stein error bounds. Furthermore, we describe the data set of human TFs and how the PFMs are selected. The Section Results applies the formulas for window length and the Chen-Stein error bounds to a selected pairs of TFs.

2 Methods

We assume that each TF is given by a PFM. For each position j of a sequence, we have an indicator random variable $Y_j(A)$ which is 1 if the summed score at this position reaches the threshold. We denote the random variables for the complementary strand by a prime, e.g. $Y'_j(A)$. The threshold can be controlled by the type I error $\alpha_A := P(Y_j(A) = 1) = P(Y'_j(A) = 1)$ in a random sequence. The model for the random sequence is assumed to be an i.i.d. sequence defined by the GC content. We assume this simple background model, since we require the distribution of hits on both strands to be equal.

As stated before, a CRM is a window of given length w with at least one hit for TF A and one hit of TF B. We split up the calculation of this co-occurrence event into three parts: Let $N_w(A) = \sum_{j=1}^{w}(Y_j(A) + Y'_j(A))$ denote the random variable for the number of hits of TF A in a random sequence of length w where we allow hits overlapping the boundary of the window. Now, we can state the probability $p(w)$ of a CRM in a given window of length w by $p(w) := P(N_w(A) > 0, N_w(B) > 0)$. Calculation using the inclusion-exclusion formula and transformations as described in [PV08] yields for the probability of the co-occurrence event $p(w) \approx 1 - e^{-r_A \cdot w} - e^{-r_B \cdot w} + e^{-r_{AB} \cdot w}$ where r_A resp. r_B correspond to rates for the occurrence of TF A resp. B and r_{AB} contains the joint rate of A and B considering overlaps.

2.1 Calculate Window Size

In practice, the probability for the co-occurrence event is given as parameter and the window size has to be computed. In this case, we have to find the roots of

$$1 - \exp(-r_A \cdot w) - \exp(-r_B \cdot w) + \exp(-r_{AB} \cdot w) - p. \tag{1}$$

Using the Newton approach, we obtain following recursion starting from a chosen initial value w_0:

$$w_{i+1} = w_i - \frac{1 - \exp(-r_A \cdot w_i) - \exp(-r_B \cdot w_i) + \exp(-r_{AB} \cdot w_i) - p}{r_A \exp(-r_A \cdot w_i) + r_B \exp(-r_B \cdot w_i) - r_{AB} \exp(-r_{AB} \cdot w_i)}. \tag{2}$$

In case one requires a closed formula, one can also apply a Taylor expansion to the formula for the co-occurrence probability. E.g., the formula for a 2nd order expansion which already gives accurate results for small p is given with $a = r_{AB} - r_A - r_B$ and $b = r_{AB}^2 - r_A^2 - r_B^2$ by

$$w(p) = \frac{a}{b} + \sqrt{\left(\frac{a}{b}\right)^2 + \frac{2p}{b}}. \tag{3}$$

2.2 p-value for Co-operativity

Previously, we showed how to compute the co-occurrence probability $p(w)$ in a given window. To compute co-operativity, we suggest to decompose the sequence into non-overlapping windows of equal size and count the number x of CRMs (windows with the co-occurrence event). We define for each window i a Bernoulli random variable W_i which is 1 if the corresponding window contains a co-occurrence event and otherwise 0. Denoting the number of windows by $m = n/w$ with sequence length equal to n, we define $W := \sum_{i=1}^{m} W_i$. The number W of windows with co-occurrence events is distributed as Poisson $\mathcal{P}(\vartheta)$ with $\vartheta = p(w) \cdot m$ if $p(w) \to 0$ and $m \to \infty$.

2.3 Bounds for Overlapping Windows

Considering overlapping windows necessiates the step size s as parameter. The number m of windows becomes $m = n/s - w + 1$. We assume that n, s, w are chosen such that m, n, s, w are positive integers and $s < w < \frac{1}{2}n$. Obviously, overlapping windows are dependent on each other. In this case, we can still use a Binomial or Poisson distribution but the dependencies lead to an error in the approximation. Using the Chen-Stein method [Che75], the error can be quantified. The quantification is done in terms of the total variation distance. Let U and V be any two random processes with values in the same space E, then the total variation distance between their distributions (denoted by $\mathcal{L}(\cdot)$) is

$$d_{\mathrm{TV}}\left(\mathcal{L}(U), \mathcal{L}(V)\right) = \sup_{D \subset E} |P(U \in D) - P(V \in D)| \tag{4}$$

where D is assumed to be measurable. Here, we focus on the Poisson Approximation since it obtains slightly better error bounds. Thus, we calculate the bound for $d_{\mathrm{TV}}(\mathcal{L}(W), \mathcal{P}(\vartheta))$. Let denote $I := \{i : 0 < i \leq m\}$ the index set of the Bernoulli variables. The main idea is to define for each Bernoulli variable W_i a neighborhood set $B_i \subseteq I$ of random variables which have strong dependencies with W_i. We also require $i \in B_i$. In our case, there are only local dependencies since only overlapping windows are dependent on each other. Therefore, we capture all dependencies in the sets B_i which means that for each window i the set B_i contains the index i and the indices of overlapping windows to the left and to the right. Hence, we obtain the bound derived from Theorem 1 in [AGG90] using an improved bound [BHJ92] $d_{\mathrm{TV}}(\mathcal{L}(W), \mathcal{P}(\vartheta)) \leq \vartheta^{-1}(1 - e^{-\vartheta})(b_1 + b_2)$ with

$$b_1 := \sum_{i \in I} \sum_{j \in B_i} E[W_i] \cdot E[W_j], \qquad b_2 := \sum_{i \in I} \sum_{j \in B_i, j \neq i} E[W_i \cdot W_j]. \tag{5}$$

The bound b_1 is straight forward to compute as it only contains the first moments. We have to consider the fact that the B_is for the first and last few windows contain less dependent variables than windows in the middle of the sequence. Let $r = w/s$, then for example, the first window has $r - 1$ overlapping windows, thus, $|B_1| = r$ since we also include

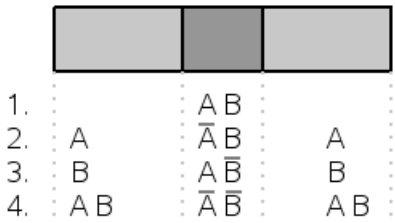


Figure 3: The four disjoint events for two windows where the dark grey area indicates the overlap. Regions containing an A or B must necessarily contain at least one hit of the corresponding TF while $\bar{A}$ and $\bar{B}$ label regions where the respective TF must not occur. In blank regions, any TF and combinations of TFs might be present.

index 1 in the set. The second window additionally overlaps with the first window, thus, $|B_2| = |B_1| + 1$. The set size is incremented by 1 until the $r+1$th window as this window has equal number of overlaps to the left and to the right. At the end of the sequence, the set size is decremented in the same way. Hence, we obtain $b_1 = p(w)^2\,(r(1-r+2m)-m)$.

The second bound b_2 is more complicated to calculate because it contains the second moments. Since we consider Bernoulli variables, the second moment is the probability that both variables are equal to one: $E[W_i W_{i+k}] = P(W_i = 1, W_{i+k} = 1)$. Considering only two TFs A and B, we can write this probability in terms of the count random variables by decomposing it into four disjoint events as illustrated in Fig. 3.

Denoting the size of each non-overlapping part by $d = k \cdot s$ while the overlapping part has a length of $v = w - d$, we obtain for the second moment:

$$\begin{aligned} E[W_i W_{i+k}] &= p(v) + \left(1 - e^{-dr_A}\right)^2 \left[1 - e^{-vr_B} - p(v)\right] && (6) \\ &\quad + \left(1 - e^{-dr_B}\right)^2 \left[1 - e^{-vr_A} - p(v)\right] + p(d)^2 e^{-vr_{AB}}. && (7) \end{aligned}$$

To compute the bound, we observe that $E[W_i W_{i+k}]$ is independent of i since all W_i are identically distributed and have the same pairwise dependencies. Therefore, we clarify notation by defining $\zeta_k := E[W_i W_{i+k}]$. For the same reason, we also obtain $\zeta_k = E[W_i W_{i-k}]$. Using the further definition of $\zeta = \sum_{k=1}^{r-1} \zeta_k$, we yield for bound b_2 applying the same logic as above:

$$b_2 = 2 \cdot \sum_{i=1}^{r} \left[\zeta + \sum_{k=1}^{i-1} \zeta_k\right] + 2(m-2r)\zeta = 2\left(m\zeta - r\zeta + \sum_{i=1}^{r}\sum_{k=1}^{i-1} \zeta_k\right). \quad (8)$$

Here, we assume that the empty sum ($\sum_{k=1}^{i-1} \zeta_k$ for $i = 1$) is equal to 0.

2.4 Data

The PFM set used here is the *vertebrate_non_redundant_minFP* set from the TRANSFAC database (v. 11.3) [MFG+03]. Since despite the name the set contains more than one PFM per transcription factor (214 in total), we only select the first PFM per TF and obtain a set of 142 PFMs. Hence, we are left with a set of one PFM per TF. However, the remaining similarities between PFMs in this set are not negligible. To show this, we measure the similarity between all pairs of PFMs by the limiting covariance [PRV08]. Then, we select the pair of PFMs with highest similarity (0.0002): `S8` (`V$S8_01`) and `CHX10` (`V$CHX10_01`). We use this pair for our analysis. To assess the influence of similarity, we also select a very dissimilar pair of PFMs. Given `S8`, the most dissimilar PFM is `HIC` (`V$HIC1_02`) with a similarity of -0.000004. Hence, we define a pair of similar PFMs `S8` and `CHX10` and a pair of dissimilar PFMs `S8` and `HIC`.

All analyses regarding PFMs are performed based on a balanced type-I error (α) in a sequence of length 500 controlled at a level of 1% (see [PGH+06] for details). In a step called regularization, we add pseudo-counts to the position specific distributions of the PFM according to the information content of the position [Rah03]. Simulated sequences are generated i.i.d. with 50% GC content.

3 Results and Discussion

First, we apply the formulas for the window size given a co-occurrence probability of $p = 0.01$ to both pairs of PFMs. The pair of similar PFMs `S8` and `CHX10` yields a window size of 54bp for both Newton iteration and Taylor expansion. Computing the co-occurrence probability for the window size 54bp yields exactly 0.01. Hence, both approximations are very accurate. The dissimilar pair `S8` and `HIC` yields for the same given co-occurrence probability a window size of 297bp using Newton iteration and 281bp using Taylor expansion. The corresponding co-occurrence probabilities are 0.01 and 0.009. Hence, the Newton iteration is slightly more accurate than the Taylor expansion. In comparison to the similar pair, one yields a 5-fold larger window size. Since similar PFMs tend to have overlapping hits, their probability of co-occurrence which includes overlapping hits is high. Therefore, an occurrence of one PFM increases the probability of an occurrence of the other PFM. In contrast, dissimilar PFMs cannot overlap. Thus, presence of one PFM decreases the probability of an (overlapping) occurrence of the other PFM. Due to the big difference in the window sizes, it is very important to consider the similarity between PFMs. The presented approach shows one can simply adjust the window size. Hence, one would use a window size of 54bp for the similar pair and of 297bp for the dissimilar pair. Then, both pairs have equal co-occurrence probabilities.

We verify this prediction by a simulation study. After annotating 100 random sequences each of length 1,000,000 by the corresponding PFMs, we count the number of co-occurrence events given above window sizes. The histograms for both pairs are shown in Fig. 4. The left panel contains the histogram for the similar pair. The distribution has a mean of 0.007

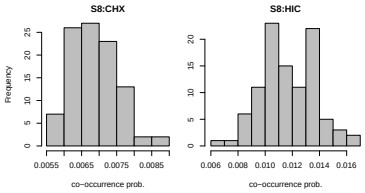


Figure 4: Histograms of empirical co-occurrence probabilities for (left panel) the similar pair S8 and CHX10 with window size 54bp and for (right panel) the dissimilar pair S8 and HIC with window size 297bp.

and a standard deviation of 0.0006. Hence, the approximated co-occurrence probability of 0.01 is slightly biased to lower probabilies. The reason is that the approximation of the co-occurrence probability only considers first-order dependencies between occurrences. This means overlaps between more than two occurrences are ignored. The right panel of Fig. 4 shows the histogram for the dissimilar pair. The mean is 0.012 with standard deviation 0.002. Thus, the approximated probability is still within one standard deviation of the mean. Since the corresponding PFMs do not overlap, the first-order approximation yields more accurate results. In contrast, applying the same window size (e.g. 297bp) to both pairs would yield a co-occurrence probability of around 0.04 (retrieved by simulation) for the similar pair. Hence, the difference between co-occurrence probabilities decreases from almost $3-4$-fold to quite comparable co-occurrence probabilities by adjusting the window size.

Based on the selected window sizes, one can compute Chen-Stein error bounds for the co-operativity p-value approximation. Using windows which overlap by 10% yields an error bound of 0.04 for the similar pair S8 and CHX10 on a sequence of length 1000bp. Hence, it will be difficult to obtain significant results since one cannot obtain p-values less than 0.04. In general, similar PFMs have a high approximation error for overlapping windows since overlapping occurrences induce high dependencies between two windows. In contrast, the dissimilar pair S8 and HIC has an error bound of 0.002. The bound is much smaller for two reasons: First, the window is much larger, thus, less windows are used for the sequence. Second, overlapping windows are less dependent due to small probabilities for the overlap of two occurrences. Hence, in case of dissimilar PFMs one can use overlapping windows and still obtain significant co-operativity.

In conclusion, we can state that detection of significant co-occurrences and co-operativity based on PFM occurrences is a difficult problem due to strong dependencies induced by similarity between PFMs. We show a reasonable approximation to adjust the window size such that co-occurrence and co-operativity probabilities are comparable between similar and dissimilar PFMs. Therefore, statistical follow-up analyses can ignore the similarity issue. In addition, we propose a new approximation for co-operativity using overlapping windows. Using the Chen-Stein technique, we can bound the approximation error. Results show that similar PFMs imply strong dependencies between overlapping windows. This leads to high approximation errors. In contrast, dissimilar PFMs yield low approxi-

mation errors. Based on our error bounds, one can pre-compute the approximation errors and select an appropriate overlap-scheme before running the analysis. In general, the approach can be extended to deal with sets of TFs. Furthermore, a more general sequence background model would be eligible.

References

[AGG90] Arratia, Goldstein and Gordon. Poisson Approximation and the Chen-Stein method. *Statistical Science*, 5:403–434, 1990.

[BCR+07] Boeva, Clément, Régnier, Roytberg and Makeev. Exact p-value calculation for heterotypic clusters of regulatory motifs and its application in computational annotation of cis-regulatory modules. *Algorithms Mol Biol*, 2:13, 2007.

[BHJ92] Barbour, Holst and Janson. *Poisson Approximation*. Oxford University Press, 1992.

[BHVvR07] Bleser, Hooghe, Vlieghe and van Roy. A distance difference matrix approach to identifying transcription factors that regulate differential gene expression. *Genome Biol*, 8(5):R83, 2007.

[BNP+02] Berman *et al.* Exploiting transcription factor binding site clustering to identify cis-regulatory modules involved in pattern formation in the Drosophila genome. *Proc Natl Acad Sci U S A*, 99(2):757–62, Jan 2002.

[Che75] Chen. Poisson approximation for dependent trials. *Ann. Probab.*, 3:534–545, 1975.

[CMW+03] Clyde, Corado, Wu, Pare, Papatsenko and Small. A self-organizing system of repressor gradients establishes segmental complexity in Drosophila. *Nature*, 426(6968):849–853, 2003.

[CRB97] Crowley, Roeder and Bina. A statistical model for locating regulatory regions in genomic DNA. *J Mol Biol*, 268(1):8–14, Apr 1997.

[FFY+04] Frith, Fu, Yu, Chen, Hansen and Weng. Detection of functional DNA motifs via statistical over-representation. *Nucl. Acids Res.*, 32(4):1372–1381, 2004.

[FHW01] Frith, Hansen and Weng. Detection of cis -element clusters in higher eukaryotic DNA. *Bioinformatics*, 17(10):878–889, 2001.

[FSHW02] Frith, Spouge, Hansen and Weng. Statistical significance of clusters of motifs represented by position specific scoring matrices in nucleotide sequences. *Nucleic Acids Res.*, 30(14):3214–3224, 2002.

[Fic96] Fickett. Coordinate positioning of MEF2 and myogenin binding sites. *Gene*, 172(1):GC19–GC32, 1996.

[GL05] Gupta and Liu. De novo cis-regulatory module elicitation for eukaryotic genomes. *Proceedings of the National Academy of Sciences*, 102(20):7079–7084, 2005.

[GS01] GuhaThakurta and Stormo. Identifying target sites for cooperatively binding factors. *Bioinformatics*, 17(7):608–621, 2001.

[Guh06] GuhaThakurta. Computational identification of transcriptional regulatory elements in DNA sequence. *Nucleic Acids Res.*, 34(12):3585–3598, 2006.

[HGL+04] Harbison *et al.* Transcriptional regulatory code of a eukaryotic genome. *Nature*, 431:99–104, 2004.

[HL02] Hannenhalli and Levy. Predicting transcription factor synergism. *Nucl. Acids Res.*, 30(19):4278–4284, 2002.

[Kri04] Krivan. Searching for transcription factor binding site clusters: How true are true positives? *J Bioinform Comput Biol*, 2(2):413–6, 2004.

[KV07] Klein and Vingron. Using Transcription Factor Binding Site Co-Occurrence to Predict Regulatory Regions. *Genome Informatics*, 18:109–118, 2007.

[LMNP03] Lifanov, Makeev, Nazina and Papatsenko. Uniform clusters in Drosophila. *Genome Res*, 13(4):579–588, 2003.

[MFG+03] Matys *et al.* TRANSFAC(R): transcriptional regulation, from patterns to profiles. *Nucleic Acids Res.*, 31(1):374–378, 2003.

[PGH+06] Pape, Grossmann, Hammer, Sperling and Vingron. A new statistical model to select target sequences bound by transcription factors. *Genome Informatics*, 17(1):134–140, 2006.

[PRSV08] Pape, Rahmann, Sun and Vingron. Compound Poisson approximation of number of occurrences of a Position Frequency Matrix (PFM) on both strands. Accepted by J. Comput. Biol., 2008.

[PRV08] Pape, Rahmann and Vingron. Natural Similarity Measures between Position Frequency Matrices with an Application to Clustering. *Bioinformatics*, 24(3):350–357, 2008.

[PV08] Pape and Vingron. Statistics for Co-Occurrence of DNA Motifs. Accepted by IWAP 2008.

[PSC01] Pilpel, Sudarsanam and Church. Identifying regulatory networks by combinatorial analysis of promoter elements. *Nat Genet*, 29(2):153–9, Oct 2001.

[Rah03] Rahmann. Dynamic programming algorithms for two statistical problems in computational biology. In *Proceedings of the 3rd Workshop of Algorithms in Bioinformatics (WABI)*, pages 151–164, Heidelberg, 2003. Springer Verlag.

[Sto00] Stormo. DNA binding sites: representation and discovery. *Bioinformatics*, 16:16–23, 2000.

[Wag99] Wagner. Genes regulated cooperatively by one or more transcription factors and their identification in whole eukaryotic genomes. *Bioinformatics*, 15(10):776–784, 1999.

[WF98] Wasserman and Fickett. Identification of regulatory regions which confer muscle-specific gene expression. *J Mol Biol*, 278(1):167–81, Apr 1998.

[YBD98] Yuh, Bolouri and Davidson. Genomic Cis-Regulatory Logic: Experimental and Computational Analysis of a Sea Urchin Gene. *Science*, 279(5358):1896–1902, 1998.

[YLZQ06] Yu, Lin, Zack and Qian. Computational analysis of tissue-specific combinatorial gene regulation: predicting interaction between transcription factors in human tissues. *Nucleic Acids Res*, 34(17):4925–36, 2006.

[ZW04] Zhou and Wong. CisModule: De novo discovery of cis-regulatory modules by hierarchical mixture modeling. *Proceedings of the National Academy of Sciences*, 101(33):12114–12119, 2004.

Intuitive Modeling of Dynamic Systems with Petri Nets and Fuzzy Logic

Lukas Windhager*, Ralf Zimmer

Institut für Informatik, Ludwig-Maximilians-Universität München,
Amalienstrasse 17, 80333 München, Germany
Lukas.Windhager@bio.ifi.lmu.de

Abstract: Current approaches in modeling dynamic biological systems often lack comprehensibility, especially for users without mathematical background. We propose a new approach to overcome such limitations by combining the graphical representation provided by the use of Petri nets with the modeling of dynamics by powerful yet intuitive fuzzy logic based systems. The mathematical functions and formulations typically used to describe or quantify dynamic changes of systems are replaced by if-then rules, which are both easy to read and formulate. Precise values of kinetic constants or concentrations are substituted by more natural fuzzy representations of entities. We will show that our new approach allows a semi-quantitative modeling of biological systems like signal transduction pathways or metabolic processes while not being limited to those cases.

1 Introduction

To gain insight into a biological system, computational models are built based on current knowledge and hypotheses. The behavior of these models is investigated under different constraints and compared to experimental observations, known facts or other data to verify or falsify the current model. Many of the currently availabe approaches for modeling biological systems are based on ordinary differential equations (ODEs), Bayesian or boolean networks, different types of Petri nets (PNs), combinations thereof as well as other, less common techniques like signal-flow diagrams and system dynamics models. See [GFG+06, MPLD04, OSV+05] for some reviews concerning computational modeling. ODE based modeling of dynamic changes in systems is probably the most widespread method. Entities of the modeled system (proteins, metabolites, etc.) are described by state variables which typically correspond to the concentrations or amounts of those entities at a given time. The change of these variables over time is hereby described by a set of differential equations which involve not only the state variables but also several kinetic constants. ODE based modeling was applied for example for the analysis of yeast cell cycle [CCNG+00], E. coli carbohydrate uptake [KBG07], dynamics of yeast pheromone pathway [KK04] or the modeling of the EGF receptor induced MAP kinase

*Corresponding author.

cascade [SEJGM02]. Some widely used graph-based approaches to systems biology modeling are based on Petri nets (see [Cha07] for a recent review and [Mur89] for an extensive introduction to Petri net theory). Generally, Petri nets are graphical representations of (biological) entities like proteins, genes and metabolites as well as (biological) processes like enzymatic reactions, transport, degradation, etc.. There are several different types of Petri net modeling techniques in use, ranging from the basic type (see [RLM96]) to more involved and extended types like hybrid functional Petri nets (HFPN; [MTA$^+$03]). HFPNs extend the definition of basic Petri nets by introducing additional arc types (inhibitory and test arcs), a more sophisticated definition of tokens and the use of arbitrary functions instead of fixed arc-weights. These functions are typically similar to ODEs, incorporating concentrations of neighboring places and pre-defined kinetic constants. See [GKV01] for an executable Petri net model of glycolysis and citric acid cycle, [LZLP06] for a colored Petri net model of the EGF receptor induced MAP kinase cascade or [LGN$^+$07] for a timed Petri net model of the apoptosis pathway.

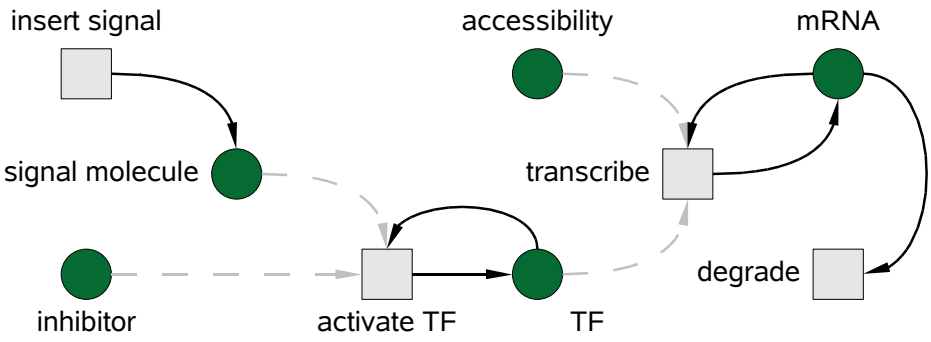


Figure 1: A Petri net representation of a small biological system. *Places* (filled circles) visualize arbitrary system entities or properties like proteins, metabolites, etc.. In our framework, their current states are described using weighted fuzzy sets replacing the commonly used tokens. *Transitions* (grey squares) visualize arbitrary (biological) processes like enzymatic reactions, transport, etc.. *Arcs* visualize dependencies of places and transitions (test arcs, dashed arrows) or define which and how places are affected whenever a transition fires (input and output arcs, solid arrows). In our framework, these effects are defined using fuzzy logic systems instead of the commonly used weights or other mathematical functions.

In this article, we introduce and motivate a new modeling approach (termed **PNFL**, Petri Nets with Fuzzy Logic) which provides a powerful and intuitive tool for investigating biological processes and systems. PNFL provides an environment where hypotheses in biological systems can be formulated, visualized and simulated in a quite intuitive and natural way and overcomes limitations of ODE-based modeling by:

1. Replacing mathematical formulations of dynamics by natural language based rule systems to facilitate comprehensibility.
2. Omitting use, definition and estimation of exact parameter values through a fuzzier, thus natural, definition of typically qualitative knowledge about entities and processes.

3. Allowing for incorporation of entities and their concentrations as well as other, arbitrary properties of entities or systems by a uniform framework based on fuzzy logic.
4. Using Petri nets as graphical frameworks for development and simulation of user-defined systems to provide a clear visualization and distinction of entities and processes.

The main innovation of our PNFL approach is the use of elements from fuzzy logic theory to describe biological systems: *Fuzzy sets* describe arbitrary entities or properties of a system; *Fuzzy logic systems* define the dynamics of biological processes and dependencies between entities. Petri nets are used as a scaffold for the fuzzy logic based definitions of biological entities and processes (figure 1).

2 Fuzzy Logic Based Modeling

The real world has an approximate and inexact nature and sets of objects in this world are usually characterized by inexact boundaries. For example, defining the "set of highly concentrated metabolites" as "the set of metabolites present at a level of more than $1 * 10^6$ molecules per mol" is unsatisfactory as this strict border is probably artificial. It is difficult to argue, that a metabolite present at $1.01 * 10^6$ molecules per mol is "highly concentrated" while it would not be "highly concentrated" at $0.99 * 10^6$ molecules per mol. In order to capture the inexact nature of our surrounding world, Lotfi A. Zadeh introduced the notion of fuzzy sets and extended the two-valued $\{0,1\}$ logic to the interval [0,1], allowing a gradual transition from falsehood to truth [Zad65, Zad96]. Fuzzy sets also allow the representation of imprecise, subjective knowledge and linguistic information. Elements are not seen as being either part of a set or not but instead they are defined as being similar to elements *described* by a set. The similarity is quantified by assigning a value between 0 (dissimilar) to 1 (equal). A fuzzy set, defined over a universe of discourse U, is characterized by its *membership function* $FS : U \rightarrow [0, 1]$. The membership function defines the similarity of an item to the fuzzy set. The universe of discourse U contains all elements that could possibly be part of the set, e.g. a set describing "high concentrations" may be defined over $[0, \infty]$ (all possible concentrations). For an extensive introduction to fuzzy logic see [Men95, Lee90a, Lee90b].

As different fuzzy sets may describe elements of the same (biological) concept, for example the concept "concentration of a protein P", we subsume fuzzy sets to *fuzzy concepts*, which correspond to the real-world concepts. Fuzzy concepts are defined as tuples $< FS_1, ..., FS_n >$, where all fuzzy sets FS_i are defined over the same universe of discourse. The fuzzy sets combined to a fuzzy concept usually have differently shaped membership functions as they describe different aspects of the underlying (biological) concept. An exemplary fuzzy concept *concentration* may include fuzzy sets *low*, *medium*, *high* and *saturated*, each describing a different "level" of concentration (figure 2).

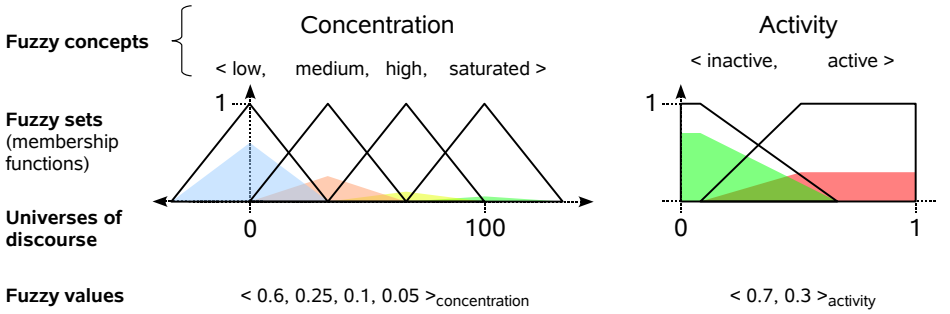


Figure 2: The current state of an entity with respect to a (biological) concept (e.g. its current concentration) can be described by the membership function values of fuzzy sets, which belong to an according fuzzy concept. A *fuzzy value* is a tuple $< w_1, ..., w_n >_{FC}$ specified with respect to a fuzzy concept $FC = < FS_1, ..., FS_n >$. The membership function values $w_i \in [0, 1]$ are called *weights* and describe the current state of an entity with respect to the fuzzy concept. Colored areas visualize the currently assigned weights.

2.1 How Fuzzy Values Represent Concentrations and Other Properties

Concentrations or amounts of proteins, RNA, metabolites, etc. are typically represented as positive real numbers. Such real numbers can in turn be represented as fuzzy values with respect to a fuzzy concept. To create suitable fuzzy concepts, several modeling decisions have to be made:

1. **Define the universe of discourse.** Basically, the whole set of real numbers could be used but it is also possible to define an explicit range, for example when the concentration of an entity is bounded by some equilibrium constraints.
2. **Define the number of fuzzy sets.** A higher number of fuzzy sets allows more detailed representations of states and the associated dynamics. On the other hand the size of rule tables in fuzzy logic systems increases, allowing a higher number of different outcomes.
3. **Define the shape and position of membership functions.** Arbitrary membership function shapes can be defined although symmetric triangular, trapezoidal or gaussian shapes should suffice for most applications. Position, shape and spread of fuzzy sets can be freely defined according to modeling requirements.

It is part of the modeling decision to utilize the same fuzzy concept for only one, some or all entities of a system. If the concentration of an entity is known it can be transferred (fuzzyfied) to a fuzzy value simply by computing the according membership function values for each fuzzy set. If it is not known but only some rough guesses are available, weights can be assigned directly to fuzzy sets. For example, if the concentration of an entity is only known to be “quite low” a suitable fuzzy value may look like $< 0.8, 0.2, 0.0, 0.0 >_{concentration}$.

2.2 How Fuzzy Logic Systems Replace Differential Equations

Dynamic processes within a system are induced and influenced by the current state of the system and its entities and in turn influence and change them. If the current states of entities are defined by fuzzy values, processes have to be modeled by functions that operate on weighted fuzzy sets. These functions can be defined using natural language terms and without use of mathematical formulas.

A fuzzy logic system (FLS) consist of a set of rules mapping (weighted) fuzzy sets of several places (premises) to a set of output fuzzy sets (conclusions), thereby defining new weights for them. Fuzzy logic systems are specified as sets of natural language based rules. Single rules are defined as IF-THEN sentences, where several fuzzy sets, connected by AND-operators, in the IF-clause (premises) are mapped to a single concluding fuzzy set (conclusion).
Fuzzy logic theory offers several set theoretic operations to evaluate a fuzzy logic system. We decided for the frequently used and very intuitive sum-product logic ([Men95]):

1. **Inference of the weight of single conclusions depending on their premises.** Weights of premises are multiplied to infer the weight of a conclusion (product-inference).
2. **Combination of those conclusions referring to the same property.** Weights of conclusions with identical fuzzy sets are summed (sum-composition).

Generally (and intuitively) it holds that the higher the confidence of the premises (the higher they are weighted), the more confident is the conclusion (the higher it is weighted) (figure 3).

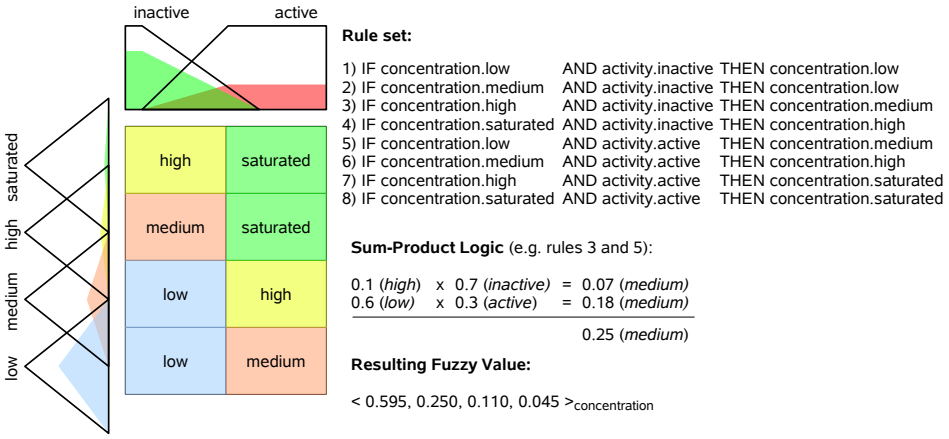


Figure 3: An example of a fuzzy logic system which uses the fuzzy sets and values described in figure 2 as premises to calculate weights of a fuzzy value of type *concentration* (conclusion). The rule set can be represented as a table (left). The premises (top and left, with membership functions and visualized weights) are mapped to the conclusions (center, without visualized membership functions or weights).

3 On the Use of Fuzzy Values and Fuzzy Logic Systems in Petri Nets

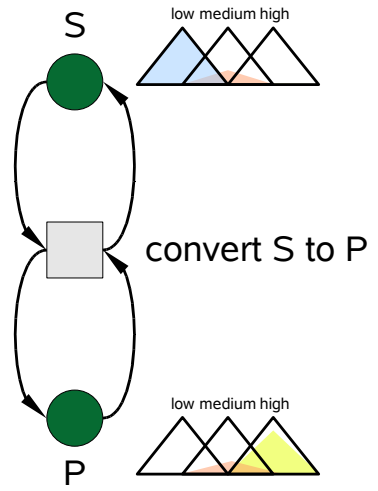
Fuzzy values are used to describe the current state of an entity with respect to a fuzzy concept. An arbitrary number of different fuzzy values can be used to describe each entity. All fuzzy values describing a single entity form a set of *fuzzy tokens* on the respective place in the Petri net model of the system. The set of fuzzy tokens represents those properties (concepts) an entity could *possibly* exhibit while the current weight assignment reflects the properties an entity *currently* exhibits. Fuzzy logic systems define the dynamics of a system. One or several of them serve as inscriptions of arcs. Whenever a transition fires, fuzzy tokens of adjacent places are consumed and a new set of fuzzy tokens is created by the FLS's of incident arcs. We distinguish three types of arcs which correspond to input, output and test arcs as defined for hybrid functional Petri Nets ([MTA$^+$03]). Input and output arcs consume and produce tokens whenever the incident transition fires while test arcs do not affect tokens. Test arcs symbolize a functional dependency of processes and entities, they allow the usage of fuzzy tokens of incident places as premises of fuzzy logic systems without consuming them.

4 Results and Conclusion

The adaption of fuzzy sets for representing states and properties and fuzzy logic based reasoning for describing processes can be used to model biological systems. Fuzzy sets capture the typically inexact, qualitative knowledge about biological entities and are well suited to represent limited knowledge, inexact measurements as well as error prone data. Due to the fact that they can stand for arbitrary properties, it is possible to uniformly represent all types of external and internal factors influencing a system. Fuzzy sets can be designed freely by a user according to his needs. Fuzzy logic systems allow the formulation of biological processes using simple yet powerful rule systems, which can be formulated using natural language. Therefore, hypotheses concerning the behavior of entities or influences between entities can be translated directly into executable systems (application 1, figures 4 and 5). The representation using Petri nets clearly visualizes entities, processes and dependencies within a biological system. A Petri net and fuzzy logic based system can easily be outlined in a pen-and-paper style by creating drafts of entities and their dependencies and describing the desired properties and effects of dependencies and influences in natural language.

The extension of fuzzy sets, fuzzy concepts and fuzzy values to represent arbitrary (non-quantifiable) properties or states of entities is straightforward. In fact, no changes of the definitions of these terms are necessary. Properties which are not per se quantifiable, like the current state of a cell in the cellcycle, may be described similar to concentrations using several fuzzy sets. Such fuzzy sets, for example belonging to the fuzzy concept *cellcycle state*, are then weighted to define the current state of an entity and represented as a fuzzy value. Although the described entity (the "cell cyle state") has no inherent reference to a real value, the universe of discourse of these fuzzy sets can still be defined as arbitrary range within the set of real numbers for the sake of uniformity. Modeling the state of a

Application 1: Minimal model of a Higgins-Sel'kov oscillator



Rule set defining concentration of *P*:

IF S.low		THEN P.low
IF !S.low	AND P.low	THEN P.medium
IF !S.low	AND !P.low	THEN P.high

Rule set defining concentration of *S*:

IF	P.high	THEN S.low
IF S.low	AND !P.high	THEN S.medium
IF !S.low	AND !P.high	THEN S.high

ODE model:

$$dS / dt = v_0 - k_1 S P^2$$
$$dP / dt = k_1 S P^2 - k_2 P$$

where $v_0 = 1$, $k_1 = 1$, $k_2 = 1.00001$,
$S(0) = 2$ and $P(0) = 1$.

Figure 4: A minimal model of an oscillator similar to the Higgins-Sel'kov Oscillator (ODE model taken from [KHK+05]). The underlying process and the ODE model (figure 5) can be described by few sentences: (1) S increases P; (2) P increases P strongly; (3) If P reaches a high level, S decreases strongly; (4) If S reaches a low level, P decreases strongly; and directly converted to a set of rules. The stated six rules suffice to create an oscillating behavior qualitatively similar to the ODE model. If a fuzzy set is negated, its current weight w is replaced by $(1 - w)$ during the execution of a FLS.

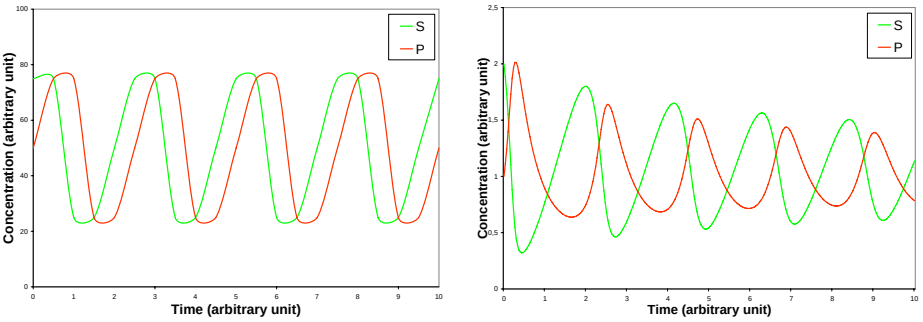


Figure 5: Time courses of the minimal fuzzy logic based model (left) and the ODE based model (right) of the Higgins-Sel'kov Oscillator described in figure 4. The PNFL model qualitatively reflects the oscillating behavior of S and P. A more involved PNFL model with extended fuzzy logic systems and two additional transitions modeling input of S and output of P is able to reproduce the dampening of oscillations as observed in the ODE model (data not shown).

Application 2: Hierarchical modeling of oscillating behavior

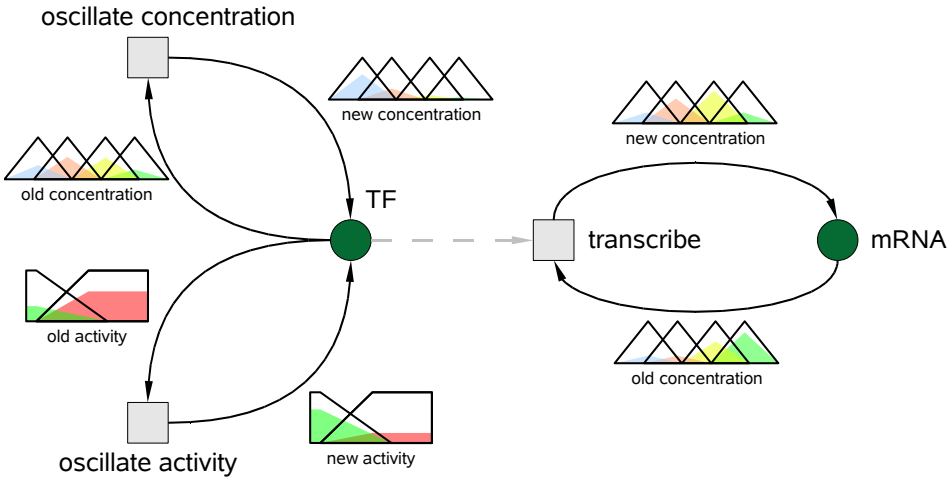


Figure 6: The concentration and activity of a transcription factor *TF* are conrolled by two transitions and exhibit an oscillating behavior. The underlying biological processes are not explicitly modeled but are described using appropriate rule systems to reduce the size of the model. Concentration and activity in turn influence the current concentration of mRNA molecules via the fuzzy logic system described in figure 3.

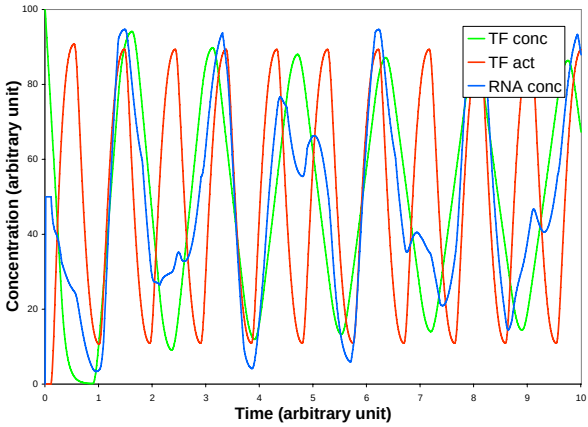


Figure 7: A time course of the dynamical behavior of the system described in figure 6. The oscillations of *TF*s concentration and activity differ in frequency and induce a quite irregular behavior of mRNA concentration.

system and its properties by the same framework as used for quantifiable entities is one of the main advantages of our fuzzy logic based approach. A uniform representation of quantifiable entities and other, more abstract properties is possible while dynamical changes of those parts of a system can be performed using the same technique, namely fuzzy logic systems. Their rule-based description allows modeling of complex behavior and is more powerful than a simple description of dependencies as *activating* or *inhibiting*, as it is common in boolean networks.
It is possible to model the behavior of entities by an explicit formulation of the underlying biological processes, for example an oscillation of a protein level by modeling a negative feedback loop delayed by transport via the core membrane. At the other hand one could force entities to behave in a particular way by *defining* their behavior with appropriate rules and without explicitly modeling real biological processes. This is for example very useful when a certain behavior of entities can be observered experimentally but not yet explained adequately by a model, while at the same time the modeling of the observed behavior is crucial as it affects other parts of the system. Additionally, replacing the extensive elaboration of biological processes by simpler systems mimicking their behavior also allows a hierarchical modeling (application 2, figures 6 and 7).

The described approach (PNFL) is currently improved and extended, including a GUI suited for model building, defining fuzzy sets, formulation of FLS rule sets and visualizing simulation runs and results. The implementation will also support concurrent simulations of biological systems in several cells. A prototype system was successfully applied during different developmental stages to several small test systems, like an in-silico network ([ZDGS01, ZGSD03]), typical network motifs (e.g. feed-forward loops, switches) and several oscillator models (Higgins-Sel'kov, minimal mitotic, coupled oscillators; [KHK$^+$05]). As a larger application a model of the EGF signal transduction pathway as defined in [LZLP06] was evaluated by replacing mass action kinetics by fuzzy logic systems.

References

[CCNG$^+$00] K. C. Chen, A. Csikasz-Nagy, B. Gyorffy, J. Val, B. Novak, and J. J. Tyson. Kinetic Analysis of a Molecular Model of the Budding Yeast Cell Cycle. *Mol. Biol. Cell*, 11(1):369–391, January 2000.

[Cha07] C. Chaouiya. Petri net modelling of biological networks. *Brief Bioinform*, 8(4):210–219, July 2007.

[GFG$^+$06] D. Gilbert, H. Fuß, X. Gu, R. Orton, S. Robinson, V. Vyshemirsky, M. J. Kurth, C. S. Downes, and W. Dubitzky. Computational methodologies for modelling, analysis and simulation of signalling networks. *Brief Bioinform*, 7(4):339–353, November 2006.

[GKV01] H. Genrich, R. Küffner, and K. Voss. Executable Petri net models for the analysis of metabolic pathways. *International Journal on Software Tools for Technology Transfer (STTT)*, 3(4):394–404, 2001.

[KBG07] A. Kremling, K. Bettenbrock, and E. D. Gilles. Analysis of global control of Escherichia coli carbohydrate uptake. *BMC systems biology*, 1(42), 2007.

[KHK+05] E. Klipp, R. Herwig, A. Kowald, C. Wierling, and H. Lehrach. *Systems Biology in Practice. Concepts, Implementation and Application.* WILEY-VCH, 2005.

[KK04] B. Kofahl and E. Klipp. Modelling the dynamics of the yeast pheromone pathway. *Yeast*, 21(10):831–850, July 2004.

[Lee90a] C. C. Lee. Fuzzy logic in control systems: fuzzy logic controller. I. *Systems, Man and Cybernetics, IEEE Transactions on*, 20(2):404–418, 1990.

[Lee90b] C. C. Lee. Fuzzy logic in control systems: fuzzy logic controller. II. *Systems, Man and Cybernetics, IEEE Transactions on*, 20(2):419–435, 1990.

[LGN+07] C. Li, Q. W. Ge, M. Nakata, H. Matsuno, and S. Miyano. Modelling and simulation of signal transductions in an apoptosis pathway by using timed Petri nets. *Journal of biosciences*, 32(1):113–127, January 2007.

[LZLP06] D. Y. Lee, R. Zimmer, S. Y. Lee, and S. Park. Colored Petri net modeling and simulation of signal transduction pathways. *Metabolic Engineering*, 8(2):112–122, March 2006.

[Men95] J. M. Mendel. Fuzzy logic systems for engineering: a tutorial. *Proceedings of the IEEE*, 83(3):345–377, 1995.

[MPLD04] J. Mandel, N. M. Palfreyman, J. A. Lopez, and W. Dubitzky. Representing bioinformatics causality. *Brief Bioinform*, 5(3):270–283, January 2004.

[MTA+03] H. Matsuno, Y. Tanaka, H. Aoshima, A. Doi, M. Matsui, and S. Miyano. Biopathways representation and simulation on hybrid functional Petri net. *In silico biology*, 3(3):389–404, 2003.

[Mur89] T. Murata. Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, 77(4):541–580, 1989.

[OSV+05] R. J. Orton, O. E. Sturm, V. Vyshemirsky, M. Calder, D. R. Gilbert, and W. Kolch. Computational modelling of the receptor-tyrosine-kinase-activated MAPK pathway. *Biochem J*, 392(Pt 2):249–261, December 2005.

[RLM96] V. N. Reddy, M. N. Liebman, and M. L. Mavrovouniotis. Qualitative analysis of biochemical reaction systems. *Computers in biology and medicine*, 26(1):9–24, January 1996.

[SEJGM02] B. Schoeberl, C. Eichler-Jonsson, E. D. Gilles, and G. Müller. Computational modeling of the dynamics of the MAP kinase cascade activated by surface and internalized EGF receptors. *Nat Biotechnol*, 20(4):370–375, April 2002.

[Zad65] L. A. Zadeh. Fuzzy sets. *Information and Control*, 8:338–353, 1965.

[Zad96] L. A. Zadeh. Fuzzy logic = Computing with words. *IEEE Transactions on Fuzzy Systems*, 4(2):103–111, 1996.

[ZDGS01] D. E. Zak, F. J. Doyle, G. E. Gonye, and J. S. Schwaber. Simulation studies for the identification of genetic networks from cDNA array and regulatory activity data. *Proc. 2nd Intl. Conf. Systems Biology*, pages 231–238, 2001.

[ZGSD03] D. E. Zak, G. E. Gonye, J. S. Schwaber, and F. J. Doyle. Importance of input perturbations and stochastic gene expression in the reverse engineering of genetic regulatory networks: insights from an identifiability analysis of an in silico network. *Genome Res*, 13(11):2396–2405, November 2003.

Utilizing promoter pair orientations for HMM-based analysis of ChIP-chip data

Michael Seifert[1], Jens Keilwagen[1], Marc Strickert[1], and Ivo Grosse[2]

[1]Leibniz Institute of Plant Genetics and Crop Plant Research, Gatersleben, Germany
[2]Martin Luther University, Institute of Computer Science, Halle, Germany

seifert@ipk-gatersleben.de

Abstract: Array-based analysis of chromatin immunoprecipitation data (ChIP-chip) is a powerful technique for identifying DNA target regions of individual transcription factors. Here, we present three approaches, a standard log-fold-change analysis (LFC), a basic method based on a Hidden Markov Model (HMM), and an extension of the HMM approach to an HMM with scaled transition matrices (SHMM) to incorporate different promoter pair orientations. We compare the prediction of ABI3 target genes for the three methods and evaluate these genes using Genevestigator expression profiles and transient assays. We find that the application of the SHMM leads to a superior identification of ABI3 target genes. The software and the ChIP-chip data set used in our case study can be downloaded from http://dig.ipk-gatersleben.de/SHMMs/ChIPchip/ChIPchip.html.

1 Introduction

In recent years, array-based analysis of chromatin immunoprecipitation data (ChIP-chip) has become a powerful technique to identify DNA target regions of individual transcription factors. ChIP-chip was firstly applied to yeast by [RRW$^+$00] and [IHS$^+$01] based on promoter arrays. Nowadays, with the availability of sequenced genomes, ChIP-chip is mostly based on tiling arrays [JLG$^+$08]. The analysis of ChIP-chip data is challenging because of the huge data sets containing thousands of hybridization signals. Most of the available methods focus on the analysis of ChIP-chip tiling array data. Examples include a moving average method by [KvdLDC04], a Hidden Markov Model (HMM) approach by [LML05], or TileMap by [JW05] including both approaches.

Regarding *A. thaliana*, ChIP-chip is still far from being used routinely. In the trilateral project ARABIDOSEED, ChIP-chip based on promoter arrays was established for the seed-specific transcription factor ABI3. ABI3 is one of the fundamental regulators of seed development that is involved in controlling chlorophyll degradation, storage product accumulation, and desiccation tolerance [VCC05].

Here, we describe and compare three methods for the detection of transcription factor target genes from ChIP-chip data. The first method, which we abbreviate by LFC, is a

standard log-fold change analysis in which the genes belonging to the promoters with the highest log-fold changes in the intersection of repeated experiments are considered to be putative target genes. The second method is based on a two-state (target promoter state and non-target promoter state) HMM. The principle architecture of the HMM follows the proposed two-state architecture by [LML05]. Our approach is extended in that way that all HMM parameters are directly learned from the ChIP-chip data. The HMM scores all promoters by the probability of being in the target promoter state, and we consider all genes belonging to promoters with the highest scores in the intersection of repeated experiments as putative target genes. The HMM allows statistical dependencies between ChIP-chip measurements of adjacent promoters along the chromosomes. The existence of such dependencies is clearly shown for ChIP-chip data of ABI3 in Fig. 1. We find that adjacent promoters in head-head orientation show significantly greater correlations than promoter pairs in head-tail, tail-head, or tail-tail orientation. The high correlations in ChIP-chip measurements of head-head promoter pairs can be explained by the array design: since proximal promoters but not complete intergenic regions are spotted. Thus, high positive correlations of measurements for head-head promoter pairs result from DNA segments of the intergenic region that bind to both promoter spots, or fragments of these segments where some of them bind to the one spot while the others bind to the other spot. The observation of correlations between ChIP-chip measurements of adjacent promoters motivates the extension of the HMM approach to an HMM with scaled transition matrices (SHMM). The general concept of SHMMs was developed by [Sei06] and applied to the analysis of tumor expression data by exploiting chromosomal distances of adjacent genes yielding to an improved detection of over-expressed and under-expressed genes. Here, we use this concept for discriminating head-head promoter pairs from other promoter pair orientations. The key assumption is that it is more likely for promoters in head-head orientation that both promoters are either target promoters or non-target promoters compared to other promoter orientations.

We use an ABI3 ChIP-chip data set for comparing the prediction of ABI3 target genes by the LFC, the HMM, and the SHMM method. We evaluate putative ABI3 target genes using (i) publicly available expression data from Genevestigator [ZHHHG04] and (ii) transient assays to test whether a putative target promoter is controlled by ABI3.

In general, good introductions to HMMs are given by [Rab89] or [DEKM98], extensions of standard HMMs to HMMs with transition matrices are described in [KSSW03], and some more details to SHMMs can be found in [Sei06]. A concept similar to SHMMs has been developed by [MD04] with an application to gene prediction.

2 Methods

2.1 Data acquisition and pre-processing

To determine target genes of the ABI3 transcription factor the ChIP-chip technique by [RRW$^+$00] and [IHS$^+$01] was applied to *A. thaliana* wildtype seeds. Isolated DNA fragments bound by ABI3 were amplified, radio-labeled, and hybridized to a macroarray con-

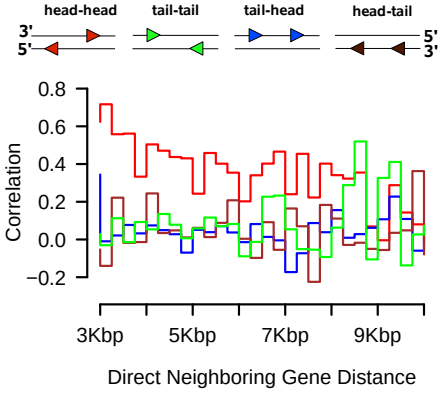


Figure 1: Pearson's correlations for the four promoter pair orientations based on log-ratios of ABI3 ChIP-chip experiments in steps of 250bp within the corresponding gene pair distance interval $[3, 10]$Kbp. A triangle represents a promoter and the orientation of its tip describes the reading direction of the gene belonging to this promoter.

taining 11,904 promoters of *A. thaliana*. The corresponding control sample was obtained from the input chromatin of the wildtype seeds by fragmentation, amplification, labeling, and hybridization to another promoter macroarray. In total, each of these two experiments was repeated five times. In a first normalization step, we center the median of each experiment to zero and perform a quantile normalization [BIAS03] separately for the ABI3 ChIP-chip experiments and the control experiments. In a second step, we combine each normalized ABI3 ChIP-chip experiment with its corresponding control experiment by calculating the log-ratio $o_t = I_{ABI3}(t) - I_{Control}(t)$ for all promoters t, where $I_{ABI3}(t)$ is the $\log_2$-signal intensity of promoter t in the ABI3 ChIP-chip experiment, and $I_{Control}(t)$ is the $\log_2$-signal intensity of promoter t in the control experiment. We map all of the log-ratios of such an experiment combination to their corresponding positions in the genome of *A. thaliana* based on the TAIR7 genome annotation, resulting in one ChIP-chip profile $o = o_1, \ldots, o_T$ per chromosome. As *A. thaliana* has five chromosomes 25 ChIP-chip profiles were obtained from the five replicates.

2.2 Standard Log-Fold-Change analysis (LFC) for target gene detection

The log-ratio of a promoter characterizes the potential of the gene belonging to this promoter to be a target gene of the ABI3 transcription factor. Thus, we expect that putative ABI3 target genes have log-ratios that are significantly greater than zero in repeated experiments. For each of the five replicated experiments, we create a list containing all of the promoter identifiers of the ChIP-chip profiles of the five chromosomes in decreasing order of their log-ratios. That is, promoters with log-ratios significantly greater than zero are at the top of this list. We use these five lists to determine the intersection of the top

k candidate promoters of each list. This proceeding allows to assess the degree of reproducibility between the five replicates. We interpret all genes belonging to the promoters in the intersection as putative target genes of ABI3.

2.3 Hidden Markov Model (HMM) for target gene detection

HMM description: We use a two-state HMM $\lambda = (S, \pi, A, E)$ with Gaussian emission densities for the genome-wide detection of putative ABI3 target genes. The basis of this HMM is the set of states $S = \{-, +\}$. State $-$ corresponds to a promoter that is not a target of ABI3, and state $+$ corresponds to a promoter that is a target of ABI3. We denote the state of promoter t by $q_t \in S$, and we assume that a state sequence $q = q_1, ..., q_T$ belonging to a ChIP-chip profile o is generated by a homogeneous Markov model of order 1 with start distribution $\pi = (\pi_-, \pi_+)$ and stochastic transition matrix $A = (a_{ij})_{i,j \in S}$ where $\pi_-, a_{--}, a_{++} \in (0, 1)$, $\pi_+ = 1 - \pi_-$, $a_{-+} = 1 - a_{--}$, and $a_{+-} = 1 - a_{++}$. The state sequence is assumed to be not observable, i.e. hidden, and the log-ratio o_t of promoter t is assumed to be drawn from a Gaussian emission density, whose mean and standard deviation depend on state q_t. We denote the vector of emission parameters by $E = (\mu_-, \mu_+, \sigma_-, \sigma_+)$ with means μ_- and μ_+, and standard deviations σ_- and σ_+ for the Gaussian emission density $b_i(o_t) = 1/(\sqrt{2\pi}\sigma_i) \exp(-0.5(o_t - \mu_i)^2/\sigma_i^2)$ of log-ratio o_t given state $i \in S$.

HMM initialization: In general, an initial HMM has to discriminate ABI3 target promoters from non-target promoters with respect to their log-ratios in the ChIP-chip profile. Hence, a histogram of log-ratios of all five replicates helps to find good initial HMM parameters. The choice of initial parameters addresses the presumptions that the proportion of non-target promoters is much higher than that of target promoters, and that the number of successive non-target promoters is also much higher than the number of successive target promoters. In our case study we use $\pi_- = 0.9$ resulting in an initial start distribution $\pi = (0.9, 0.1)$. Thus, we choose an initial transition matrix A with equilibrium distribution π. That is, we set $a_{--} = 1 - s/\pi_-$ and $a_{++} = 1 - s/\pi_+$ with respect to the scale parameter $s = 0.05$ to control the state durations. We characterize the states by proper means and standard deviations using initial emission parameters $\mu_- = 0$, $\mu_+ = 2$, $\sigma_- = 1$, and $\sigma_+ = 0.5$. We refer to the initial HMM by λ^1.

HMM training: We train the initial HMM based on all ChIP-chip profiles using a maximum a posteriori (MAP) variant of the standard Baum-Welch algorithm ([Rab89], [DEKM98]). This algorithm is part of the class of EM algorithms ([DLR77]), which iteratively maximize their optimization function. With respect to the underlying biological question, the choice of the prior influences the quality of the trained HMM. We include biological a priori knowledge into the MAP training using a Dirichlet prior with hyper-parameters $\vartheta_- = \vartheta_+ = 2$ for start distribution π, a product of Dirichlet priors with hyper-parameters $\vartheta_{ab} = 1$ with $a, b \in S$ for transition matrix A, and a product of Normal-

Gamma priors for emission parameters E with hyper-parameters $\eta_- = 0$ and $\eta_+ = 2$ (a priori means), $\epsilon_- = \epsilon_+ = 1,000$ (scale of a priori means), $r_- = 1$ and $r_+ = 100$ (shape of standard deviations), and $\alpha_- = \alpha_+ = 10^{-4}$ (scale of standard deviations). The choice of these prior parameters ensures a good characterization of both HMM states. On that basis we iteratively maximize the posterior of the HMM λ^h given all ChIP-chip profiles resulting in new HMM parameters λ^{h+1}. We stop the MAP training if the increase of the log-posterior of two successive MAP iterations is less than 10^{-9}.

HMM target gene detection: The state $+$ of the trained HMM λ models the potential of promoters to be targets of ABI3. To quantify this potential we calculate the probability $\gamma_t(+) = P[Q_t = +|O = o, \lambda]$ for each promoter t within a ChIP-chip profile o to be a target promoter. This state posterior of state $+$ is computed using the Forward-Backward procedures of HMMs ([Rab89], [DEKM98]). For each of the five replicated experiments we create a list containing all of the promoter identifiers of the ChIP-chip profiles of the five chromosomes in decreasing order of their state posteriors $\gamma_t(+)$. We use these five lists to determine the intersection of the top k candidate promoters of each list. In analog to the standard LFC approach, we interpret all genes belonging to the promoters in the intersection as putative target genes of ABI3.

2.4 Hidden Markov Model with scaled transition matrices (SHMM) for target gene detection

SHMM description: The general concept of SHMMs enables us to analyze ChIP-chip profiles in the context of orientations of neighboring genes on the DNA. Two directly neighboring genes on DNA occur either in head-head, tail-tail, tail-head, or head-tail orientation to each other. Among these orientations the head-head orientation is of prime importance for the analysis of promoter array data. In this orientation the two corresponding genes have the potential to share a common promoter region depending on the distance between these genes. This fact in combination with the observation that the log-ratios of promoters for genes in head-head orientation show significantly higher correlations compared to all other orientations is the basis to design a specific SHMM. We assume that it is more likely for two genes in head-head orientation to show the same promoter status, that means either ABI3 target or not, in comparison to all other orientations. For that reason we assign to each pair of successive promoters t and $t+1$ of a chromosome one promoter pair orientation class $c(d_t)$ depending on the orientation of both promoters to each other in combination with the chromosomal distance d_t of the two genes belonging to these promoters. The promoter pair orientation class of successive promoters t and $t+1$ is

$$c(d_t) \quad = \quad \begin{cases} 2, & t \text{ and } t+1 \text{ are head-head and } d_t \leq b \\ 1, & \text{otherwise} \end{cases}$$

using a pre-defined distance threshold $b \in \mathbb{N}$. We incorporate these information into a two-state SHMM $\lambda_L = (S, \pi, A, \vec{f}, E)$ with $L = 2$ promoter pair orientation classes to

detect putative ABI3 target genes. The parameters S, π, A, and E are defined like in the HMM approach, and $\vec{f} = (f_1 := 1, f_2)$ with $f_2 \in \mathbb{R}^+$ and $f_2 > f_1$ is the vector of scaling factors. In contrast to the standard HMM approach, we now assume that the state sequence q of a ChIP-chip profile o is generated by an inhomogeneous Markov model of order 1 with start distribution π and two scaled stochastic transition matrices A_1 and A_2 for discriminating head-head orientations from others based on the promoter pair orientation classes. The transition matrix A_l with $l \in \{1, 2\}$ is defined by

$$A_l \quad = \quad \frac{1}{f_l} \begin{pmatrix} a_{--} - 1 + f_l & a_{-+} \\ a_{+-}, & a_{++} - 1 + f_l \end{pmatrix}$$

with respect to the scaling factor f_l that scales the expected state duration of state $i \in S$ in A from $1/(1-a_{ii})$ to $f_l/(1-a_{ii})$ in A_l. A transition from state q_t to state q_{t+1} is achieved by using the corresponding transition matrix $A_{c(d_t)}$ based on the integrated promoter pair orientation class $c(d_t)$. The self-transition probability of each state $i \in S$ increases strictly from matrix A_1 to A_2, and thus, for a head-head promoter pair that is modeled by A_2 it is more likely that both promoters are targets or no targets of ABI3 compared to other promoter pairs modeled by A_1. The log-ratios of promoters are modeled as described in the HMM approach.

SHMM initialization: The basic initialization of the SHMM is done like for the HMM. In addition to that, we must choose a distance threshold b for the promoter pair orientation classes and a scaling factor f_2 to specify the degree of differentiation between head-head orientation modeled by A_2 and all others modeled by A_1. Motivated by Fig. 1 we always use $b = 9$Kbp in our case study because in greater chromosomal distance the correlations of head-head promoter pairs do not significantly differ from others. Moreover, we consider all f_2 from 1.1 to 10 in steps of 0.1.

SHMM training: The SHMM is trained like the HMM using the MAP variant of the Baum-Welch algorithm with identical prior hyper-parameters. The only difference between both models occurs during the estimation of their transition matrices. Details of the parameter estimation are described by [Sei06].

SHMM target gene detection: The putative target genes of ABI3 are determined in analog to the HMM approach. The calculation of the state posterior $\gamma_t(+)$ is now done with respect to the promoter pair orientation classes using the Forward-Backward procedures of HMMs.

3 Results and discussion

3.1 Study of differences between HMM and SHMMs

The HMM approach enables us to analyze ChIP-chip data in the context of chromosomal locations of promoters, and the application of SHMMs extends this analysis by discriminating different types of promoter pair orientations. In a first study, we investigate how SHMMs behave compared to the standard HMM. For that reason, we use the Viterbi algorithm ([Rab89], [DEKM98]) to compare the most likely state sequence q for a ChIP-chip profile o under the trained HMM to that of all trained SHMMs with scaling factor f_2 increasing from 1.1 to 10 in steps of 0.1. Here, the annotation of a promoter t with log-ratio o_t is given by $q_t \in S$, which we interpret as this promoter is either a putative ABI3 target or not. The scaling factor allows to directly influence the annotation behavior for head-head promoters. That is, the higher f_2 the more likely it is that both promoters of such head-head pairs are either putative ABI3 targets or not, and the closer we choose f_2 to one the closer is the annotation behavior of the SHMM to that of the HMM. The results are illustrated in Fig. 2a. We observe that the number of head-head promoter pairs where both promoters of such a pair have identical annotations increases for increasing scaling factor f_2, and as consequence the number of head-head promoter pairs where both promoters of such a pair have different annotations decreases. Obviously, each change in the annotation of a head-head promoter pair leads either to a change in the annotation of the upstream, downstream, or both of these promoter pairs. We see that the number of non-head-head promoter pairs where both promoters of such a pair are annotated as putative ABI3 targets decreases only slightly for SHMMs with increasing scaling factor f_2 compared to the HMM. We clearly see substantially more decrease in the number of non-head-head promoter pairs where both promoters of such a pair are annotated as putative non-target promoters for SHMMs with increasing scaling factor f_2 in relation to the HMM. Consequently, the number of non-head-head promoter pairs where both promoters of such a pair have different annotations increases with increasing scaling factor f_2. This study demonstrated that the annotation results of SHMMs can differ significantly from that of the HMM resulting in a more general model for the prediction of putative target genes.

3.2 Comparison of LFC, HMM, and SHMM to predict ABI3 target promoters

We use the LFC method for scoring putative ABI3 target promoters based on the log-ratios of the promoters neglecting chromosomal locations and promoter pair orientations. For comparison, we make use of the HMM that models chromosomal locations of promoters and the SHMM that models chromosomal locations and orientations of promoter pairs whereas both methods score putative ABI3 target promoters via the state posterior of state $+$. In this comparison study we set the threshold for the maximal number of candidates in a top list to 200 because the mean log-ratio of 1.06 at this level is already relatively small, and beyond, at a threshold of 300 we did not get new putative ABI3 target genes by the three methods. Moreover, we use the SHMM with scaling factor $f_2 = 4$ in all further

analyses because this model is already quite different from the standard HMM (Fig. 2a), and the comparison of this model to SHMMs with scaling factor $f_2 = 6$ and $f_2 = 10$ yielded identical target promoters. For each approach, we score all five experiments to determine the intersection of putative ABI3 target promoters for the top 50, 100, 150 and 200 candidates under these experiments. Then, we use Venn diagrams to directly compare the candidate promoters for these four top lists under all three methods. The results are shown in Fig. 2b. We observe that the SHMM predicted the greatest number of putative ABI3 target promoters, whereas the LFC method predicted the smallest number. When we consider the Venn diagrams from the top 100 list to the top 200 list all candidates that are predicted by the LFC method are also completely predicted by both the HMM and the SHMM. In addition to this, the candidates additionally predicted by the HMM from the top 150 list to the top 200 list are completely predicted by the SHMM. In summary, this emphasizes that the SHMM approach tends to be more general in the prediction of putative ABI3 target promoters than the HMM and the LFC approach.

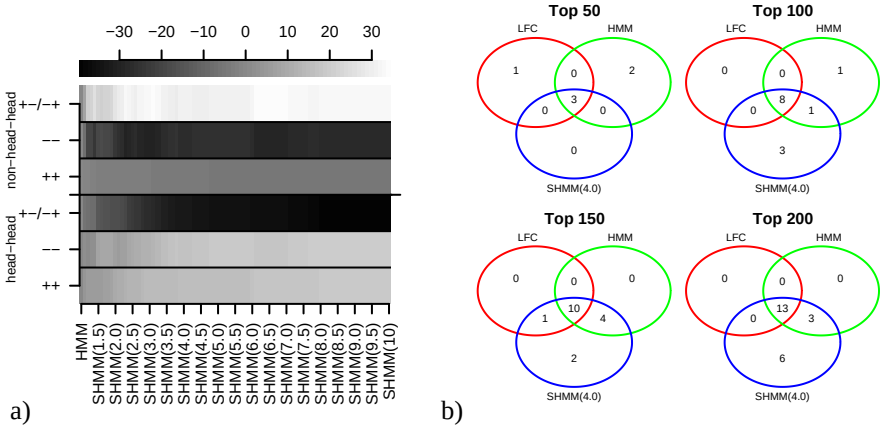


Figure 2: **a)**: Frequencies of promoter pair annotations of the trained SHMM(f_2) with scaling factor $f_2 \in [1.1, 10]$ in steps of 0.1 in relation to the trained HMM based on Viterbi annotations. The grey gradient in the upper part expresses the quantity of annotation differences whereas the HMM is encoded by the grey with value zero. The annotations $++$, $--$, and $+-/-+$ of promoter pairs mean that either both promoters are putative targets, non-targets, or only one promoter is a putative target of ABI3. **b)**: Venn diagrams to compare putative ABI3 target promoters predicted by the LFC method, the trained HMM, and the trained SHMM(4.0).

3.3 Gene expression analysis of putative ABI3 target genes belonging to predicted ABI3 target promoters

Next we investigate how putative target genes are regulated by ABI3. Therefore, we use Genevestigator [ZHHHG04] as independent source of *A. thaliana* gene expression data to analyze putative target genes. In Genevestigator, ABI3 is mainly expressed within the categories inflorescence, silique, and seed. Based on that, we quantify the expression of all putative target genes by dividing the sum of expression values within these three categories

by the sum of expression values in all categories. This provides a quantitative measure, which we call Genevestigator score, for analyzing how a putative ABI3 target gene follows the expression profile of ABI3. Additionally, transient assays have been performed to test whether putative target promoters in fusion with the glucuronidase (GUS) reporter gene react on ABI3. The results are shown in Tab. 1. Calculating the Genevestigator score, 16 of 22 putative target genes show significantly high scores at the level of the 95%-quantile 0.15 based on the distribution of the Genevestigator scores for 1,000 randomly selected genes. The promoters of these 16 genes have been tested in transient assays, and we find that 15 of these promoters can activate the GUS expression through ABI3, and the promoter of gene T21 shows nearly a two-fold repression of the GUS expression. Interestingly, the genes T21 and T22 are in head-head orientation to each other, and so they have the potential to share a common promoter region. Based on the results of the transient assays the first gene might be repressed while the second is activated. Hence, it seems that activation and repression signals can be transmitted by ABI3 to these two target genes in head-head orientation via a potential common promoter region. Additionally, we point out that only the SHMM approach was able to predict 3 of these 15 target genes activated by ABI3 and the one target gene repressed by ABI3. In contrast to these 16 target genes, the 6 remaining putative target genes do not significantly differ in their Genevestigator scores at the level of the 5%-95%-quantile range $[0.02, 0.15]$ based on the distribution of the Genevestigator scores for the 1,000 randomly selected genes. Interestingly, 5 of these 6 putative target genes are in head-head orientation to one of the previous target genes activated by ABI3, and so the potential common promoter region can already receive signals from ABI3. Next we address the question if these 6 putative ABI3 target genes are also under control of ABI3 via the potential common promoter region. To test this hypothesis, the promoters of 4 of these 6 putative target genes have been tested in transient assays. The promoters of the genes T2 and T11 show a low activation of the GUS expression, the promoter of gene T13 shows a two-fold repression of the GUS expression, and the promoter of gene T9 does not seem to react on ABI3. In addition to this, gene T13 is in head-head orientation with gene T23 that is not represented by its own proximal promoter fragment on the promoter arrays. The Genevestigator score of T23 is significantly higher than those of the 1,000 random genes at the level of the 95%-quantile, and the promoter of this gene shows activation of the GUS expression in a transient assay. Hence, this gene pair seems to behave like the gene pair T21 and T22. In summary, independent gene expression profiles from Genevestigator give first hints which genes might be activated by ABI3. Additionally, transient assays help to validate this results if the underlying test system is capable of simulating the natural situation in seeds. Twenty percent of the ABI3 activated target genes with high Genevestigator scores could only be predicted through the application of the SHMM approach and would have been missed using the LFC or HMM approach. Moreover, the SHMM predicted over forty percent more putative ABI3 target genes compared to the LFC method. For these 9 genes the promoters of 7 have been tested in transient assays whereas 1 promoter does not react, 1 represses the GUS expression, and the 5 others activate the GUS expression. This results emphasize the relevance of SHMMs in the detection of ABI3 target genes.

ID	LFC	HMM	SHMM(4.0)	Genevestigator	Transient Assay
T1	1	1	1	0.94	5
T2	1	1	1	0.11	2.5
T3	1	1	1	0.86	12
T4	0	0	1	0.03	-
T5	0	0	1	0.39	3
T6	1	1	1	0.72	15
T7	1	1	1	0.90	7
T8	0	0	1	0.46	12
T9	0	0	1	0.07	1
T10	0	0	1	0.95	6
T11	0	1	1	0.09	2
T12	1	1	1	0.74	24
T13	1	1	1	0.09	0.4
T14	1	1	1	0.93	8
T15	0	1	1	0.10	-
T16	1	1	1	0.95	27
T17	1	1	1	0.98	27
T18	0	1	1	0.98	27
T19	1	1	1	0.98	27
T20	1	1	1	0.57	8
T21	0	0	1	0.20	0.6
T22	1	1	1	0.81	30

Table 1: Overview of predicted ABI3 target genes at the level of the top 200 candidates in Fig. 2b. The ID column contains anonymized target gene identifiers (our biologists prepare a manuscript discussing target genes). The numbers 1 and 0 in the method columns LFC, HMM, and SHMM(4.0) encode whether a gene is predicted or missed. Genevestigator quantifies the gene expression of a target gene within the categories inflorescence, silique, and seed as described in Section 3.3. Transient Assay contains the measured fold-change for a target gene promoter under ABI3 expression vs. target gene promoter lacking ABI3 expression.

4 Conclusions and outlook

We introduced the LFC, the HMM, and the SHMM approach for the analysis of ChIP-chip promoter array data and compared these methods on ABI3 ChIP-chip data. The motivation for the usage of HMMs is based on the observation of positive correlations between ChIP-chip measurements of adjacent promoters on the DNA (Fig. 1). Especially, the SHMM approach is motivated by the fact that ChIP-chip measurements of head-head promoter pairs show significantly higher correlations than those of other promoter pair orientations. Based on SHMMs, we demonstrated that discriminating promoters in head-head orientations from other promoter orientations can lead to significantly different predictions of target and non-target promoters compared to the HMM (Fig. 2a). Regarding all three methods, the SHMM predicted the highest number of putative ABI3 target promoters and all target promoters predicted by the LFC or the HMM have been included (Fig. 2b), but the number of predicted putative ABI3 target promoters is not an optimal criterion to decide which of the methods should be preferred. For this reason, we used publicly available expression profiles from Genevestigator to analyze how a putative target gene follows the expression profile of ABI3, and transient assays have been performed to test whether the promoter of a putative target gene reacts on ABI3 (Tab. 1). We showed that expression data from Genevestigator can give first hints which genes might be activated by ABI3, and that the validation can be done by transient assays. Twenty percent of the target genes with significantly high Genevestigator scores and activation in transient assays could only be predicted by the SHMM and would have been missed by the LFC or HMM approach. In total, the SHMM predicted more than forty percent more putative target promoters (9 of 22) compared to the LFC method. Seven of these promoters have been tested in transient assays whereas one promoter does not react, one represses the GUS expression, and the five others activate the GUS expression. Taking this together, we conclude that the SHMM

can be seen as the more general model that should be preferred for the prediction of ABI3 target genes. We conjecture that the proposed SHMM might possibly be useful for the analysis of other promoter array ChIP-chip data.
In the future, the study of seed development continues. For instance, we are awaiting ChIP-chip data of the transcription factors LEC1, LEC2, and FUS3. This will provide us first insights into the transcriptional regulatory network involved in seed development. In cooperation with us, our biologists prepare a manuscript with details to the ABI3 ChIP-chip experiments including the discussion of ABI3 target genes.

5 Acknowledgments

We thank the groups of Lothar Altschmied, Helmut Bäumlein, and Udo Conrad and especially Urs Hähnel and Gudrun Mönke for ChIP-chip data, transient assays, and valuable discussions. This work was supported by the BMBF grants 0312706A and 0313155, and by the Ministry of culture Saxony-Anhalt grant XP3624HP/0606T.

References

[BIAS03] BM Bolstad, RA Irizarry, M Astrand, and TP Speed. A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics*, 19(2):185–193, 2003.

[DEKM98] R Durbin, S Eddy, A Krogh, and G Mitchision. *Biological sequence analysis - Probabilistic models of proteins and nucleic acids*. Cambridge University Press, 1998.

[DLR77] A Dempster, N Laird, and D Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38, 1977.

[IHS+01] VR Iyer, CE Horak, CS Scafe, D Botsein, M Snyder, and PO Brown. Genomic binding sites of the yeast cell-cycle transcription factors SFB and MBF. *Nature*, 409:533–538, 2001.

[JLG+08] DS Johnson, W Li, DB Gordon, A Bhattacharjee, B Curry, and L Brizuela et al. Systematic evaluation of variability in ChIP-chip experiments using predefined DNA targets. *Genome Res*, 18:393–403, 2008.

[JW05] H Ji and WH Wong. TileMap: create chromosomal map of tiling array hybridizations. *Bioinformatics*, 21(18):3629–3636, 2005.

[KSSW03] B Knab, A Schliep, B Steckemetz, and B Wichern. Model-based clustering with Hidden Markov Models and its application to financial time-series data. *In M. Schader, W. Gaul, and M. Vichi, editors, Between Data Science and Applied Data Analysis, Springer*, pages 561–569, 2003.

[KvdLDC04] S Keles, MJ van der Laan, S Dudoit, and SE Cawley. Multiple testing methods for ChIP-chip high density oligonucleotide array data. *Working Paper Series 147*, 2004. U.C. Berkeley Division of Biostatistics, University of California, Berkeley, CA.

[LML05] W Li, CA Meyer, and XS Liu. A hidden Markov model for analyzing ChIP-chip experiments on genome tiling arrays and its application to p53 binding sequences. *Bioinformatics*, 21:i274–i282, 2005.

[MD04] I M Meyer and R Durbin. Gene structure conservation aids similarity based gene prediction. *Nucleic Acids Research*, 32(2):776–783, 2004.

[Rab89] L Rabiner. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[RRW+00] B Ren, F Robert, JJ Wyrick, O Aparicio, EG Jennings, I Simon, J Zeitlinger, J Schreiber, N Hannett, E Kanin, TL Volkert, CJ Wilson, SP Bell, and RA Young. Genome-Wide Location and Function of DNA Binding Proteins. *Science*, 290(5500):2306–2309, 2000.

[Sei06] M Seifert. Analysing Microarray Data Using Homogeneous And Inhomogeneous Hidden Markov Models. Diploma Thesis; Martin Luther University; seifert@ipk-gatersleben.de, 2006.

[VCC05] J Vicente-Carbajosa and P Carbonero. Seed maturation: developing an intrusive phase to accomplish a quiescent state. *Int. J. Dev. Biol.*, 49:645–651, 2005.

[ZHHHG04] P Zimmerman, M Hirsch-Hoffman, L Hennig, and W Gruissem. GENEVESTIGATOR. Arabidopsis Microarray Database and Analysis Toolbox. *Plant Physiol.*, 136:2621–2632, 2004.

Temporal Analysis of Oncogenesis Using MicroRNA Expression Data

Thomas Zichner,[1,2] Zelmina Lubovac,[1] Björn Olsson[1]

[1]Bioinformatics Research Group, Systems Biology Centre, Department of Life Sciences, University of Skövde, Box 408, S-54128 Skövde, Sweden
[2]Institut für Informatik, Friedrich-Schiller-Universität Jena, Ernst-Abbe-Platz 2, D-07743 Jena, Germany
tzi@minet.uni-jena.de, zelmina.lubovac@his.se, bjorn.olsson@his.se

Abstract: MicroRNAs (miRNAs) have rapidly become the focus of many cancer research studies. These small non-coding RNAs have been shown to play important roles in the regulation of oncogenes and tumor suppressors. It has also been demonstrated that miRNA expression profiles differ significantly between normal and cancerous cells, which indicates the possibility of using miRNAs as markers for cancer diagnosis and prognosis. However, not much is known about the regulation of miRNA expression. One of the issues worth investigating is whether deregulations of miRNA expression in cancer cells occur according to some pattern or in a random order. We therefore selected two approaches, previously used to derive graph models of oncogenesis using chromosomal imbalance data, and adapted them to miRNA expression data. Applying the adapted algorithms to a breast cancer data set, we obtained results indicating the temporal order of miRNA deregulations during tumor development. When analyzing the specific deregulations appearing at different time points in the derived model, we found that several of the deregulations identified as early events could be supported through literature studies.

1 Introduction

One of the important issues that have dominated cancer research during the last decade has been to identify molecular biomarkers [Lu05], i.e., indicators of cancer staging and tumor subtypes. MicroRNAs (miRNAs) have been seen as potential biomarkers that not only may serve for diagnostic and prognostic purposes, but are also assumed to have a great therapeutic potential in cancer [Sa08].

MicroRNA expression profiles have been used in previous work to classify tumors and differentiate between normal and cancerous tissue [Io05][Lu05][Vo06]. To the best of our knowledge, however, there are no existing approaches that consider changes in miRNA expression patterns during cancer progression, in order to reveal the possible temporal ordering of aberrant miRNA expression. Therefore, we selected two existing methods for deriving graph models of oncogenesis, previously applied to comparative genomic hybridization data, and applied them to miRNA expression data. The purpose with the adapted methods is to derive models illustrating the temporal order of events

during cancer progression. A deeper understanding of miRNAs during tumor progression, including the temporal order of their deregulations, may lead to novel methods regarding the prediction of survival of cancer patients and the choice of treatment, as well as for cancer subtype prediction.

2 Method

2.1 Data set

To perform temporal analyses of miRNAs a data set generated in [Io05] was used. It contains the expression levels of 157 human miRNAs and 69 human precursor miRNAs in 109 breast cancer samples (primary tumor samples as well as human breast cancer cell lines) and in normal breast tissue. The normal samples consisted of six pools of five normal breast tissues each and four additional single breast tissues, of which we used only two because of an observed unreasonable deviation of the other two. Further details about the used data set can be obtained from [Io05]. The raw data can be obtained from ArrayExpress [Br03], which can be accessed via http://www.ebi.ac.uk/microarray-as/aer/. The ID of the used data set is E-TABM-23. For our analyses, we used the normalized data set kindly provided by Marilena V. Iorio.

2.2 Determining aberrant expression

The first step of the analysis was to determine the subset of miRNAs that are most likely to have aberrant expression in breast cancer compared to normal tissue. In all further analyses we focused only on this subset of miRNAs. To derive this set, the two-sample Kolmogorov-Smirnov test as well as the Wilcoxon rank sum test was applied to each miRNA's expression profile. For both tests, the null hypothesis is that the expression values of a certain miRNA are derived from the same distribution for the normal as well as the cancer samples. All microRNAs with $p < 0.05$ in both tests were considered as deregulated and included into the subset. The reason for choosing the Kolmogorov-Smirnov and Wilcoxon rank sum tests is that they do not make any assumptions concerning the value distribution.

The next step was to determine in which breast cancer tumor samples each miRNA was aberrantly expressed. The assumption is that expression values in tumor samples which differ by more than two standard deviations σ from the mean expression value μ in normal tissue can be considered as deregulated. This method is commonly used in microarray analysis to identify differentially expressed genes [CQB03][Ka06].

For the further work, we classified each miRNA in each tumor sample as under-, normal, or over-expressed, depending on whether the expression value was lower than μ-2σ, between $\mu - 2\delta$ and $\mu + 2\delta$, or greater than $\mu + 2\delta$.

2.3 Creating a set of events

The approach to determine aberrant expression described in section 2.2 results in a matrix *D* showing single deregulations, i.e., single cases of under- and over-expression. Assuming *k* as the number of deregulated miRNAs (i.e., the size of the miRNA subset) and *m* as the number of tumor samples, matrix *D* is defined by:

$$D = (d_{ij})_{\substack{1 \le i \le k \\ 1 \le j \le m}}$$

where d_{ij} = -1 if miRNA *i* is under-expressed in sample *j*, d_{ij} = 0 if it is normally expressed, and d_{ij} = 1 if it is over-expressed.

For further analyses, we distinguished between deregulations (i.e., the combination of a particular miRNA and the kind of deregulation) instead of just miRNAs, because a miRNA may be over-expressed in some tumor samples and under-expressed in some others. This distinction is necessary since there might be different causes for the different kinds of deregulations. Each pair of a miRNA and a type of deregulation is also referred to as an *event*. Thus, instead of *k* miRNAs, 2*k* events are considered.

Since it is quite difficult to make reliable assumptions about events which occur very rarely we restricted the further analyses to events that were present in at least 15% of the tumor samples. This resulted in an event matrix *E*:

$$E = (e_{ij})_{\substack{1 \le i \le n \\ 1 \le j \le m}}$$

where e_{ij} = 1 if event *i* is present in sample *j*, and e_{ij} = 0 otherwise. Furthermore, *n* denotes the number of events and $\sum_{j=1}^{m} e_{ij} \ge 0.15 \cdot m$ for all $1 \le i \le n$.

2.4 Temporal analysis and generating graph models of oncogenesis

Following the approaches by Höglund et al. [Hö01][Hö05] and Beerenwinkel et al. [Be05], the miRNA data set was analyzed and graph models representing the events during cancer progression generated. The following text explains the steps of the methods.

The approach for analyzing temporal relations described here is adapted from [Hö01]. Considering all tumor samples that show a certain deregulation, the average number of simultaneously observed events is calculated. If there are only a few such events, the considered deregulation is assumed to be an early event in the oncogenesis. Otherwise, i.e., if there are many simultaneously occurring events, it is assumed to be a late event. This reasoning is based on the well-established knowledge that tumors in late stages have usually accumulated a large number of mutations, leading to genomic instability [To02].

Instead of considering gains and losses of chromosomal parts, as in [Hö01], we considered the over- and under-expression of miRNA genes as events. First, for each

tumor sample, the number of miRNAs which are considered to have aberrant expression in the sample is calculated. In the following, this number is called NDPT - number of deregulations per tumor sample. Secondly, for each event, the NDPT is recorded for every tumor sample in which the event is present. By determining the median NDPT, the time of occurrence (TOC) of an event can be estimated.

To identify possible patterns in the data set, we performed a principal component analysis (PCA). PCA is a multivariate method frequently used to search for underlying structures in the data. It is a technique to reduce multidimensional data sets to lower dimensions. Briefly, principal components are linear combinations of the original variables, orthogonal, and ordered with respect to their variance so that the first principal component has the largest variance. The idea of applying PCA is to retain just those characteristics of a data set that contribute most to its variance. We performed the PCA with the deregulations as variables and the tumor samples as observations. To show the results we plotted all deregulations in relation to the first two principal components, which explain more than 40% of the total variance in the miRNA data set.

As the third analysis, we built oncogenetic tree mixture models, a graph-based representation of oncogenetic pathways, using the approach proposed by Beerenwinkel et al. [Be05a]. In addition to the temporal ordering, tree models also indicate possible alternative pathways of tumor development, characterized by different combinations (and/or orderings) of events. They thus have the potential to provide insights about tumor subtypes. Details about the approach by Beerenwinkel et al. can be found in [Be05a][Ra05]. It is a further development of an approach originally proposed by Desper et al. [De99], which is based on an algorithm for finding minimum weight branching trees. We used the software package mtreemix [Be05b] (http://mtreemix.bioinf.mpi-sb.mpg.de/) to learn the tree models with the event matrix E as input.

3 Results

3.1 Determining aberrant expression

The subset of miRNAs that are most likely to be deregulated in breast cancer was first selected, as explained in section 2.2. The two statistical tests, applied to find significant deregulations, identified 48 miRNAs that are significant according to both tests. This set of miRNAs was used in further analysis. The next step was to determine in which of the breast cancer tumor samples a certain miRNA is aberrantly expressed, as explained in section 2.2. The distribution of the number of deregulations per miRNA (NDPM) as well as the distribution of the number of deregulations per tumor sample (NDPT) is shown in Figure 1. A table showing the *p*-values and the number of tumor samples in which each miRNA is considered as over- or under-expressed is available from the authors.

Figure 1 and the table resulted in some observations regarding specific miRNAs. The miRNA mir-210 has the lowest *p*-value in both tests. The most down-regulated miRNA is mir-19a, while the most up-regulated miRNA is mir-21. It is also apparent that the variance in the number of deregulations per miRNA is large.

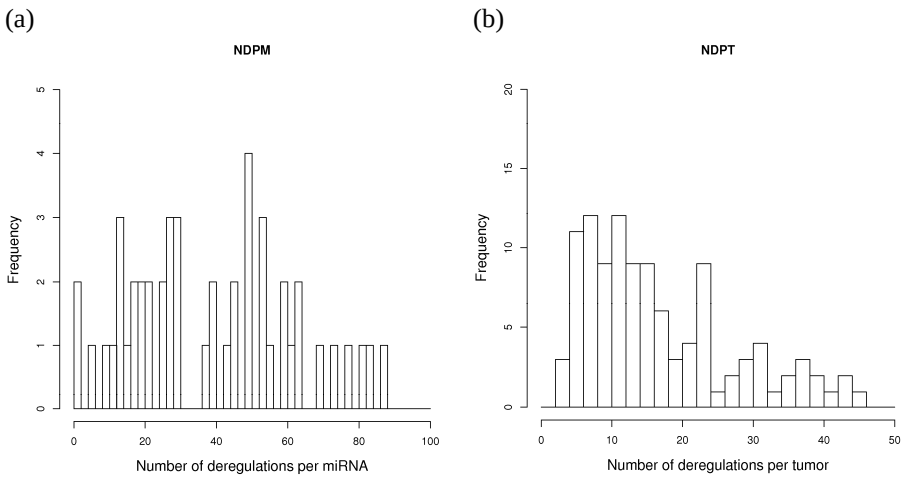


Figure 1: Distribution of the number of deregulations per miRNA (a) and the number of deregulations per tumor sample (b). No distinction was made between over- and under-expression.

In the following, we examined events, instead of considering only the miRNAs. As already described, an event is defined as a pair consisting of a specific miRNA and a specific kind of deregulation, i.e., either over- or under-expression. Events are here denoted by the miRNA name followed by a plus or minus sign. Thus, mir-125b-1− signifies the event that mir-125b-1 is under-expressed. Note that an event is not the same as an observation, since the same event (e.g., mir-125b-1−) may be observed in a large number of tumor samples. As we only considered events that were observed in at least 15% of the samples, the resulting set used in further analysis consisted of 36 events.

3.2 Temporal analysis

We here adapted the approach from [Hö01], with the aim to reveal the temporal ordering of miRNA deregulation events. As described in 2.4, the events are ordered according to the time of occurrence, estimated by the number of co-occurring events. The results, shown in Figure 2, indicate an evident temporal ordering of the events. On average, events like mir-125b-1− co-occur with significantly fewer events than, for instance, event mir-208−. Thus, it can be assumed that mir-125b-1− is an early event compared to mir-208−, which indicates that mir-125b-1 may play an important role in the onset of tumor development.

PCA was performed to determine the underlying structure behind the data. The aim is to calculate the largest principal components which can describe most of the overall variance shown in the data. The number of principal components equals the number of variables, i.e., in our case the number of events. The PCA resulted in 36 components, of which the first two explain about 45% of the total variance, and the first three components explain more than the half of the total variance. All considered events are plotted in relation to the first two components in Figure 3. It can be seen that most events

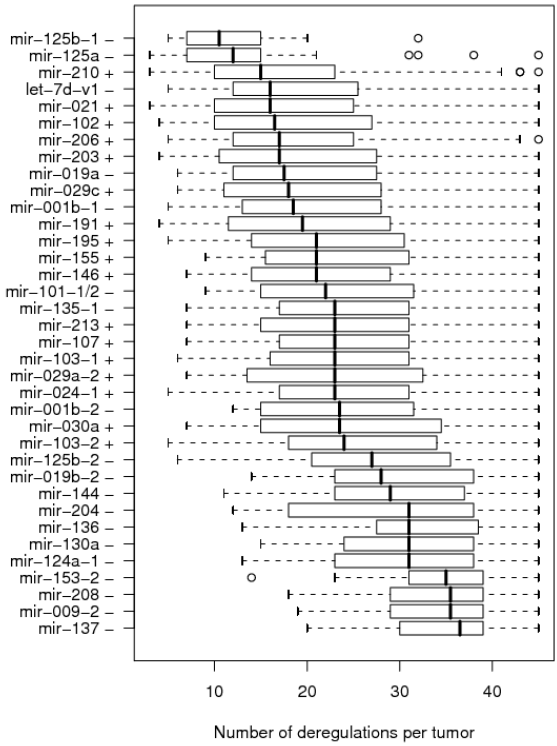


Figure 2: Estimated time of occurrence of the events. The sign '+' after a miRNA name indicates over-expression, while '−' indicates under-expression. Vertical bars indicate the median of the number of deregulations per tumor (NDPT) of the corresponding samples, boxes show the boundaries of the 25th and 75th percentiles, and the outer lines indicate the non-outlier minimum and maximum values.

differ only in relation to the second component, except for the events we assumed to occur as a cluster (see Figure 3). Further, there are also sets of events with almost identical values regarding the first two components.

3.4 Building oncogenetic tree mixture models

As a final step in the analysis, we built oncogenetic tree mixture models [Be05a] using the software package mtreemix developed by Beerenwinkel et al. [Be05b]. The first tree model that we derived shows all considered events. The second model shows a subset of events generated by a method proposed in [Br82]. These two models are not shown due to space limitations, but are available from the authors on request. The third model shows a subset of 15 events (Figure 4), which were selected because they were more separated in the PCA-plot (Figure 3) and because they were also identified in [Io05]. The number of trees K in the mixture model was set to two (i.e., one non-noise component) for the first two models. In the third model K was set to three according to the estimation of mtreemix select.

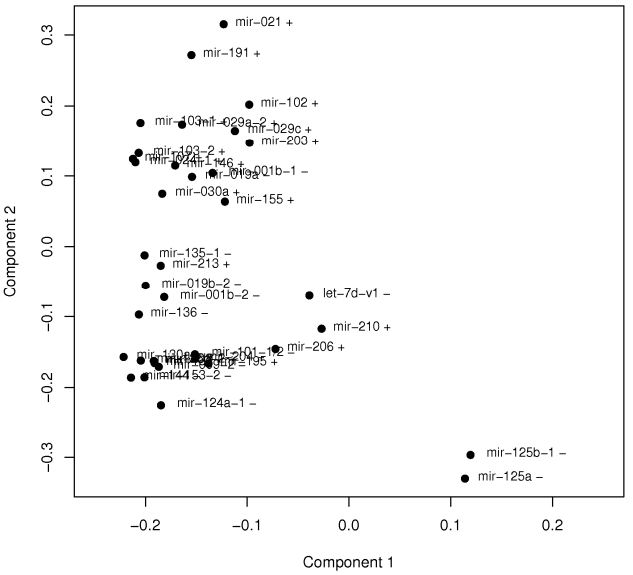


Figure 3: Considered events plotted in relation to the first two components of the PCA. An interesting observation is that in the co-occurrence statistics (data not shown) mir-206+, mir-210+, let-7d-v1−, mir-125a−, and mir-125b-1− are separated and form a cluster, i.e., they occur very often together, but rather seldom in the presence of other deregulations.

From the figures it can be seen that the mixture model that includes all events is not very stable (several edges were not present in any of the 1000 bootstrap iterations). Also, it may be observed that there are several edges which are present in all three or at least two models. For instance:

wild type → mir-021+ wild type → mir-102+ mir-125a− → mir-125b-1−

The edge "wild type → mir-021+" is present in all models. This applies also to the edge between mir-125a− and mir-125b-1−; however, in this case both directions are present.

3.5 Evaluation

We evaluated our results by applying the methods to randomly altered data sets. For each miRNA the order of expression values (tumor samples) was randomly altered. Figure 5 shows the results of temporal order analysis and PCA for the randomized data. In contrast to the original data set, it can be easily seen that there is no detectable temporal order of events (Figure 5a). There is only a difference of about 2 between the median numbers of deregulations per tumor (NDPT) of the "earliest" and the "latest" event, compared to a difference of about 27 when considering the original data set. Additionally, there are two large sets of events (14 and 15 items, respectively) which have the same median NDPT, and thus considered as simultaneously occurring, which also indicates the absence of a significant temporal order.

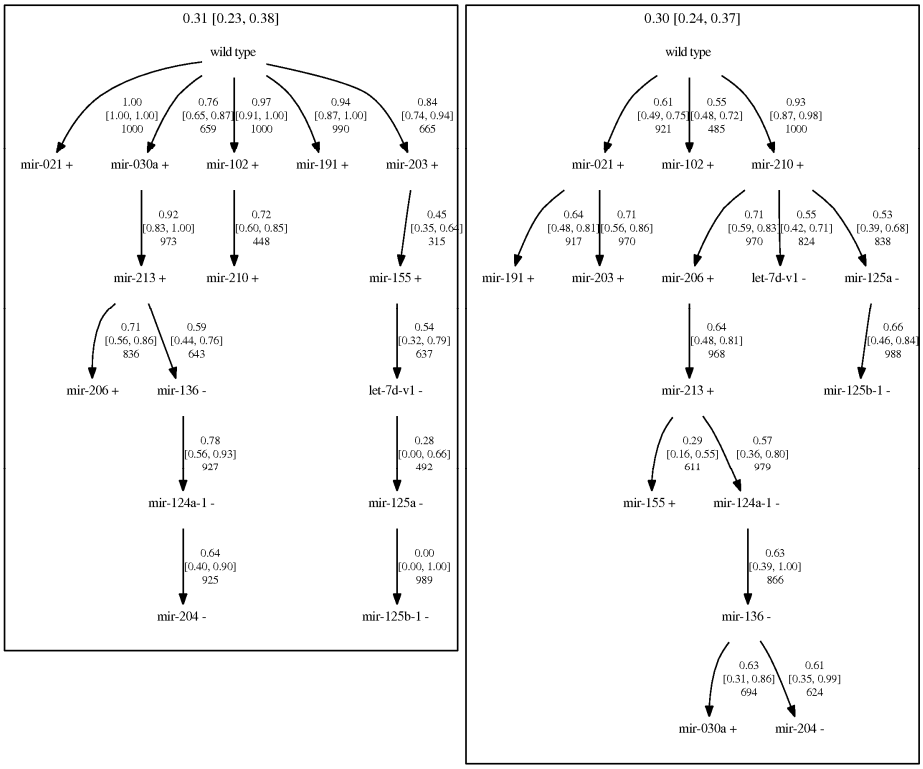


Figure 4: Oncogenetic tree mixture model of 15 events selected based on the PCA results. The star-shaped noise component is not shown. Edges are annotated with the transition probability (with confidence interval, CI) and the bootstrap samples count as a measure of stability. The weight of each mixture component (with CI) is given at the top of the box.

Also the PCA of the altered data shows much less underlying structure (Figure 5b). The events plotted in relation to the first two components are more uniformly distributed. The first three components explain only ~19% of the total variance, which is much less than in the original data set, where the first three components explained >50% of the variance.

4 Discussion

We performed several analyses to derive information about temporal and occurrence relations between miRNA deregulations. All analyses, especially the comparison to randomized data set, show that there is an underlying structure behind the used data set. This means that the observed events do not occur randomly.

There is an agreement between the derived temporal order (i.e., the results from the temporal analysis according to [Hö01][Hö05]) as well as the oncogenetic trees built according to [Be05a] and the principal component analysis. The order of the events in

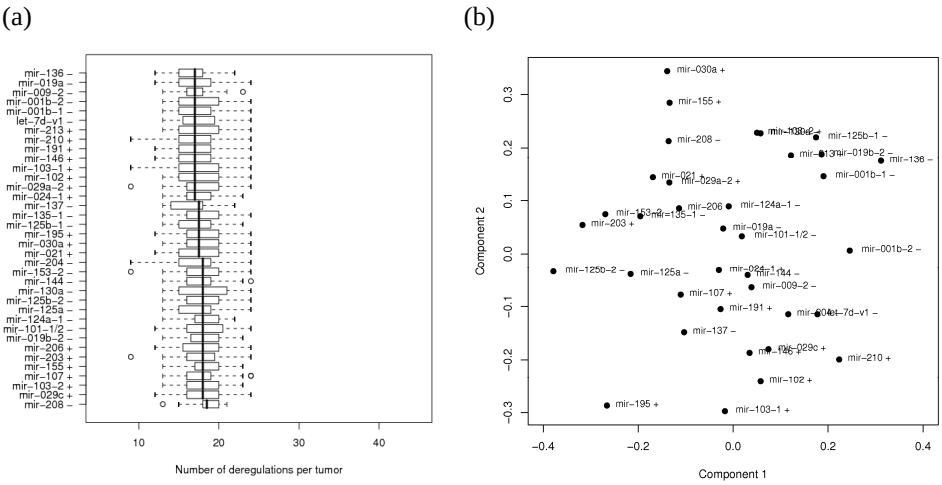


Figure 5: Results of the different analyses of the randomized data set.

relation to the second principal component correlates slightly with the observed time of occurrence. A high value of the second component indicates early occurrence, whereas a low value indicates a later occurring event. According to all three analyses, mir-21+ and mir-191+ are, among others, considered as the initial events in tumor progression. This assumption is also supported by literature [Io05][Vo06], because both miRNAs have been identified to be consistently over-expressed in cancer.

The events mir-206+, mir-210+, let-7d-v1−, mir-125a−, and mir-125b-1− are peculiar in some of the analyses (including the co-occurrence statistics, which are not shown). For instance, they are slightly separated in the PCA plots (Figure 3). These are among the very few events which show a variation in the first principal component compared to the other events. We assume that these events form a cluster, i.e., they occur very often together, but rather seldom in the presence of other deregulations. For all these miRNAs important roles in cancer have already been shown [Ad07][Ca08][Io05][Vo06].

There are also some disagreements in the results. For example, it is obvious that only small parts of the oncogenetic tree models agree with each other. Examples include edges like “wild type → mir-021+” or “wild type → mir-102+”. But many events occur in totally different places within the trees, i.e., in some as early and in some as late events, and always with different events as predecessor and successor. It is likely that the data set, although sufficiently large and informative to derive the temporal order, is not sufficiently large to derive accurate tree models. Since tree models branch out in different directions from the root node, the corresponding subsets of data become increasingly small, which rapidly leads to a lack of data for accurate modeling. Thus, it is to be expected that only the edges closest to the root will show consistency between different trees, unless a very large data set is used. The most important future work will therefore be to evaluate the tree models on larger data sets, as well as on different types of cancer.

References

[Ad07] Adams, B.D.; Furneaux, H.; White, B.A.: The micro-ribonucleic acid (miRNA) mir-206 targets the human estrogen receptor-alpha (eralpha) and represses eralpha messenger RNA and protein expression in breast cancer cell lines. Mol Endocrinol, 21(5):1132-1147, May 2007.

[Be05a] Beerenwinkel, N.; Rahnenführer, J.; Däumer, M.; Hoffmann, D.; Kaiser, R.; Selbig, J.; Lengauer, T.: Learning multiple evolutionary pathways from cross-sectional data. J Comput Biol 12(6):584-598, 2005.

[Be05b] Beerenwinkel, N.; Rahnenführer, J.; Kaiser, R.; Hoffmann, D.; Selbig, J.; Lengauer, T.: Mtreemix: a software package for learning and using mixture models of mutagenetic trees. Bioinformatics, 21(9):2106-2107, May 2005.

[Br82] Brodeur, G.M.; Tsiatis, A.A.; Williams, D.L.; Luthardt, F.W.; Green, A.A.: Statistical analysis of cytogenetic abnormalities in human cancer cells. Cancer Genet Cytogenet 7(2):137-152, 1982.

[Br03] Brazma, A.; Parkinson, H.; Sarkans, U.; Shojatalab, M.; Vilo, J.; Abeygunawardena, N.; Holloway, E.; Kapushesky, M.; Kemmeren, P.; Lara, G.G.; Oezcimen, A.; Rocca-Serra, P.; Sansone, S-A.: Arrayexpress-a public repository for microarray gene expression data at the EBI. Nucleic Acids Res 31(1):68-71, 2003.

[Ca08] Camps, C.; Buffa, F.M.; Colella, S.; Moore, J.; Sotiriou, C.; Sheldon, H.; Harris, A.L.; Gleadle, J.M.; Ragoussis, J.: hsa-mir-210 is induced by hypoxia and is an independent prognostic factor in breast cancer. Clin Cancer Res, 14(5):1340-1348, Mar 2008.

[CQB03] Causton, H.C.; Quackenbush, J.; Brazma, A.: Microarray Gene Expression Data Analysis: A Beginner's Guide. Wiley-Blackwell, 2003.

[De99] Desper, R.; Jiang, F.; Kallioniemi, O.P.; Moch, H.; Papadimitriou, C.H.; Schäffer, A.A.: Inferring tree models for oncogenesis from comparative genome hybridization data. J Comput Biol 6(1):37-51, 1999.

[Hö01] Höglund, M.; Gisselsson, D., Mandahl, N.; Johansson, B.; Mertens, F.; Mitelman, F.; Säll, T.: Multivariate analyses of genomic imbalances in solid tumors reveal distinct and converging pathways of karyotypic evolution. Genes Chromosomes Cancer 31(2):156-171, 2001.

[Hö05] Höglund, M.; Frigyesi, A.; Säll, T.; Gisselsson, D.; Mitelman, F.: Statistical behavior of complex cancer karyotypes. Genes Chromosomes Cancer 42(4):327-341, 2005.

[Io05] Iorio, M.V; Ferracin, M.; Liu, C-G.; Veronese, A.; Spizzo, R.; Sabbioni, S.; Magri, E.; Pedriali, M.; Fabbri, M.; Campiglio, M.; Ménard, S.; Palazzo, J.P.; Rosengerg, A.; Musiani, P.; Volinia, S.; Nenci, I.; Calin, G.A.; Querzoli, P.; Negrini, M.; Croce, C.M.: MicroRNA gene expression deregulation in human breast cancer. Cancer Research 65(16):7065-7070, 2005.

[Ka06] Kawada, J.I.; Kimura, H.; Kamachi, Y.; Nishikawa, K.; Taniguchi, M.; Nagaoka, K.; Kurahashi, H.; Kojima, S.; Morishima, T.: Analysis of gene-expression profiles by oligonucleotide microarray in children with influenza. J Gen Virol 87(6):1677-1683, 2006.

[Lu05] Lu, J.; Getz, G.; Miska, E.A.; Alvarez-Saavedra, E.; Lamb, J.; Peck, D.; Sweet-Cordero, A.; Elbert, B.L.; Mak, R.H.; Fernando, A.A.; Downing, J.R.; Jacks, T.; Horvitz, H.R.; Golub, T.R.: MicroRNA expression profiles classify human cancers. Nature 435(7043):834-838, 2005.

[Ra05] Rahnenführer, J.; Beerenwinkel, N.; Schulz, W.A.; Hartmann, C.; von Deimling, A.; Wullich, B.; Lengauer, T.: Estimating cancer survival and clinical outcome based on genetic tumor progression scores. Bioinformatics 21(10):2438-2446, 2005.

[Sa08] Sassen, S.; Miska, E.A.; Caldas, C.: MicroRNA-implications for cancer. Virchows Arch, 452(1):1-10, 2008.

[To02] Tomlinson, I.; Sasieni, P.; Bodmer, W.: How many mutations in a cancer? Am J Pathol:160:755–8, 2002.

[Vo06] Volinia, S.; Calin, G.A.; Liu, C-G.; Amps, S.; Cimmino, A.; Petrocca, F.; Visone, R.; Iorio, M.; Roldo, C.; Ferracin, M.; Prueitt, R.L.; Yanaihara, N.; Lanza, G.; Scarpa, A.; Vecchione, A.; Negrini, M.; Harris, C.C.; Croce, C.M.: A microRNA expression signature of human solid tumors defines cancer gene targets. Proc Natl Acad Sci U S A 103(7):2257-2261, 2006.

A Comparative Study of Robust Feature Detectors for 2D Electrophoresis Gel Image Registration

Birgit Möller, Oliver Greß and Stefan Posch

Institute of Computer Science, Martin-Luther University Halle-Wittenberg
{birgit.moeller, oliver.gress, stefan.posch}@informatik.uni-halle.de

Abstract: In this study we consider the performance of different feature detectors used as the basis for the registration of images from two-dimensional gel electrophoresis. These are three spot detectors also used to identify proteins, and two domain independent keypoint detectors. We conduct a case study with images from a publically available data set which are synthetically distorted using thin plate splines. The performance is assessed by the repeatability score, the probability of an image structure to be detected in original and distorted images with reasonable localization accuracy.

1 Introduction

Two-dimensional gel electrophoresis is a well established approach for separating proteins in cell samples, and along with mass spectrometry one of the key technologies for comparative proteomics [Spe04]. To assess protein quantification and differences from varying experimental conditions and technical or biological replicates, it is essential to account for variations and distortions between gels and resulting gel images due to the experimental procedure. To ease this analysis, and especially with increasing amount of gel data available, the automatic analysis of gel images is of large interest. Typically, a first step in this process is the registration of pairs of gel images [DDY03].

Due to the global and local characteristics of deviations between gel images non-rigid transformations have to be applied. Registration techniques can be distinguished in feature-less and feature-based approaches, where also combinations were proposed [ZF03]. The first category directly exploits the intensity information of the images [WGP08]. In contrast, the latter one first detects features in both images which are subsequently matched and used to guide the computation of a suitable transformation for registration. Results and quality of feature-based registration obviously depend on the amount, spatial distribution, and localization accuracy of features used for matching.

For registration of gel images, protein spots have typically been used as features within feature-based registration methods (e.g. [P$^+$99, R$^+$04, SK08]) as they are detected anyway to identify proteins. However, for the registration process there is no need to restrict potential types of features to spots. In this work we aim at assessing the appropriateness of five feature detectors as basis for subsequent matching and registration of gel images. Among these are three spot detectors, namely the Laplace, Ring, and Meaningful Boundaries detector. We contrast these with two keypoint detectors, SIFT and SURF, which are

widely used for various image analysis tasks, however, have not been applied to gel images. We expect our results to give guidance to select suitable feature types and detectors for robust and precise registration algorithms.

The remainder of this paper is organized as follows. After reviewing related work in Section 2 we briefly review the feature detectors evaluated. The test data and our evaluation strategy are detailed in Sec. 4, and the results are presented in Sec. 5.

2 Related Work

Over the years various techniques for automatically registering pairs of 2D electrophoresis images have been published. Regardless of whether the registration method is exclusively based on feature correspondences, or if it is a combined feature-based and feature-less technique, the quality of the registration result is directly linked to the quality of the correspondences provided. The more robust these matches are, the more uniformly distributed over the entire image area and the less outliers they contain, the better the registration will work. A large number of correct matches is especially important with regard to the domain of gel image registration, since here *non-rigid* image transformations have to be applied, requiring much more parameters than rigid ones and, thus, more correspondences for robust transformation estimation (cf. Sec. 4). In addition, common statistically robust estimators, like RANSAC, are not applicable for these transformations due to the high-dimensional parameter space and a high computational effort.

An indispensable prerequisite to determine robust correspondences for pairs of images is the detection of stable *features* in each single image, which are then matched for correspondence selection. Given the domain of gel images it is straight-forward to extract such features by explicitly detecting the most striking image patterns, i.e., protein spots. Common techniques for detecting those are, e.g., Laplacians [R$^+$04], watersheds [P$^+$99], morphological operators [CP92] or parametric spot models like 2D Gaussians [PF89]. Also more complex algorithms have been proposed, e.g., based on Markov Random Fields [Bak00]. However, spot-like structures in gel images are only one possible choice for stable features.

In various other computer vision applications, like camera motion recovery [HZ04], mosaicing [Cap04] or robot navigation [GZBSV03], also robust features for correspondence extraction are indispensable. In these scenarios usually no assumptions about specific image contents or structures can be made. Thus, for feature detection flexible keypoint detectors have been devised that yield stable features independent of a certain image domain or application context, and also under severe image deformations and degradations.

In [MS04] a thorough analysis of various general keypoint detectors is presented. Most of them are either based on image derivatives, i.e., Hessian or moment matrices. One of the most prominent ones is probably the Harris corner detector [HS88]. Alternative approaches, e.g., rely on evaluation of local image intensity patterns [SB97]. More recently a new class of scale invariant detectors, the Scale Invariant Feature Transform (SIFT) [Low04] and Speeded-Up Robust Features (SURF) [BTvG06], became popular. Compared to explicit spot and local feature detectors these have the advantage that they also detect more and other meaningful image structures. In particular, their scale invariance allows for extraction of characteristic intensity configurations on larger scales, e.g., striking intensity distributions in the images, which yields a larger flexibility in feature extraction.

3 Feature Detection for 2D Gel Image Registration

Our aim in this paper is to provide a thorough performance evaluation of common gel image specific spot detectors, and in particular, to compare those to more general keypoint detectors in the domain of 2D electrophoresis gel images. In detail, we will analyze the robustness of different techniques with regard to non-rigid image deformations and non-uniform image structure distributions as they are typical for gel images. Below, different approaches for feature detection included in our case study are discussed in more detail.

3.1 Spot Detectors

Laplace Detector One of the most simple and fast techniques for spot detection in gel images is given by Laplacian detection (e.g., [R$^+$04, RG07]). Spot centers are modeled as image locations with significant local curvature, detectable in terms of significant values in 2nd order derivatives. The spot detection itself is done by smoothing the image applying a Gaussian mask and then simply thresholding the Laplace images ∂_x^2 and ∂_y^2:

$$f(x,y) = \begin{cases} 1, & \text{if } (\partial_x^2(x,y) > t_L) \wedge (\partial_y^2(x,y) > t_L) \text{ with } t_L = 0 \\ 0, & \text{otherwise} \end{cases}$$

As detected spot locations are usually not isolated, connected components are extracted from the binary image f, and only their mass centers are kept as valid locations (Fig. 2).

Ring and Ellipse Operators The ring operator proposed in [WTN97] for spot detection is based on the assumption that spots usually show a circular or elliptical shape, with the inner parts of the ellipse being darker than the outer ones. Related structures are detected by initially smoothing the gel image with a Gaussian mask, and then applying Otsu thresholds to the original image intensity values, and also to local gradient magnitudes.

In the resulting two binary images all pixels (x, y) that show a low intensity and lie in homogeneous image regions are further analyzed. The main idea is to search for spot-specific intensity distributions given two sets of pixels, $C_{x,y}$ and $R_{x,y}$, for each (x, y):

$$\begin{aligned} C_{x,y} &= \{(u,v) | (u-x)^2 + (v-y)^2/\alpha^2 \leq r_M^2\} \\ R_{x,y} &= \{(u,v) | r_m^2 \leq (u-x)^2 + (v-y)^2/\alpha^2 \leq r_M^2\} \end{aligned}$$

$C_{x,y}$ includes all pixels lying in an elliptical region (specified by α) around pixel (x, y) with distances up to r_M to the center pixel (x, y), while $R_{x,y}$ contains only the pixels of $C_{x,y}$ with a distance of at least r_m to the center. The ring detector itself is then given by

$$h(x,y) = \min_{(u,v) \in R_{x,y}} I(u,v) - \min_{(u,v) \in C_{x,y}} I(u,v).$$

For final spot detection, h is thresholded with $t_H = 0$, connected components are labeled in the resulting binary image, and spots are extracted as the components' centers of mass.

Level Lines and Meaningful Boundaries The concept of meaningful boundaries defines a measure of meaning for closed curves based on the Helmholtz principle [A$^+$07].

The level lines of the lower gray level sets are extracted from a 2D gel image and examined for their meaning to derive meaningful level lines and by this detect spots. The meaning of a level line is determined by its length and the probability of occurence of a contrast in the image, which is larger than the minimal contrast on the level line. Meaningful level lines are reduced to one contour per spot, and the enclosed area determines the position of the spot by its center of mass (Fig. 2, right clip).

3.2 Image Content Independent Feature Detectors

If no assumptions about image contents and structures can be made, feature detectors independent of such knowledge are needed. Optimally, these are invariant against scale and transformations. Recently, two such scale invariant keypoint detectors were published, the Scale Invariant Feature Transform [Low04] and Speeded-Up Robust Features [BTvG06]. Both are quite robust against affine transformations and gained large importance due to their proven general applicability in various scenarios. In the context of our study we evaluate their robustness with regard to the domain of 2D gel images, and regarding non-rigid transformations which has not been done systematically until now.

SIFT - Scale Invariant Feature Transform The basic concept of SIFT [Low04] is a thorough analysis of image characteristics in scale space. Different scales are acquired by downsampling the input image $I(x, y)$ applying Gaussian convolution kernel functions $G_\sigma(x, y)$ of specific standard deviation σ:

$$I_\sigma(x, y) = G_\sigma(x, y) * I(x, y)$$

Combining different scales of an image into a continuous function of scale yields the image *scale space*. Between neighboring scales σ is varied by a constant factor k. Keypoints are then given by local extrema in the difference images $D_\sigma(x, y)$ between two subsequent scales:

$$D_\sigma(x, y) = I_{k\sigma}(x, y) - I_\sigma(x, y)$$

For extrema detection, the difference value $D_\sigma(x, y)$ of each point (x, y, σ) in scale space is compared to all neighbors in a $3 \times 3 \times 3$ neighborhood. By fitting a 3D quadratic function to the local point the extremum can be localized with subpixel accuracy (Fig. 2, left).

SURF - Speeded-Up Robust Features More than SIFT the SURF approach is tuned for efficiency, but the features nevertheless show a high stability [BTvG06]. The main idea is given by an analysis of local Hessian matrices $H(x, y, \sigma)$ over various scales:

$$H(x, y, \sigma) = \begin{bmatrix} L_{xx}(x, y, \sigma) & L_{xy}(x, y, \sigma) \\ L_{yx}(x, y, \sigma) & L_{yy}(x, y, \sigma) \end{bmatrix},$$

where $L_{..}$ are the results of convolving the input image with 2nd order Gaussian derivatives in xx, yy, xy and yx direction, respectively. However, for efficiency reasons the entries of the matrix are calculated only approximately applying discrete box filters as approximations to the Gaussian derivative kernels. Convolutions with box filters can quite efficiently be calculated given integral images. In addition, in contrast to usual scale space approaches, the images are not resampled within a pyramid, but detection results for various scales are produced by simply applying differently sized filters to the input image.

In SURF robust keypoints are defined by maximal determinant values of local Hessian matrices. Accordingly, detection is done by searching for maximum determinants. Initially a non-maximum suppression in a $3 \times 3 \times 3$ neighborhood of each point is performed, and maxima locations are interpolated in scale and space, like in the SIFT approach (Fig. 2).

4 Experimental Evaluation

Evaluating the efficiency of spot and keypoint detectors, respectively, is a difficult task. The main problem is usually a lack of ground truth data with known corresponding feature point locations. Accordingly, one common approach is to generate synthetically deformed images from a given reference image by applying a known transformation so that correspondences can be calculated directly (e.g. [MS04, BTvG06]). Before and after transformation features are then detected in the test images applying different detectors. To assess quality and robustness of the various detectors, meaningful quality measures are used.

Dataset For testing the various feature detection techniques we used a selection of gels from the LECB 2-D PAGE Gel Images Data Sets [LLL84], freely available for public use[1]. In detail we selected 45 images from the Human leukemias data set, each image sized 512×512 pixels in 8-bit GIF format. All images were converted to PGM format and automatically cropped given the annotated valid spot areas within the gels as specified in the complementary description files. Since not all area specifications were accurate and sometimes artefacts remained at the border of images, 10 images were manually post-processed afterwards (cropping, filling of spurious white regions with local background color) to also remove these artefacts and prevent detectors from selecting spurious features.

Categorization of Images The images of the dataset show a wide variety of complexity, ranging from bright images with very few spots to very dark images with lots of structure. To enable a thorough comparison, the gels were manually classified into 4 different complexity classes, where each class contained 7 to 15 images:

C_0 gels with only some few spots;
C_1 gels with a moderate number of spots;
C_2 gels with lots of spots;
C_3 gels that were quite dark and spot segmentation quite difficult in large areas.

Synthetic Image Deformations using Thin Plate Splines The deformation of 2D electrophoresis gel images is often modelled applying bilinear transformations [SA$^+$02] or thin plate splines [Ped02]. Since thin plate spline (TPS) transformations [Boo89] can take full advantage of the information provided by landmark points [DDY03] we choose this model for our experiments. 25 basis functions were applied to model local and global deformations. For each image, the centers for the 25 basis functions were uniformly sampled in the image domain. For each basis function a displacement vector was drawn from a Gaussian distribution with standard deviation σ_D and zero mean. The standard deviation σ_D was varied to simulate different amounts of distortion of the 2D gels. A global affine transformation was added to these displacements, which again was randomly

[1] http://www.lecb.ncifcrf.gov/2DgelDataSets/

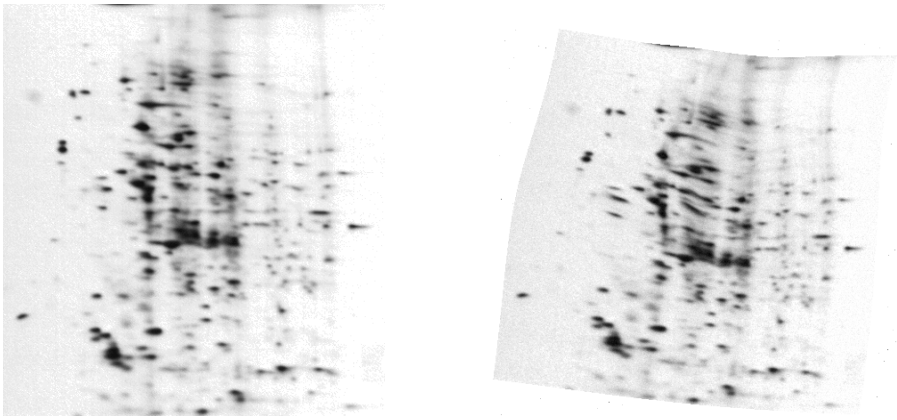


Figure 1: A sample gel from class C_2, undistorted (left) and deformed with $\sigma_D = 4$ (right).

sampled. The rotation was uniformly drawn from the interval $[-10°; 10°]$, the shearing axis uniformly from $[-90°; 90°]$ and the two scale factors uniformly from the interval $[0.9; 1.1]$. Given the resulting displacements for the centers, a TPS transformation was determined and the original image transformed accordingly. To simulate variations in the gray value structure, white noise was added to the interplolated intensities which was sampled independently from a Gaussian distribution with standard deviation 5. For each $\sigma_D \in \{1, 2, 3, 4, 5\}$ we generated 10 randomly distorted images for each original image. This results for each σ_D in an evaluation set of $45 \times 10 = 450$ distorted images. For an example of a distorted gel image see Fig. 1. All images with one distorted image for each distortion level and detected features are available as supplemental material on our server[2].

Performance Measure Comparing the robustness and efficiency of feature detectors is an important task in computer vision, and various performance measures exist. With regard to the topic of this paper we are particularly interested in the *repeatability score* Rs_r of a certain detector (cf. [MS04]). It quantifies the probability of a feature in an undistorted image I to be re-localized in a deformed version I_T of the same image with accuracy r:

$$\text{Rs}_r(I, I_T) \;\; = \frac{|P_T|}{|P_I|} \;\; \text{with} \;\; P_T = \{\, p_t \mid r > \| \, p_i - TPS(p_t) \, \| \} \tag{1}$$

P_I is the set of features p_i detected in the undistorted original gel image, and P_T is the set of features p_t detected in the transformed image, that have a distance not bigger than r to their initial counterparts in the original image after TPS transformation. For the evaluation in this work we used $r = 1.5$ pixels. This value has already proven its suitability in evaluating feature detectors for non-rigid registration, allowing for convenient registration results given the robustness and flexibility of up-to-date feature descriptors (cf. [MS04]) .

Obviously the overall number of final correspondences not only depends on the initially detected features, but also on the subsequent matching process where suitable feature descriptors have to be applied. In this work, we concentrate on robust detection of features as an indispensable prerequisite and fundamental precondition for any matching process.

[2]http://www2.informatik.uni-halle.de/agprbio/AG/Publication/OnlineMaterial/GCB_2008/Gels

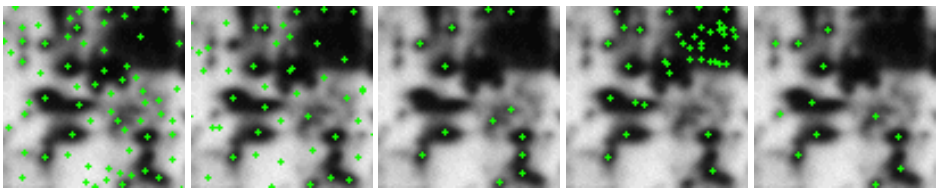


Figure 2: Prototypical detection results for SIFT, SURF, the ring operator, Laplacians and the meaningful boundaries (from left to right) for an image of class C_3. Green crosses mark spot centers.

5 Results and Discussion

Five different spot and keypoint detectors, respectively, were included in our study, i.e., Laplacians ('laplace'), the ring detector ('ring'), meaningful boundaries ('level'), SIFT ('sift'), and SURF ('surf'). For SIFT and SURF we used publicly available software packages, i.e., the free C++ implementation of SIFT by A. Vedaldi[3] and the original SURF library provided by its authors[4]. All other detectors were re-implemented by ourselves.

Each of the detectors was applied to all original images within the four complexity classes and all images within the five distortion levels. All detectors were initially run with standard parameter settings as specified in related publications. The only exception is for the Laplacian detector adopted from [R$^+$04] where the number of spots to be detected was explicitly specified manually. The authors select the 400 most intense spots from their gel images. However, for images in our experiments this number appeared too high, especially for image categories with few spots. The detector is enforced to extract spots even from more or less homogeneous background regions. Hence we chose more suitable spot numbers for each complexity class in our test dataset (Tab. 1).

Detector	C_0	C_1	C_2	C_3
Laplace	100	150	275	300
Ring	13	33	87	226
Level	45	101	153	158
SIFT	247	534	826	1100
SURF	43	148	330	573

Table 1: Avg. number of features detected with standard parameter settings for the gel images in each complexity class C_0 to C_3.

First it is noted that the number of features detected on average in undistorted images varies significantly for different detectors (see Tab. 1 and Fig. 2 for an example of detection results from a clipped section of one gel image). SIFT and SURF almost always extract significantly more keypoints than the spot detectors. The ring operator yields less than 100 spots for classes C_0 to C_2, which is well below the counts for the other spot detectors. As a large number of correspondences and, thus, features is required for precise registration, the keypoint detectors show superior compared to the spot detectors regarding this aspect.

Of course, the total number of features detected is not sufficient for high detector quality. As important is the repeatability score of the detector, i.e., the number of initially detected features that are expected to be re-localized in deformed and degraded images. The repeatability score as defined in Equ. (1) in Sec. 4 relates the number of re-detected features to the number of features detected initially. Accordingly, it is normalized with regard to the

[3] http://vision.ucla.edu/~vedaldi/code/siftpp/siftpp.html

[4] http://www.vision.ee.ethz.ch/~surf/

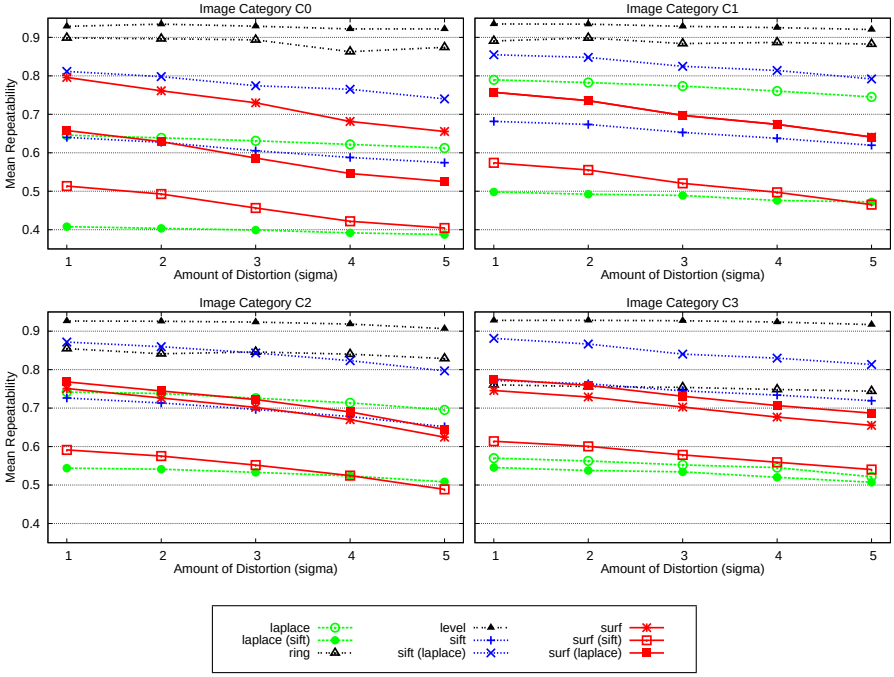


Figure 3: Average repeatability scores per class for various detectors applied to all four complexity classes for varying distortion levels σ.

total number of features. As a consequence, the repeatability scores achieved for different detectors are only comparable if approximately the same number of features was detected. To this end we have performed additional evaluation runs where the parameters for SIFT, SURF and Laplacian detectors were adjusted to yield approximately the same number of features. The ring detector and meaningful boundaries were not included since there is no reasonable way to adapt them for detecting comparable numbers of features.

The results of our experiments are summarized in Fig. 3. For each class the mean repeatability scores for each detector are plotted, calculated by averaging the detection results for all images of a class with a given distortion level. The graphs 'sift', 'surf', 'laplace', 'ring' and 'level' give results for experiments with standard parameter settings. For SIFT, SURF and Laplace there are additional curves in each plot, related to non-standard parameter settings. For 'sift(laplace)' and 'surf(laplace)' both detectors were adjusted to detect the same number of keypoints as the Laplace does for its standard settings. Likewise 'laplace(sift)' and 'surf(sift)' give the results for both detectors parameterized to yield the same high number of features that SIFT detects with default settings (cf. Tab. 1 for exact numbers).

For standard parameters, the meaningful boundaries give a repeatability of about 90% for all configurations, the ring operator yields also about 90% for categories C_0 and C_1, which drops to about 75% for C_3. SIFT and SURF show performances in the range of about 60% to 80%. The standard repeatability score of Laplace for categories C_0 to C_2 is comparable to the ones of SIFT and SURF, and drops to $55 - 50\%$ for category C_3.

Adjusting SURF to detect the same number of feature points as SIFT, which results approximately in doubling the number of keypoints, reduces its performance significantly to about $45 - 65\%$. Accordingly, considering both the total number of detected features and the repeatability score, SIFT appears to have advantages over SURF, independent of the image category. For the Laplace detector, the repeatability goes down to about $40 - 55\%$, particularly for images with little structure as in categories C_0 and C_1. This is not surprising, but underlines the superiority of standard SIFT in these classes. It becomes obvious that the Laplace detector is by no means suitable for detecting large numbers of features.

Restricting the feature number of SIFT to the smaller number of the Laplace detector yields significant improvements of the repeatability which increases by $\approx 15 - 20\%$ in each category. In contrast, if SURF is restricted to the same number of features its repeatability remains more or less unchanged except for category C_0, where it declines significantly[5].

In general, our evaluation results show that the image category has little influence on the repeatability scores of the various detectors, but mainly yields significant differences in the total number of detected feature points. Increasing the amount of distortion has also little influence for the meaningful boundaries and ring detector, but decreases the performance for the others of about 10%. If only the repeatability is considered, the meaningful boundaries and ring detector yield the highest scores and the largest robustness. Contrary, if a large number of robust features is required general scale invariant detectors, in particular SIFT, appear favorable compared to explicit spot detectors. Given their repeatability scores they form a suitable foundation for extracting a large number of robust feature correspondences essential for high-quality feature-based gel image registration.

6 Conclusion

In current approaches for feature-based registration of gel images, correspondences are almost always based on protein spots as domain inherent features. The first contribution of this paper is a novel systematic quantitative analysis of various commonly used spot detectors. It allows for an objective evaluation of the detectors with regard to stability and repeatability. Secondly, we propose the application of more general keypoint detectors for feature extraction, i.e., SIFT and SURF. Compared to explicit spot detectors a significantly larger number of features per image is extracted on average with a likewise higher repeatability. Since large numbers of stable features yield an important basis for robust correspondence detection and also high-quality image registration, SIFT and SURF show advantages over conventional techniques and should no longer be ignored in this field.

References

[A+07] A. Almansa et al. Processing of 2D Electrophoresis Gels. In *1st Int. Works. on Comp. Vision Appl. for Developing Regions (ICCV)*, 2007.

[Bak00] An automatic registration and segmentation algorithm for multiple electrophoresis images. In *Medical Imaging*, pages 426–436, 2000.

[5] For image category C_1 the graphs 'surf' and 'surf(laplace)' are identical, as in this category SURF and Laplace detect the same number of features with standard settings (see Tab. 1).

[Boo89] F. L. Bookstein. Principal Warps: Thin-Plate Splines and the Decomposition of Deformations. *IEEE Trans. Pattern Anal. Mach. Intell.*, 11(6):567–585, 1989.

[BTvG06] H. Bay, T. Tuytelaars, and L. van Gool. SURF: Speeded Up Robust Features. In *Proc. of European Conference on Computer Vision*, pages I: 404–417, 2006.

[Cap04] D. Capel. *Image Mosaicing and Super-resolution*. Springer, 2004.

[CP92] K. Conradsen and J. Pedersen. Analysis of two-dimensional electrophoresis gels. *Biometrics*, 48:1273–1287, 1992.

[DDY03] A.W. Dowsey, M.J. Dunn, and G.-Z. Yang. The role of bioinformatics in two-dimensional gel electrophoresis. *PROTEOMICS*, 3(8):1567–1596, 2003.

[GZBSV03] N. Gracias, S. Zwaan, A. Bernardino, and J. Santos-Victor. Mosaic-based Navigation for Autonomous Underwater Vehicles. *IEEE J. of Oceanic Engineering*, 28(4), 2003.

[HS88] C.J. Harris and M. Stephens. A Combined Corner and Edge Detector. In *Proc. of Alvey Vision Conference*, pages 147–151, Manchester, England, 1988.

[HZ04] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2nd edition, 2004.

[LLL84] E.P. Lester, P.F. Lemkin, and L.E. Lipkin. Protein indexing in leukemias and lymphomas. *Ann N Y Acad Sci.*, 428:158–72, 1984.

[Low04] David G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *Int. Journal of Comp. Vision*, 60(2):91–110, 2004.

[MS04] K. Mikolajczyk and C. Schmid. Scale & Affine Invariant Interest Point Detectors. *Int. J. Comput. Vision*, 60(1):63–86, 2004.

[P+99] K.-P. Pleißner et al. New algorithmic approaches to protein spot detection and pattern matching in two-dimensional electrophoresis gel databases. *Electrophoresis*, 20:755–765, 1999.

[Ped02] L. Pedersen. *Analysis of Two-Dimensional Electrophoresis Gel Images*. PhD thesis, Informatics and Mathematical Modelling, Technical University of Denmark, 2002.

[PF89] P.F. Lemkin PF. GELLAB-II, A workstation based 2D electrophoresis gel analysis system. In *Proc. of 2D Electrophoresis*, pages 52–57. VCH Press, 1989.

[R+04] M. Rogers et al. 2D Electrophoresis Gel Registration Using Point Matching and Local Image-Based Refinement. In *Proc. of BMVC*, Kingston, UK, 2004.

[RG07] M. Rogers and M. Graham. Robust and Accurate Registration of 2D Electrophoresis Gels using Point-Matching. *IEEE Trans. on Image Processing*, 16(3):624–635, 2007.

[SA+02] J. Salmi, T. Aillokallio, et al. Hierarchical grid transformation for image warping in the analysis of two-dimensional electrophoresis gels. *Proteomics*, (2):1504–1515, 2002.

[SB97] S.M. Smith and J.M. Brady. SUSAN - A New Approach to Low Level Image Processing. *Int. Journal of Comp. Vision*, 23(1):45–78, 1997.

[SK08] T. Srinark and C. Kambhamettu. An image analysis suite for spot detection and spot matching in two-dimensional electrophoresis gels. *Electroph.*, 29(3):706–715, 2008.

[Spe04] D.W. Speicher. *Proteome Analysis - Interpreting the Genome*. Elsevier, 2004.

[WGP08] J. Wensch, A. Gerisch, and S. Posch. Optimised coupling of hierarchies in image registration. *Image and Vision Computing*, 26(7):1000–1011, 2008.

[WTN97] Y. Watanabe, K. Takahashi, and M. Nakazawa. Automated Detection and Matching of Spots in Autoradiogram Images of Two-Dimensional Electrophoresis for High-speed Genome Scanning. In *Proc. of Int. Conf. on Image Proc.*, pages (3):496–499, 1997.

[ZF03] B. Zitová and J. Flusser. Image registration methods: a survey. *Image and Vision Computing*, 21(11):977–1000, 2003.

Small Molecules as Rotamers: Generation and Docking in ROSETTALIGAND

Kristian W. Kaufmann, Jens Meiler

Department of Chemistry
Vanderbilt University
465 21st Ave. South
Nashville, TN 37209
jens.meiler@vanderbilt.edu

Abstract: We introduce small molecule rotamers into the rotamer search protocol used in Rosetta to model small molecule flexibility in docking. Rosetta, a premier protein modeling suite, models side chain flexibility using discrete conformations observed in the Protein Data Bank (PDB). We mimic this concept and build small molecule rotamers based on conformations from the Cambridge Structural Database. We evaluate to the small molecule rotamer generation protocol on a test set of 628 conformation,s taken from the PDBBind database, of small molecules with ≤ 6 rotable bonds. Our protocol generates ensembles in which the closest conformation is 0.45 $\pm 0.31 \AA$ RMSD from the crystallized conformation. Further, in a set of 21 small molecule protein complexes, 16 of 21 cases a native-like model was in the top 1 % of models by energy.

1 Introduction

Representing protein flexibility through side chain rotamers[DK93] (discretized conformations observed in the Protein Databank) has been central to the success of protein structure prediction, protein docking, protein design. Full atom contacts, modeled using rotamers, is critical to the success of the ROSETTA program in the *de novo* prediction of protein structure[BMB05]. Furthermore, rotamers form critical components of successful protein docking and protein design strategies such as ROSETTADESIGN[KDI$^+$03][KOK$^+$01][DKC$^+$03] and ROSETTADOCK[GMW$^+$03][SFWB05]. Finally, Rosetta incorporates the rotamer probability when performing alanine scanning mutagenesis to identify important residues for protein-protein binding[KKB04]. The above success of rotamers for protein side chain flexibility makes adapting the concept for small molecule flexibility attractive.

Leach first introduced modeling small molecule flexibility in docking using rotamers[Lea94]. He took small molecule conformations in local minima of molecular dynamics forcefield as small molecule rotamers. However Leach observed a failure of the scoring function in his protocol. We independently implemented a method similar to that implemented by Leach with rigid ligands and full sidechain flexibility in the ROSETTA [MB06] protein modeling suite. The ROSETTALIGAND energy function identified native conformations

for 71 of 100 small molecule protein complexes in a self docking test and 14 of 20 small molecule protein complexes in a cross docking benchmark. In the cross docking benchmark, a small conformational ensemble containing 10 conformations, one of which was close to the crystallized conformation, was used to simulate small molecule flexibility. Here our objective is to simulate small molecule flexibility using small molecule rotamer ensemble generated from crystal structures, thus capitalizing of the knowledge base responsible for the success of ROSETTA.

In an analogous manner to the amino acid side chain rotamers, we employ small molecule crystal structures from the Cambridge Structural Database (CSD)[All02] to construct small molecule rotamers. Unlike amino acids side chains in the PDB, in the case of small molecules we lack multiple conformations of the same configurational chemistry. Instead, torsion profiles are created from chemical similar groups. Omega, a highly regarded program for generating small molecule conformations, makes use of profiles extracted from the CSD. Omega generates conformational ensembles from overlapping fragments in a rule based manner using torsion profiles[BGG03]. Perola and Charifson, in a study crystallized bioactive small molecules, found Omega to be the best available tool for generating ensembles containing the bioactive conformation.

Our objective in the following is to show the concept of rotamers in protein structure prediction can be extended to small molecules. We show that small molecule rotamers can be created using crystal structure data. In addition, these small molecule rotamer ensemble contain conformations close to bioactive conformations for molecules with torsions similar to those in protein side chains. We show that these rotamer ensembles are successful two small molecule docking benchamrks.

2 Methods

2.1 Creating Torsion Profiles from the Cambridge Structural Database

We use 28 atom types defined by element, hybridization, and number of bonded hydrogens described previously[MMWM02]. We examine all heavy atom torsions for each atom type pairing are measured, excluding torsions in ring systems, for all structures in the Cambridge Structural Database(CSD)[All02]. Each torsion is placed in a 10° bin of a histogram. Histograms are constructed for every pair of the 37 atom types. Histograms with less then 100 data points are excluded as containing to little information. The symmetric distributions are constructed from the remaining histograms by summing counts of symmetry equivalent bins.
A knowledge based energy is calculated using the inverse Boltzmann equation 1

$$E = -ln(Propensity_{torsion}) \tag{1}$$

where $P_{torsion}$ is the propensity of torsion. To generate the propensity, first a pseudo count of 1 is added to each of the bins. Next, each count is normalized by the total number of counts. Finally, the propensity is the normalized count divided by the random probability

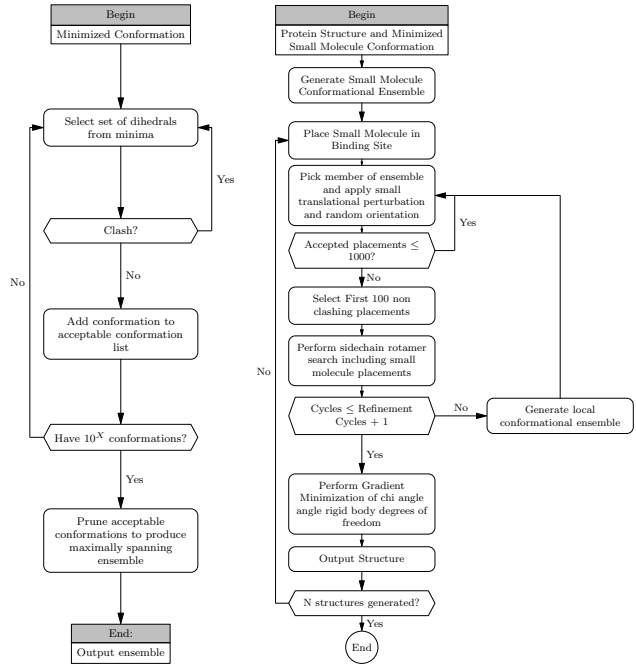
of selecting that torsion bin i.e. 1 over the number of bins. The discrete energy profile is the fit using cubic splines to generate a smooth differentiable periodic function as described in "Numerical Recipes in C++"[Pre02]. The minima in the energy profile define the states sampled while generating the small molecule rotamer ensemble.

2.2 Small Molecule Rotamer Ensemble Generation

The small molecule ensemble generation protocol (see Figure 1a) creates a maximal spanning ensemble of acceptable energy rotamers as measured by root mean squared deviation (RMSD). Starting from a conformation with idealized bond lengths and angles, a set of dihedral angles is chosen from the minima of the appropriate torsion profiles. Rotamers containing overlapping atoms are discarded. If the energy is acceptable then the rotamer is provisionally accepted. Otherwise, a new set of dihedral angles are chosen. After a user defined number of rotamers (N=10000) have been obtained, the rotamers are pruned. The pruning first selects the lowest energy rotamer for the ensemble. The energy incorporates van der Waals interaction for atoms separated by 4 or more bonds, the knowledge based torsion energy described in the previous section, an intra-molecular hydrogen bonding term, a desolvation energy based on the Lazaridis-Karplus approximation , and a coulomb electrostatics term. Next the protocol iteratively adds the furthest conformation to the members of the ensemble, as measured by best fit RMSD. The protocol continues until the desired number of rotamer has been reached (default=500) or all rotamer potential rotamers are within a user defined cutoff(cutoff=0.2 $\mathring{A}$ RMSD) .

2.3 Flexible Small Molecule Docking

Given a protein structure and small molecule conformation the protocol (see Figure 1b) first generates a conformational ensemble for the small molecule. Next a position in the binding site is chosen. A conformation is chosen from the ensemble and placed at the selected spot. A small random translation (0 ± 0.2 $\mathring{A}$) and random rotation (random angle on a sphere) is applied to the conformation. The placement is added to a list. This is repeated until 1000 placements have been generated. Placements are then evaluate to see if they clash with the backbone of the protein. The first 100 non-clashing placements are incorporated into the protein side-chain rotamer search. After the rotamer search a local conformational ensemble is created by allowing small changes of $\leq 5^{\circ}$ to the rotable bonds. This ensemble then takes the place of the general ensemble in the packing cycles for a refinement search. After the 4 refinement rotamer packing cycles, a gradient minimization of the side chain chi angles and rigid body degrees of freedom take the structure to a local minimum. This structure is then written out. The sequence is repeated until N (N=3000) structures have been generated.



(a) Ensemble Generation Protocol

(b) Docking Protocol

Figure 1: Flexible Small Molecule Docking Protocol

2.4 Small Molecule Flexibility Benchmark Sets

Compounds for the ensemble generation test set were taken from the 2007 PDBBind database [WFLW04]. All molecules with ≤ 6 rotable heavy atom torsions were selected. The small molecule files in Tripos Sybyl mol2 format were converted into ROSETTA compatible MDL mol format using in house python scripts.

Two docking benchmarks were carried out. The self docking benchmark evaluates the ability for our protocol to recover the correct conformation and orientation of a small molecule in protein crystal structure solved with that small molecule. The structures used in the self docking benchmark are listed in Table 1. The cross docking benchmark makes use of two crystal structures of the same protein. The cross docking benchmark assess the capacity of the protocol to recover the placement of a small molecule in the first protein crystal structure in context of the second protein crystal structure. Changes in the protein conformation of the second crystal structure simulation a real world situation more closely. The structures used are listed in Table 1 All structures in both docking benchmarks were previously evaluated by Meiler and Baker [MB06] for the rigid small molecule case. The set was reduced to contain only small molecules with ≤ 6 rotable heavy atom torsions.

Table 1: PDB IDs of Structures Used in Docking Benchmarks

Self Docking Structure Structure	Cross Docking ligand/protein	number of torsions
1aq1	1aq1/1dm2	1
1dm2	1dm2/1aq1	0
1dbj	1dbj/2dbl	0
2dbl	2dbl/1dbj	6
1pph	1pph/1ppc	5
1p8d	1p8d/1pqc	4
	1p8d/1pq6	
2ctc	2ctc/7cpa	3
2prg	2prg/1fm9	5
4tim	4tim/6tim	4
6tim	6tim/4tim	4

3 Results and Discussion

3.1 Torsion Profile

The torsion profiles generated cover 103 common bond types (see supplement). The profiles obtained show similar chacteristics to profiles in the AMBER[WWC$^+$04] forcefield (see Figure 2). However, some profiles exhibit minima not present in the AMBER forcefield. The aryl oxygen profile, Shown in Figure 2d, displays additional minima at $\pm$ 90°. Klebe and Meitzner found that these additional minima arise from meta substituted compounds[KM94]. The additional minima give the CSD torsion profiles an advantage, since they allow the ensemble generator to sample conformations that might otherwise be excluded.

3.2 Small Molecule Rotamer Ensemble Generation

The ensemble generator created ensembles for small molecules with $\leq$ 6 heavy atom torsion taken from 628 crystal structures. Each ensemble contained upto 500 conformations. No conformation was allowed to be closer 0.2 $\mathring{A}$ RMSD. Ten thousand conformations were generated while constructing the ensemble. On the set 628 molecules, the ensemble generator produced a rotamer with 0.46 $\pm$ 0.31 $\mathring{A}$ RMSD to the crystalized conformation. As expected, the accuracy decreases from 0.14 $\pm$ 0.16 $\mathring{A}$ RMSD to 0.79 $\pm$ 0.32 $\mathring{A}$ RMSD as the number of rotable heavy atom torsions increases from 1 to 6 (see Table 2). Improvement of these numbers might be possible by increasing the size of the ensemble, and increasing the number of rotamers generated during construction of the ensemble. The additional cost of such increases may outway the benefits.

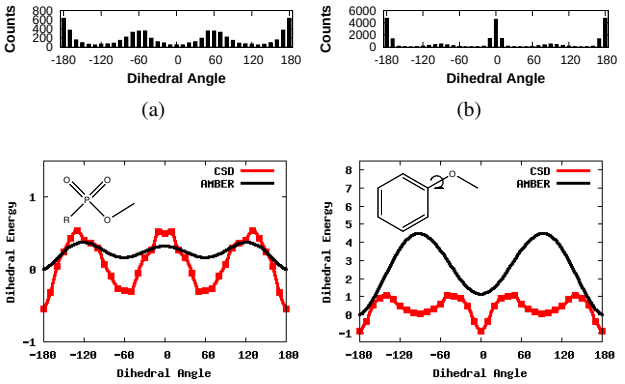


Figure 2: Torsion profiles for (a) a Phosphate Ether Oxygen Bond and (b) an Aromatic Carbon Oxygen Bond

3.3 Flexible Small Molecule Docking

The small molecule docking results are summarized in Table 3. For the self docking, 9 of the 10 cases show a native-like model in the top 1 % by energy. In 7 of the 10 cases the top ranked model is native-like. For the cross docking benchmark 7 of 11 cases show a native-like structure in the top 1% by energy. In only 2 of the 11 cases was the top ranked model a native-like model. In Figure 3a the RMSD score plot demonstrates that the scoring function identifies the native binding mode (see Figure 3c) as the most favorable. However, in other cases the RMSD score plots appear like that of Figure 3b. Some models are present in the native binding mode (see Figure 3d), but low score does not imply low

Table 2: Performance of Ensemble Generator

Number of Rotable Bonds	Number of Molecules	Ave. $\pm$ Std. Dev RMSD of closest rotamer
1	92	0.14 ± 0.16
2	118	0.33 ± 0.26
3	118	0.41 ± 0.22
4	135	0.47 ± 0.21
5	97	0.61 ± 0.30
6	118	0.79 ± 0.32
Overall Total	628	0.46 ± 0.31

Table 3: Performance on Small Molecule Docking Test Set. Cases where top ranked structure is within 2.00 $\mathring{A}$ RMSD shown in **bold**. Cases where structure is within 2.00 $\mathring{A}$ RMSD and top 1 % by energy shown in *italics*

Self Structure	rank	RMSD
Cross Docking Structure	rank	RMSD

1AQ1	**1**	**0.25**
1DM2	4296	1.87

1P8D	**1**	**1.63**
1PQ6	181	1.62
1PQC	*10*	*1.28*

1DM2	**1**	**0.31**
1AQ1	**1**	**0.56**

2CTC	*3*	*0.82*
7CPA	*3*	*0.95*

1DBJ	**1**	**1.36**
2DBL	**1**	**1.80**

2DBL	**1**	**1.450**
1DBJ	468	3.49

1PPH	*6*	*1.49*
1PPC	*2*	*1.96*

4TIM	**1**	**1.87**
6TIM	*2*	*1.90*

6TIM	**1**	**1.77**
4TIM	*5*	*1.77*

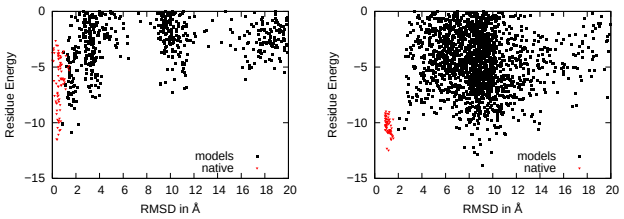
2PRG	639	1.94
1FM9	16	2.02

RMSD.

The self docking results are comparable to those in Meiler and Baker [MB06]. Meiler and Baker achieved a 71% success rate in in a self docking benchmark of 100 ligands. We see the same success rate on our reduced set despite the increased conformational sampling. However in the cross docking benchmark our results fall short. One possible cause is the increased small molecule conformational sampling in the current protocol. The previous evaluation used an ensemble size of ten in which one conformation was close the crystallized conformation. Here, we create unbiased ensembles with up to 500 rotamers. The increase in conformational diversity represents an increased challenge to the search process as well as the scoring function.

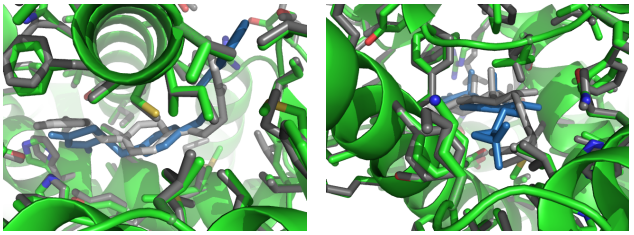
4 Conclusion

We have extended of amino acid concept of rotamers to include small molecules. When the number of torsions is in the same range as those seen in amino acids small molecule rotamer ensembles contain conformations close to those seen in crystal structures of protein small molecule complexes. Rotamer ensembles can simulate flexibility for small molecules. However, as the number of rotamers grow (due to increased flexibility) and the precision of the protein structures decrease (due to inaccuracy in protein backbone), the discriminatory power of the scoring function decreases. The components of the scoring function have not been optimized for the increased flexibility; doing so may yield increased discrimination. Improved fine grain sampling of protein backbone motion may also assist in the docking process. Additionally, the method must be extended to larger small molecules. We intend on expanding our method by breaking small molecules into multiple residues. The residues would then be reassembled in the protein binding site to form the small molecule. Thereby, we decrease the conformational complexity and incorporate information from the protein environment.



(a) 1P8D ligand docked into 1P8D structure

(b) 2PRG ligand docked into 1FM9 structure



(c) 1P8D ligand docked into 1P8D structure

(d) 2PRG ligand docked into 1FM9 structure

Figure 3: RMSD score funnels show successful discrimination of binding funnel in (a), and failure to define a singular binding funnel in (b). (c) shows the best scoring model overlaid on the crystal structure in grey. (d) shows the best scoring model below 2 $\mathring{A}$ RMSD overlaid on the 2PRG crystal structure in grey

5 Acknowledgements

K.W.K. would like to thank Robert Guenther for discussion on the topic. K.W.K was support by the Molecular Biophysics Training Grant and by the DARPA Protein Design Processes Grant.

References

[All02] F. H. Allen. The Cambridge Structural Database: a quarter of a million crystal structures and rising. *Acta Crystallogr B*, 58(Pt 3 Pt 1):380–8, 2002. 0108-7681 (Print) Journal Article.

[BGG03] J. Bostrom, J. R. Greenwood, and J. Gottfries. Assessing the performance of OMEGA with respect to retrieving bioactive conformations. *J Mol Graph Model*, 21(5):449–62, 2003. 1093-3263 (Print) Evaluation Studies Journal Article.

[BMB05] P. Bradley, K. M. Misura, and D. Baker. Toward high-resolution de novo structure prediction for small proteins. *Science*, 309(5742):1868–71, 2005. 1095-9203 Journal Article.

[DK93] R. L. Dunbrack and M. Karplus. Backbone-Dependent Rotamer Library for Proteins - Application to Side-Chain Prediction. *Journal of Molecular Biology*, 230(2):543–574, 1993. Kw260 Times Cited:335 Cited References Count:30.

[DKC+03] G. Dantas, B. Kuhlman, D. Callender, M. Wong, and D. Baker. A large scale test of computational protein design: Folding and stability of nine completely redesigned globular proteins. *Journal of Molecular Biology*, 332(2):449–460, 2003. 721FQ Times Cited:25 Cited References Count:35.

[GMW+03] J. J. Gray, S. Moughon, C. Wang, O. Schueler-Furman, B. Kuhlman, C. A. Rohl, and D. Baker. Protein-protein docking with simultaneous optimization of rigid-body displacement and side-chain conformations. *J Mol Biol*, 331(1):281–99, 2003. 0022-2836 (Print) Journal Article Research Support, Non-U.S. Gov't Research Support, U.S. Gov't, P.H.S.

[KDI+03] B. Kuhlman, G. Dantas, G. C. Ireton, G. Varani, B. L. Stoddard, and D. Baker. Design of a novel globular protein fold with atomic-level accuracy. *Science*, 302(5649):1364–1368, 2003. 745HP Times Cited:77 Cited References Count:36.

[KKB04] T. Kortemme, D. E. Kim, and D. Baker. Computational alanine scanning of protein-protein interfaces. *Sci STKE*, 2004(219):pl2, 2004. 1525-8882 (Electronic) Journal Article.

[KM94] G. Klebe and T. Mietzner. A fast and efficient method to generate biologically relevant conformations. *J Comput Aided Mol Des*, 8(5):583–606, 1994. 0920-654X (Print) Journal Article.

[KOK+01] B. Kuhlman, J. W. O'Neill, D. E. Kim, K. Y. J. Zhang, and D. Baker. Conversion of monomeric protein L to an obligate dimer by computational protein design. *Proceedings of the National Academy of Sciences of the United States of America*, 98(19):10687–10691, 2001. 472CZ Times Cited:22 Cited References Count:27.

[Lea94] A. R. Leach. Ligand docking to proteins with discrete side-chain flexibility. *J Mol Biol*, 235(1):345–56, 1994. 0022-2836 (Print) Journal Article Research Support, Non-U.S. Gov't.

[MB06] J. Meiler and D. Baker. ROSETTALIGAND: protein-small molecule docking with full side-chain flexibility. *Proteins*, 65(3):538–48, 2006. 1097-0134 (Electronic) Comparative Study Journal Article Research Support, Non-U.S. Gov't.

[MMWM02] J. Meiler, W. Maier, M. Will, and R. Meusinger. Using neural networks for (13)c NMR chemical shift prediction-comparison with traditional methods. *J Magn Reson*, 157(2):242–52, 2002. 1090-7807 (Print) Journal Article.

[Pre02] William H. Press. *Numerical recipes in C++ : the art of scientific computing*. Cambridge University Press, New York, 2nd edition, 2002. William H. Press ... [et al.]. ill. ; 26 cm. 1. Preliminaries – 2. Solution of Linear Algebraic Equations – 3. Interpolation and Extrapolation – 4. Integration of Functions – 5. Evaluation of Functions – 6. Special Functions – 7. Random Numbers – 8. Sorting – 9. Root Finding and Nonlinear Sets of Equations – 10. Minimization or Maximization of Functions – 11. Eigensystems – 12. Fast Fourier Transform – 13. Fourier and Spectral Applications – 14. Statistical Description of Data – 15. Modeling of Data – 16. Integration of Ordinary Differential Equations – 17. Two Point Boundary Value Problems – 18. Integral Equations and Inverse Theory – 19. Partial Differential Equations – 20. Less-Numerical Algorithms – App. A. Table of Function Declarations – App. B. Utility Routines and Classes – App. C. Converting to Single Precision.

[SFWB05] O. Schueler-Furman, C. Wang, and D. Baker. Progress in protein-protein docking: Atomic resolution predictions in the CAPRI experiment using RosettaDock with an improved treatment of side-chain flexibility. *Proteins-Structure Function and Bioinformatics*, 60(2):187–194, 2005. 941YB Times Cited:1 Cited References Count:34.

[WFLW04] R. Wang, X. Fang, Y. Lu, and S. Wang. The PDBbind database: collection of binding affinities for protein-ligand complexes with known three-dimensional structures. *J Med Chem*, 47(12):2977–80, 2004. 0022-2623 (Print) Journal Article Research Support, Non-U.S. Gov't.

[WWC+04] J. Wang, R. M. Wolf, J. W. Caldwell, P. A. Kollman, and D. A. Case. Development and testing of a general amber force field. *J Comput Chem*, 25(9):1157–74, 2004. 0192-8651 (Print) Journal Article Research Support, U.S. Gov't, P.H.S.

KIRMES: Kernel-based Identification of Regulatory Modules in Euchromatic Sequences

Sebastian J. Schultheiss,[1,2] Wolfgang Busch,[2] Jan U. Lohmann,[2]
Oliver Kohlbacher,[3] and Gunnar Rätsch[1]

[1]Friedrich Miescher Laboratory of the Max Planck Society, Spemannstr. 39, 72076 Tübingen, Germany

[2]Max Planck Institute for Developmental Biology, Spemannstr. 35, 72076 Tübingen, Germany

[3]University of Tübingen, Wilhelm Schickard Institute for Computer Science,
Division for Simulation of Biological Systems, Sand 14, 72076 Tübingen, Germany

Abstract:
Motivation: Understanding transcriptional regulation is one of the main challenges in computational biology. An important problem is the identification of transcription factor binding sites in promoter regions of potential transcription factor target genes. It is typically approached by position weight matrix-based motif identification algorithms using Gibbs sampling or heuristics for extending seed oligos. Such algorithms succeed in identifying single, relatively well conserved binding sites, but tend to fail when it comes to the identification of combinations of several degenerate binding sites as those often found in *cis*-regulatory modules.
Results: We propose a new algorithm that combines the benefits of existing motif finding with the ones of Support Vector Machines (SVMs) to find degenerate motifs in order to improve the modeling of regulatory modules. In experiments on microarray data from *Arabidopsis thaliana* we were able to show that the newly developed strategy significantly improves the recognition of transcription factor targets.
Availability: The PYTHON source code (open source–licensed under GPL), the data for the experiments and a web-service are available at `http://www.fml.mpg.de/raetsch/projects/kirmes`.
Contact: sebi@tuebingen.mpg.de

1 Introduction

One of the most important problems in understanding transcriptional regulation is the prediction of transcription factor target genes based on their promoter sequence. A transcription factor binding site (TFBS) is a short sequence segment ($\approx$10 bp) located near a gene's transcription start site (TSS) and is recognized by respective transcription factors (TFs) for gene regulation [GL05]. TFBSs recognized by the same TF usually show a conserved pattern, which is often called a TF binding motif (TFBM) [GL05]. Such TFBMs are typically identified by considering overrepresented motifs in promoter sequences of a set of genes that is enriched with targets for a specific transcription factor. The simplest approaches include the identification of overrepresented oligomers relative to a background model [BE94]. More sophisticated models include Gibbs-sampling methods [LAB$^+$93] that try to identify position weight matrices[SSGE86] characterizing binding sites in the candidate promoter sequences [Sto00].

Although these methods have been very successful for bacterial and yeast genomes, their success was limited in higher eukaryotes for which TFBMs are often degenerate and the search space is considerably larger. While some recent techniques have improved the

state-of-the-art, they all tend to fail if the motif is defined only weakly or in the context of other motifs. "Despite these challenges, there are two possible redeeming factors: (i) many eukaryotic genomes have been or are being sequenced, and comparative genomic analysis can be extremely powerful; and (ii) most eukaryotic genes are controlled by a combination of factors with the corresponding binding sites forming homotypic or heterotypic clusters known as '*cis*-regulatory modules' (CRMs)" [GL05].

In this work we developed novel methods that are able to classify genes as being either TF targets or not, based on the presence of motifs and features capable of describing CRMs. This is done by a two-step procedure. We first used *de novo* motif finding tools or known motif databases like TRANSFAC [MFG+03] or JASPAR [SAE+04] to identify a set of potential motifs. Then we used SVMs employing a newly developed kernel that is capable of capturing information about the motifs and their relative location to classify promoter sequences. Additionally, we demonstrate the potential of our approach to exploit conservation information to improve the classification performance.

Most previous approaches for discovering CRMs are based on the identification of motifs and their co-occurrences (*e.g.* [FSKB08, ST02]). Other approaches exploit site-clustering information with *de novo* motif discovery to build rules discriminating modules that preserve the ordering of motifs (*e.g.* [SS05]). Finally, [YTI+98] suggested to use Hidden Markov Models to represent CRMs and [GL05] developed a Monte Carlo method and dynamic programming approach to screen motif candidates. The main difference between our approach and most previous approaches is that we use discriminative methods that allow us to model the TFBS' more accurately. In particular, instead of using zeroth-order inhomogeneous Markov chains, we use Support Vector kernels to model higher order sequence information around putative TF binding sites.

The paper is organised as follows: We start Section 2.1 by describing the basic methodology of classifying sequences with Support Vector Machines using standard sequence kernels. It is followed by a detailed explanation of the main idea of this work in Section 2.2 for combining *de novo* motif finders with state-of-the-art motif modeling. In Section 3 we outline a problem derived from *A. thaliana* microarray expression experiments where certain transcription factors are over- or under-expressed. In our experiment we first illustrate that the straightforward approaches cannot achieve reasonable results, while the newly developed methods are able to drastically improve the target gene recognition performance.

2 Methods

2.1 Sequence Classification with Support Vector Machines

Support Vector Machine (SVMs) are a well-established machine learning method introduced by Boser, Guyon, and Vapnik [BGV92] to solve classification tasks frequently appearing in computational biology and many other disciplines. Typical examples are the classification of tumor images or gene expression measurements, the detection of biological signals in DNA, RNA or protein sequences as well as the recognition of hand-written digits or faces in images. SVMs are widely used in computational biology due to their high accuracy, their ability to deal with high-dimensional data, and their flexibility in modeling diverse sources of data [MMR+01, SS02, STV04, Nob06].

The domain knowledge inherent in the classification task is captured by defining a suitable *kernel function* $k(\boldsymbol{x}, \boldsymbol{x}')$ computing the similarity between two examples $\boldsymbol{x}$ and $\boldsymbol{x}'$. This strategy has two advantages: the ability to generate non-linear decision boundaries using methods initially designed for linear classifiers; and the possibility to apply a classifier to data that have no obvious vector space representation, for example, DNA/RNA or protein sequences as well as structures [BOS+08].

Spectrum Kernel Given two example sequences $\boldsymbol{x}$ and $\boldsymbol{x}'$ over the alphabet Σ, a simple way to compute the similarity is to count the number of co-occurring oligomers of fixed length ℓ. This idea is realized in the so-called *spectrum kernel* that was first proposed for classifying protein sequences [LEN02]: $k_\ell^{\text{spec}}(\boldsymbol{x}, \boldsymbol{x}') = \langle \Phi_\ell^{\text{spec}}(\boldsymbol{x}), \Phi_\ell^{\text{spec}}(\boldsymbol{x}') \rangle$, where $|\Sigma|$ is the number of letters in the alphabet. Φ_ℓ^{spec} is a mapping of the sequence $\boldsymbol{x}$ into a $|\Sigma|^\ell$-dimensional feature-space. Each dimension corresponds to one of the $|\Sigma|^\ell$ possible strings s of length ℓ and is the count of the number of occurrences of s in $\boldsymbol{x}$. This kernel is well-suited to characterize sequence similarity based on oligos that appear in both sequences– independent of their position.

If the classification of promoter sequences of genes as transcription factor targets would be solely based on binding to specific oligos, then the spectrum kernel appears to be a reasonable choice. If the motif is less conserved, then allowing for mismatches or gaps can be beneficial [LEWN03]. Note that this kernel is (by design) incapable of recognizing positional preferences TFs, and thus TFBSs, might have relative to the transcription start or among each other.

Weighted Degree Kernel Another kernel, the so-called *Weighted Degree Kernel* (WD) was proposed in [RS04, SRR07]. It computes the similarity of sequences of fixed length L by considering the substrings up to length ℓ starting at each position l separately:

$$k_\ell^{\text{wd}}(\boldsymbol{x}, \boldsymbol{x}') \quad = \quad \sum_{l=1}^{L} \sum_{d=1}^{\ell} \frac{\beta_d}{L} \mathbf{I}\left(\boldsymbol{x}_{[l:l+d]} = \boldsymbol{x}'_{[l:l+d]}\right) \text{ where } \beta_d = 2\frac{\ell - d + 1}{\ell^2 + \ell}, \tag{1}$$

and $\boldsymbol{x}_{[l:l+d]}$ is the substring of length d of $\boldsymbol{x}$ at position l [RS04, SRR07].

In the WD kernel, only oligos appearing at the same position in the sequence contribute to the similarity of two sequences. The *WD kernel with shifts* [RSS05] is an extension of the WD kernel allowing some positional flexibility of matching oligos:[1]

$$k_{\ell,S}^{\text{wds}}(\boldsymbol{x}, \boldsymbol{x}') = \sum_{l=1}^{L} \sum_{d=1}^{\ell} \sum_{\substack{s=0 \\ s+i \leq l}}^{S} \frac{1}{2dL(S+1)} \left(\mathbf{I}\left(\boldsymbol{x}_{[l+s:l+d+s]} = \boldsymbol{x}'_{[l:l+d]}\right) + \mathbf{I}\left(\boldsymbol{x}_{[l:l+d]} = \boldsymbol{x}'_{[l+s:l+d+s]}\right) \right) \tag{2}$$

It considers oligomers up to length d, and allows them to be shifted up to S positions, starting from i, in the input sequences. This kernel is better suited for motifs with indels or at varying positions (see *e.g.* [RSS05, SSP+07]

[1] The *locality improved* and *oligo* kernel [ZRM+00, MTMM04] achieve a similar goal in a slightly different way.

2.2 Extensions

In this section we extend the WD kernel in two different ways: First, we consider an extension to use conservation information. Second, given a list of potential motifs we propose a new kernel that integrates information on the motif sequences with the information about their co-occurrence with the aim to characterize regulatory modules.

WD Kernel with Conservation Information To include conservation information, we extended the WDS kernel with a term to multiply the score of the local matches of an oligo of length d at position i with a quantity that depends on its conservation. We propose to use the average conservation of the oligo in pre-generated alignments of sequences from G other organisms:

$$\gamma^{A}_{d,i,\boldsymbol{x}} = 1 + \frac{A}{d} \sum_{g=1}^{G} \sum_{j=0}^{d} \mathbf{I}(\boldsymbol{x}_{i+j} = \boldsymbol{x}^{g}_{i+j}), \tag{3}$$

where $\boldsymbol{x}^g$ is the sequence of the syntenic regions in the genome of organism $g = 1, \ldots, G$ and $A < 0$ is a parameter allowing one to control the importance of the conservation. The fact that we add 1 means we only value an existing alignment positively, but do not further punish the absence of an alignment. All results shown were obtained with the setting of $A = 1$. Using this definition of a conservation score we can now define the *weighted degree with shifts and conservation* (WDSC):

$$k^{\mathrm{wdsc}}_{\ell,S,A}(\boldsymbol{x}, \boldsymbol{x}') = \sum_{l=1}^{L} \sum_{d=1}^{\ell} \sum_{\substack{s=0 \\ s+i \leq l}}^{S} \frac{\gamma_{d,i,\boldsymbol{x}} \gamma_{d,i,\boldsymbol{x}'}}{2d(s+1)} \left(\mathbf{I}\left(\boldsymbol{x}_{[l+s:l+d+s]} = \boldsymbol{x}'_{[l:l+d]}\right) + \mathbf{I}\left(\boldsymbol{x}_{[l:l+d]} = \boldsymbol{x}'_{[l+s:l+d+s]}\right) \right)$$

A Kernel for Regulatory Modules

Suppose we are given a set of M motifs $\mathcal{M}_m$, $m = 1, \ldots, M$ that may either come from a database or from a *de novo* motif detection method. Such motifs are often represented in a way that one can easily scan a given sequence for occurrences of the motif (*e.g.* as PWMs). In a preprocessing step we compute the best-matching position $p_{m,\boldsymbol{x}}$ of each motif $\mathcal{M}_m$ in all considered sequences $\boldsymbol{x}$. In case of PWMs, the PWM score and in case of oligo-based motifs the Hamming distance may be used to decide which position in the sequence matches best.[2]

The main idea of the kernel that we propose is to represent an input sequence $\boldsymbol{x}$ by the set of sequences $\boldsymbol{x}_m := \boldsymbol{x}_{[p_{m,\boldsymbol{x}}-w, p_{m,\boldsymbol{x}}+w]}$ originating from the region of length $2w$ around the best motif match $p_{m,\boldsymbol{x}}$ of motif $\mathcal{M}_m$. Each sequence region $\boldsymbol{x}_m$ contributes independently to the similarity between two input sequences: $k_1(\boldsymbol{x}, \boldsymbol{x}') = \sum_{m=1}^{M} k(\boldsymbol{x}_m, \boldsymbol{x}'_m)$. This term characterizes the co-occurrence of a collection of motifs in two sequences $\boldsymbol{x}$ and $\boldsymbol{x}'$. The similarity is highest if all motifs appear in both sequences (in arbitrary order). We propose to use a position specific kernel, for instance the WDS kernel, to compute the similarity of the regions.

[2] For the kernel functions, all input vectors need to be of the same length. Therefore, in our method we have to choose the same number of matches per sequence for all motifs (1 in our case), regardless of the quality of the matches. Biologically, a threshold quality seems more intuitive, then several good matches would be considered and no match for sequences that don't contain the motif. However, a soft margin during training allows the algorithm to ignore some mislabeled data points without effects on generalization.

For the first part of the kernel, the position of the motif does not influence the similarity at all. In the second part of the kernel we try to capture the relative position of the best motif matches to each other and to the transcription start site. This is done by computing all pairwise distances between match positions of motifs: $v(\boldsymbol{x}) = (p_{1,\boldsymbol{x}} - p_{tss}, \ldots, p_{M,\boldsymbol{x}} - p_{tss}, p_{1,\boldsymbol{x}} - p_{2,\boldsymbol{x}}, \ldots, p_{i,\boldsymbol{x}} - p_{j,\boldsymbol{x}}, \ldots, p_{M-1,\boldsymbol{x}} - p_{M,\boldsymbol{x}})^\top$, for all $i \neq j = 1, \ldots, M$, where p_{tss} is the position of the transcription start site in the sequence. A simple way of computing the similarity between two such vectors is to use the RBF kernel (*e.g.* [SS02]): $k^{\mathrm{rbf}}(\boldsymbol{v}, \boldsymbol{v}') = \exp\left(\frac{\|\boldsymbol{v}-\boldsymbol{v}'\|^2}{\sigma}\right)$, where σ is a kernel hyper parameter to be found by model selection.

Having both parts of the kernel defined, the question remains of how to combine them. We propose to simply add both contributions: $k(\boldsymbol{x}, \boldsymbol{x}') = k_1(\boldsymbol{x}, \boldsymbol{x}') + k^{\mathrm{rbf}}(v(\boldsymbol{x}), v(\boldsymbol{x}'))$. Please note that if we add the two kernels, it amounts to concatenating the two feature spaces. If one would multiply the contributions of distances and motif-sequence similarity, then the kernel would be in some sense similar to the previously proposed oligo kernel [MTMM04].

2.3 KIRMES Pipeline

Below we describe an integrated PYTHON [Pyt07] pipeline, called KIRMES, using the previously described kernels to classify promoter regions of genes as transcription factor targets or not.[3] It assumes that the sequences of promoter regions are given in two sets: A set enriched with transcription factor targets (labeled positive) and a second set containing no or very few targets (labeled negative). Figure 1 shows an outline of the pipeline for the classification of promoter sequences based on microarray experiments (*cf.* Section 3.1).

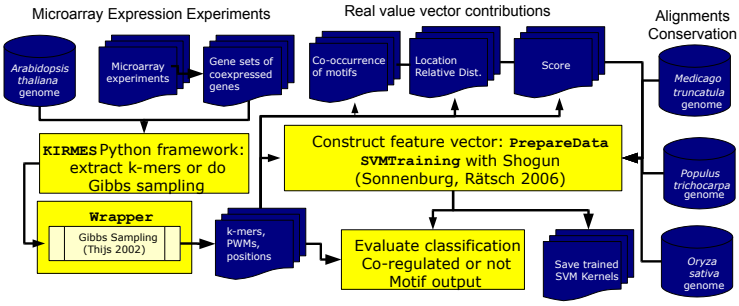


Figure 1: Workflow of KIRMES: The pre-processing step requires the genomic sequence and a set of genes that were measured to be coexpressed in microarray experiments. KIRMES extracts the k-mers and puts them into a vector along with the positional and conservation information and the score, as described in Section 2.2. The SVM is trained on a labeled data set of positives and negatives and can then be applied repeatedly on unlabeled testing datasets.

[3] Sequences considered in the set can be any part of the euchromatin of arbitrary length, *e.g.* upstream and downstream regions of a gene, intronic and exonic parts as well as untranslated regions (UTR) in the 3' or 5' direction. Good results can be obtained with a combination of upstream, UTR and intronic sequences. We used 2000 kb upstream of the translation start; in general longer sequences introduce more noise. Therefore, in organisms with shorter promoters a reduction would be beneficial for the signal to noise ratio.

Initial Motif Finding In a first step we used one of two methods to identify potential motifs. Initially, we used a common Gibbs sampling algorithm [LAB+93] called MOTIFSAMPLER from the INCLUSIVE package [TMDS+02] that finds overrepresented motifs. To make sure we do not include motifs that are too common, we use several strategies: first, a background model for this organism; second, minimum occurrences were set to 15% or three genes of the set, whichever is more; third, one thousand random gene sets were generated and searched for motifs of the same length and determinacy. This was measured through the information content of the position frequency matrix of the motif, an output of the Gibbs sampling program.

Since this last step takes a significant amount of time depending on the length of the sequences, we searched for alternatives. We settled on one approach, where we count the occurrence of any oligomer of length six in positive sequences (*oligo-counting*). We select a subset of those oligomers that appear in at least 15% of all positive sequences. This simple strategy certainly leaves room for improvements, but our experiments in Section 3 illustrate that it already works rather well.

SVM Training We use the large scale machine learning toolbox SHOGUN [SRSS06] through its PYTHON interface. It provides implementations of all kernels described in this work and allows fast training using several different SVM implementations, *e.g.* SVM^{light} [Joa99].

Galaxy Web Service KIRMES is available publicly on our Galaxy webserver at `http://www.fml.mpg.de/raetsch/projects/kirmes`. Galaxy is an open-source, scalable framework for tool and data integration [GRH+05]: Users can upload their sequence files and KIRMES will classify the input gene set and return the names of the co-regulated genes in a list. This can be done for any regulatory region like promotors, introns, or even the whole chromatin of arbitrary length, and for any organism. To successfully use the positional information in promotor regions, it is a good idea to select the sequences in such a way that the translation start site is at the same position in each of them.

For this web service, the 6-mer enumeration strategy and the weighted degree kernel with shifts is used. The use of conservation information is not supported as it depends on the organism from which the sequences were obtained, it may not always be available and would require a significantly larger infrastructure. There is no upper limit on the amount of input sequences, but at least 5 sequences should be uploaded.

3 Experiments

We first describe a dataset which has been used to test the presented methods. The goal is to predict the expression change status of potential target genes for over-expressed transcription factors based on their promoter sequence.

3.1 Microarray Expression Data

We derive sets of co-expressed genes from microarray experiments performed with the commercial *Affymetrix GeneChip Arabidopsis ATH1* array. This chip is designed to measure transcript abundance of more than 20 000 genes of the model organism *Arabidopsis thaliana* [RHTT04].

The sets are obtained through a stringent analysis of expression change using the software GeneSpring [Agi06]. We labeled genes as co-expressed when they showed a four-fold change of expression in the experiment as compared to the control, and considered those genes not co-expressed if their levels remain the same, compared to the control, within a margin of 0.2 fold change. Thus we obtained sets of co-expressed genes.[4]

We used microarray data from two different experimental setups (*cf.* Appendix A.1 in the Supplementary Materials). The first setup uses leaves from wild type *Arabidopsis thaliana* plants exposed to medium at 38 °C *versus* leaves exposed to the same medium at room temperature, expression measurement taken one hour after exposure [BWS05]. The second setup uses inducible over-expression of *Arabidopsis* meristem regulators with the AlcR/AlcA system. Plants harboring 35S::AlcR/AlcA::GOI (GUS control, LEAFY, SHOOTMERISTEMLESS, WUSCHEL) constructs were grown in continuous light for 12 days and induced with 1% ethanol. After 12 hours of EtOH treatment, seedlings were dissected and RNA was processed from the shoot apex and from young leaves. Affymetrix ATH1 arrays were hybridized in duplicates for each gene construct and condition [LTB$^+$05]. In total we considered 14 different gene sets to be discriminated by the methods.

3.2 Experimental Setup

To train and test the method we first split the data into two parts (80%:20%). The first part is used for motif finding and SVM training. For hyper-parameter tuning we used the first part with 5-fold cross-validation to find the optimal combinations of hyper-parameters. (The SVM and the considered kernels have several hyper-parameters to be given in advance. This includes the regularization parameter C of the SVM, the maximal length of oligomers ℓ and the maximal shift S considered in the WDS kernel.) The second part is used for estimating the generalization performance. Here we measure the area under the ROC curve (auROC) as the generalization performance (random guessing corresponds to 50% auROC).

The above procedure is repeated five times for different splits of training and test examples (outer cross-validation loop). As performance measure we report the average auROC over the five splits.

[4] The fold change is computed from the normalized gene expression level p in treatment and respective control, c: $n = \begin{cases} -c/p & \text{if } p/c < 1 \\ p/c & \text{if } p/c \geq 1 \end{cases}$. In this case the direction of the change is represented by the sign of n, positive means up and negative means down relative to the control. If several replicates were available, the mean after normalization is taken for every gene, for all replicates of p and c respectively.

3.3 Results

In a first experiment we illustrate that simple methods as for instance SVMs with spectrum or WDS kernel cannot easily solve the considered classification problem. The results are given in Figure 2. We can observe that essentially for all gene sets SVMs with Spectrum kernel fails to identify positive genes (auROC close to 50%). The SVM with WDS kernel is slightly better, but still produces close to random predictions.

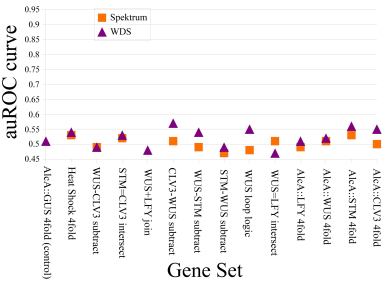


Figure 2: Accuracy of the Spectrum and WDS kernels: The prediction is rarely better than random guessing for these kernels. The kernels are not well suited for the this particular problem.

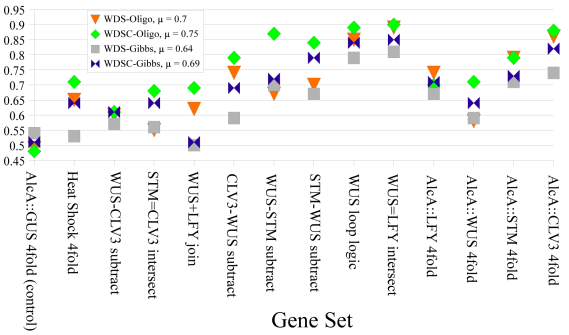


Figure 3: Accuracy of variations of the KIRMES approach: This graph shows a comparison of the basic kernels and the conservation kernels (C) combined with two different motif generation approaches: by oligo-counting (Oligo) or by Gibbs sampling (Gibbs). The average performance (μ) is given for each kernel variant. The first set is taken from a control experiment.

In Figure 3, we present results of the proposed methods in four variants: with motif discovery by Gibbs-sampling *vs.* oligo-counting as well as with and without using conservation (*cf.* Appendix A.2). We can make the following observations: (i) All four versions show a significantly improved performance relative to the base-line methods. (ii) Motif-finding using oligo-counting seems to work considerably better in combination with SVMs than Gibbs-sampling. A possible reason may be that the number of considered oligos (100-200) is higher than the number of motifs generated by the Gibbs-sampler (less than 50). (iii) Using conservation as weighting for the WDS kernel considerably improves the recognition performance. It results in an average improvement of 5 percentage points.

These results clearly illustrate the power of our approach in exploiting the relationship between motifs as well as the conservation to improve the recognition of transcription factor targets.

The algorithm can be used for any combination of regulatory regions and also any organism. Use of the web service integrated into Galaxy is straightforward and the resulting classification can help scientists with experimental microarray data select genes they want to investigate further.

An integration of protein binding data such as from chromatin immunoprecipitation experiments on a microarray chip is planned for a future extension of this method. Binding data can for example contribute to the weighting of a certain transcription factor binding

site and the surrounding sequence, just like conservation information. In that respect, the normalization scheme for the number of contributing related organisms can be remodeled to take into account their evolutionary distances and to generalize it further.

The use by experimentalists will ultimately determine the utility of this approach and govern the direction of further extensions together with technological advances such as Next Generation Sequencing methods for transcriptome or protein binding data.

Acknowledgments The authors thank the anonymous reviewers for their helpful suggestions that improved the manuscript. G.R. would like to thank Gabriele Schweikert for comments on the manuscript.

References

[Agi06] Agilent. GeneSpring GX. Technical report, Agilent Technologies, 2006.

[BE94] T.L. Bailey and C. Elkan. Fitting a mixture model by expectation maximization to discover motifs in biopolymers. In *Proc. ISMB'94*, volume 2, pages 28–36, Menlo Park, California, USA, 1994. ISCB, AAAI Press.

[BGV92] B.E. Boser, I.M. Guyon, and V.N. Vapnik. A training algorithm for optimal margin classifiers. In *Proc. COLT '92*, pages 144–152, Pittsburgh, Pennsylvania, United States, 1992. ACM Press.

[BOS+08] A. Benhur, C.S. Ong, S. Sonnenburg, B. Schölkopf, and G. Rätsch. Support Vector Machines and Kernels for Computational Biology. *PLoS Computational Biology*, 2008. forthcoming.

[BWS05] W. Busch, M. Wunderlich, and F. Schoeffl. Identification of novel heat shock factor-dependent genes and biochemical pathways in A. thaliana. *Plant J*, 41(1):1–14, 2005.

[FSKB08] M.C. Frith, N.F. Saunders, B. Kobe, and T.L. Bailey. Discovering sequence motifs with arbitrary insertions and deletions. *PLoS Comput Biol*, 4(4):e1000071, 2008.

[GL05] M. Gupta and J.S. Liu. De novo cis-regulatory module elicitation for eukaryotic genomes. *Proc Natl Acad Sci U S A*, 102(20):7079–7084, 2005.

[GRH+05] B. Giardine, C. Riemer, R.C. Hardison, R. Burhans, L. Elnitski, and et al. Galaxy: a platform for large-scale genome analysis. *Genome Res*, 15(10):1451–1455, 2005.

[Joa99] T. Joachims. Making large-Scale SVM Learning Practical. In Bernhard Schölkopf, C Burges, and Alexander J. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*. MIT Press, Cambridge, MA, USA, 1999.

[LAB+93] C. E. Lawrence, S. F. Altschul, M. S. Boguski, J. S. Liu, A. F. Neuwald, and J. C. Wootton. Detecting subtle sequence signals: a Gibbs sampling strategy for multiple alignment. *Science*, 262(5131):208–14, October 1993. 0036-8075

[LEN02] C. Leslie, E. Eskin, and W. S. Noble. The Spectrum Kernel: A String Kernel For SVM Protein Classification. In *Proc. PSB'02*, pages 564–575, 2002.

[LEWN03] C. Leslie, E. Eskin, J. Weston, and W.S. Noble. Mismatch String Kernels for Discriminative Protein Classification. *Bioinformatics*, 20(4), 2003.

[LTB+05] A. Leibfried, J.P.C. To, W. Busch, S. Stehling, A. Kehle, M. Demar, J.J. Kieber, and J.U. Lohmann. WUSCHEL controls meristem function by direct regulation of cytokinin-inducible response regulators. *Nature*, 438:1172–1175, December 2005.

[MFG+03] V. Matys, E. Fricke, R. Geffers, E. Gossling, M. Haubrock, and et al. TRANSFAC: transcriptional regulation, from patterns to profiles. *Nucleic Acids Res*, 31(1):374–8, January 2003. 1362-4962

[MMR+01] K.R. Müller, S. Mika, G. Rätsch, K. Tsuda, and B. Schölkopf. An introduction to kernel-based learning algorithms. *IEEE Trans. Neural Netw.*, 12:181–201, 2001.

[MTMM04] P. Meinicke, M. Tech, B. Morgenstern, and R. Merkl. Oligo kernels for datamining on biological sequences: a case study on prokaryotic translation initiation sites. *BMC Bioinformatics*, 5(169), 2004.

[Nob06] W.S. Noble. What is a Support Vector Machine? *Nature Biotechnology*, 12(24):1565–1567, 2006.

[Pyt07] Python Software Foundation. Python. `http://python.org`, May 2007.

[RHTT04] J.C. Redman, B.J. Haas, G. Tanimoto, and C.D. Town. Development and evaluation of an Arabidopsis whole genome Affymetrix probe array. *Plant J*, 38(3):545–561, 2004.

[RS04] G. Rätsch and S. Sonnenburg. Accurate Splice Site Detection for *Caenorhabditis elegans*. In K. Tsuda B. Schölkopf and J.-P. Vert, editors, *Kernel Methods in Computational Biology*, pages 277–298. MIT Press, 2004.

[RSS05] G. Rätsch, S. Sonnenburg, and B. Schölkopf. RASE: Recognition of Alternatively Spliced Exons in *C. elegans*. *Bioinformatics*, 21(Suppl. 1):i369–i377, June 2005.

[SAE+04] A. Sandelin, W. Alkema, P. Engstrom, W. W. Wasserman, and B. Lenhard. JASPAR: an open-access database for eukaryotic transcription factor binding profiles. *Nucleic Acids Res*, 32(Database issue):D91–4, January 2004. 1362-4962

[SRR07] S. Sonnenburg, G. Rätsch, and K. Rieck. Large Scale Learning with String Kernels. In L. Bottou, O. Chapelle, D. DeCoste, and J. Weston, editors, *Large Scale Kernel Machines*, pages 73–104. MIT Press, 2007.

[SRSS06] S. Sonnenburg, Gunnar Rätsch, C. Schäfer, and Bernhard Schölkopf. Large Scale Multiple Kernel Learning. *J of Mach Learn Res*, 7(Jul):1531–1565, July 2006.

[SS02] B. Schölkopf and A. Smola. *Learning with Kernels*. MIT Press, Cambridge, 2002.

[SS05] E. Segal and R. Sharan. A Discriminative Model for Identifying Spatial Cis-Regulatory Modules. *J of Comp Biol*, 12:822–834, 2005.

[SSGE86] T. D. Schneider, G. D. Stormo, L. Gold, and A. Ehrenfeucht. Information content of binding sites on nucleotide sequences. *J Mol Biol*, 188:415–431, April 1986.

[SSP+07] S. Sonnenburg, G. Schweikert, P. Philips, J. Behr, and G. Rätsch. Accurate splice site prediction using SVMs. *BMC Bioinformatics*, 8(Suppl. 10):S7, 2007.

[ST02] Saurabh Sinha and Martin Tompa. Discovery of novel transcription factor binding sites by statistical overrepresentation. *Nucleic Acids Res*, 30(24):5549–5560, 2002.

[Sto00] G. D. Stormo. DNA binding sites: representation and discovery. *Bioinformatics*, 16(1):16–23, January 2000.

[STV04] B. Schölkopf, K. Tsuda, and J.-P. Vert. *Kernel methods in computational biology*. MIT Press, Cambridge, MA, 2004.

[TMDS+02] G. Thijs, Y. Moreau, F. De Smet, J. Mathys, M. Lescot, S. Rombauts, P. Rouze, B. De Moor, and K. Marchal. INCLUSive: integrated clustering, upstream sequence retrieval and motif sampling. *Bioinformatics*, 18(2):331–2, February 2002.

[YTI+98] T Yada, Y Totoki, M Ishikawa, K Asai, and K Nakai. Automatic extraction of motifs represented in the hidden Markov model from a number of DNA sequences. *Bioinformatics*, 14(4):317–325, 1998.

[ZRM+00] A. Zien, G. Rätsch, S. Mika, B. Schölkopf, T. Lengauer, and K.-R. Müller. Engineering Support Vector Machine Kernels That Recognize Translation Initiation Sites. *Bioinformatics*, 16(9):799–807, September 2000.

FRANz: Fast reconstruction of wild pedigrees

Markus Riester[1], Peter F. Stadler[1,2,3,4], Konstantin Klemm[1]
[1]Bioinformatics Group, Department of Computer Science,
and Interdisciplinary Center for Bioinformatics,
University of Leipzig, Härtelstrasse 16-18, D-04107 Leipzig, Germany.
[2]RNomics Group, Fraunhofer Institut for Cell Therapy and Immunology (IZI),
Deutscher Platz 5e, D-04103 Leipzig, Germany
[3]Institute for Theoretical Chemistry, University of Vienna,
Währingerstrasse 17, A-1090 Vienna, Austria
[4]The Santa Fe Institute, 1399 Hyde Park Rd., Santa Fe, New Mexico
{markus,studla,klemm}@bioinf.uni-leipzig.de

Abstract: We present a software package for fast pedigree reconstruction in natural populations using co-dominant genomic markers such as microsatellites and SNPs. If available, the algorithm makes use of prior information such as known relationships (sub-pedigrees) or the age and sex of individuals. Statistical confidence is estimated by a simulation of the sampling process. The accuracy of the algorithm is demonstrated for simulated data as well as an empirical data set with known pedigree. The parentage inference is robust even in the presence of genotyping errors.

1 Introduction

The reconstruction of genealogical relationships among diploid species has been an active field of research for more than three decades. A well-developed statistical theory of paternity inference has been published in series of articles by E.A. Thompson, see e.g. [Tho76]. The study of parentage in natural populations was the topic of the pioneering papers by T.R. Meagher [MT86] and T.C. Marshall [MSKP98] and is recently reviewed in [Blo03, JA03, Pem08]. The pedigree structure of a sample of individuals is important for a wide range of ecological, evolutionary and forensic studies. Applications include genealogy reconstruction (e.g. for wine grape cultivars [VG06]), the estimation of heritabilities in the wild [TH00], and victim identification [LMX06].

In order to reconstruct the pedigree of a sample, the parents of each individual in the sample need to be determined. If one has a large amount of genomic data, the task of identifying first degree relationships, i.e., parent-offspring and full-sibs relations, is trivial. Unfortunately, many datasets in natural populations do not contain enough information to unambiguously determine the parents. Another problem is that datasets often contain only a subset of a population. Thus, one or both parents of an observed individual may be missing from the dataset. Furthermore, many datasets are not free of errors.

Most programs support only datasets comprising one or two generations. The approach

to partial pedigree reconstruction in one generation datasets are sibship algorithms. Here, genotype data is used to infer full-sib and half-sib relationships [TH02, Wan04, BWSD$^+$07]. The parentage inference programs for two generations typically take an offspring list, if known their mothers, and a list of candidate parents or fathers as input and generate the possible parent combinations [KTM07, HRB06]. Much less attention has been given to multi-generation pedigrees in which the offspring and candidate parent sets are not necessarily non-overlapping. This is the case for example in the absence of age data. Then the ordering of genotypes into generations is not known a priori and has to be estimated from the genotype data only. Thus, at difference with parentage inference programs, the general case treated here does not admit all possible parentage combinations as valid pedigrees. The task is therefore to find the parentage combinations that define the *maximum likelihood pedigree*. If the number of possible pedigrees is too large too enumerate, heuristics are necessary. So far, a flexible software package has not been available that allows the incorporation of prior information in addition to the genotypes and that is robust in the case of errors. It is the purpose of this contribution to fill this gap.

2 Definitions

A pedigree $\mathscr{P} = (V, A)$ is an acyclic digraph with vertex set V and arc set A. For an arc (u_i, v) we say that v is a *child* of u_i and u_i is a *parent* of v. The set of (putative) *parents* of v is denoted by $N^+(v) \subseteq V$; it may have cardinality 2 $\{u_i, u_j\}$, 1 $\{u_i\}$, or 0 $\emptyset$. In the latter case, v is called a *founder*. In selfing species, $u_i = u_j$ is allowed and $\mathscr{P}$ is a multigraph. The set of all valid parent combinations of v is denoted by $\mathscr{H}(v)$. Again we include the cases that none or only one of the parents are present in V. Note that $\mathscr{H}(v) \subset V \times V \cup V \cup \{\varnothing\}$. The Mendelian laws of inheritance and *prior information* such as sex, age and known mothers restrict $\mathscr{H}(v)$.

For each individual, we have to choose one parent combination $N^+(v) \in \mathscr{H}(v)$. Not all such combinations of parents are possible, because this may introduce directed cycles into the pedigree. $\mathscr{T}$ denotes the set of all *valid pedigrees*.

For a given individual i, we denote an observed single-locus genotype by g_i and its multilocus genotype by G_i.

3 Background

Consider a triplet of individuals (*A, B, C*) with single locus genotypes g_A, g_B and g_C. In likelihood-based paternity analyses, one compares the likelihood of the hypothesis (H_1) that the three individuals are offspring, mother and father, with the likelihood of the alternative hypothesis (H_2) that the three individuals are unrelated. This comparison is usually

expressed as a log-ratio, the *parent-pair LOD score* (e.g. [MT86]):

$$\mathrm{LOD}(g_A, g_B, g_C) = \log \frac{P(g_A, g_B, g_C|H_1)}{P(g_A, g_B, g_C|H_2)} = \log \frac{T(g_A|g_B, g_C) \cdot P(g_B) \cdot P(g_C)}{P(g_A) \cdot P(g_B) \cdot P(g_C)}$$

The likelihood of (H_2) is the probability of observing the three genotypes when randomly drawn from a population in Hardy-Weinberg equilibrium. For diploid heterozygotes, the probability of a genotype with the alleles a_1 and a_2 and with the allele frequencies p and q is $P(a_1, a_2) = 2pq$; for homozygotes, we have $P(a_1, a_1) = p^2$. The Mendelian transmission probability is denoted by $T(\cdot)$. Variations of this equation can be derived for the cases where only one parent is sampled (*single-parent* LOD scores) and for triples where the relationship of two individuals A and B, typically mother and offspring, is known [MT86, KTM07].

For each pair of individuals, we can calculate the probability that the two have a particular relationship $\mathbb{R}$: unrelated $\mathbb{U}$, parent-offspring $\mathbb{PO}$, full-sib $\mathbb{FS}$, half-sib $\mathbb{HS}$, etc. The usual way of calculating the likelihoods $P(g_A.g_B|\mathbb{R})$ uses the so-called *IBD (identical by descent) coefficients* k_0, k_1 and k_2. Alleles are identical by descent if they are identical and are segregated from a recent common ancestor. A child, for example, shares with each parent exactly one allele that is identical by descent ($k_1 = 1$); monozygotic twins share two ($k_2 = 1$) whereas unrelated individuals share no alleles ($k_0 = 1$) identical by descent. Given the allele frequencies, the probabilities that the genotype pair $g_A.g_B$ shares 0, 1 or 2 alleles identical by descent, P_0, P_1 and P_2, are then calculated and are inserted in the final IBD likelihood formula (see [Blo03] for details):

$$P(g_A.g_B|\mathbb{R}) = k_0 P_0 + k_1 P_1 + k_2 P_2 \quad (k_0 + k_1 + k_2 = 1)$$

For unlinked loci, which we assume in the following, the logarithms of the IBD relationship likelihoods and the LOD scores are additive over the loci.

Even high quality datasets contain errors where at least one allele at a given locus does not match with what we expect from the Mendelian laws. Thus it is unwise to exclude a parent immediately when observing such a mismatch. There are many reasons for such mismatches, see [BBBE$^+$04] for a review. Genotyping errors occur when the genotype determined by molecular analysis does not correspond to the real genotype. For instance, a common type of genotyping error in microsatellite datasets are null alleles, which are often the result of a mutation in the primer annealing site. Somatic mutations form another source of mismatches.

The model implemented here defines an error to be the replacement of the true genotype at a particular locus in an individual with a random genotype. This leads to a modification of the expressions for the LOD score, see [KTM07], and to corresponding modifications in the IBD likelihood calculations, see [BW98] for details.

4 Methods

4.1 Simulation of the sampling process

To estimate the power of the marker suite, our software performs several standard tests and calculations. This alone, however, will not be sufficient to estimate the accuracy of the pedigree reconstruction. A simulation of the sampling process is therefore necessary. Given the population allele frequencies and the expected typing error rate, which are either estimated using the sample itself or provided by the user, we generate individuals with known relationships to determine various distributions. To assess the degree of confidence of the parent-offspring arcs in $\mathscr{P}$, we follow [MSKP98] in using ΔLOD as test statistic. ΔLOD is the difference of the LOD scores between the two most likely parent combinations (or fathers).

Another important characteristic is the distribution of the number of mismatching loci given the expected error rate for pairs (parent-offspring *versus* unrelated) as well for triples (offspring, mother and father *versus* offspring, mother and unrelated male). This knowledge allows us to significantly speed up the algorithm, because we know when likelihood calculations can be terminated. We can furthermore omit the $O(n^3)$ triple calculation for pairs with more mismatches than maximally expected for a triple. These parameters are also important because too many allowed mismatches may lead to a high number of false positive parent-offspring arcs.

Full-sibs can be distinguished from parent-offspring pairs based on the log-likelihood differences $\Delta_{po} = \log P(G_i.G_j|\mathbb{FS}) - \log P(G_i.G_j|\mathbb{PO})$. The distribution of Δ_{po} is generated for full-sib pairs and for parent-offspring pairs. We later only consider pairs that exceed a critical value of Δ_{po} as full-sib candidates. If the intersection of their candidate parents includes at least one parent pair, we finally define this pair as full-sibs. If not, then the pair could still be a full-sib pair, but with unsampled parents. In this case, this pair could also be a half-sib pair, so we use the distribution of the log-likelihood differences $\Delta_{hs} = \log P(G_i.G_j|\mathbb{FS}) - \log P(G_i.G_j|\mathbb{HS})$ to distinguish full-sibs from half-sibs. The values of Δ_{hs} are generated for full-sib and half-sib pairs. Now, full-sib candidates without a common parent pair that exceed a critical value of Δ_{hs}, are defined as full-sibs.

4.2 Calculation of the possible parent-offspring arcs

For every individual v, we calculate the LOD scores with all candidate parents u_i, individuals we cannot exclude *a priori* as parents, for example because of their age. We discard pairs (u_i, v) or triples (u_i, u_j, v) with negative multilocus LOD scores from our further analyses. Hence, for every pair of individuals with positive single-parent LOD score, $(u_i, ?)$ is included in the set of valid parent combinations $\mathscr{H}(v)$, just as well (u_i, u_j) for every triple with positive parent-pair LOD score. Unless we know that at least one parent of v is sampled, we include the empty parent pair $(?, ?)$ in $\mathscr{H}(v)$.

We also calculate the *posterior probability* of each parentage in $\mathscr{H}(v)$ relative to all pos-

sible parentages including the ones with unsampled candidates parents [NMCP01]. Our implementation provides the functionality to filter parentages below a certain probability.

The parentage likelihood calculation is the most important step in the pedigree reconstruction procedure as these likelihoods define the set of all possible arcs in the pedigree. However, as described in detail by Meagher and Thompson [TM87], if we cannot exclude two full-sibs, v_i and v_j, as parent and offspring, they in general give a higher likelihood than do true parents. Thus, for highly probable full-sibs, a reasonable strategy is to use only the intersection of the candidate parents: $\mathscr{H}(v_i) = \mathscr{H}(v_j) = \mathscr{H}(v_i) \cap \mathscr{H}(v_j)$. The critical values of Δ_{po} and Δ_{hs} that a full-sib pair must exceed should be high enough to prevent false positives, which may result in an exclusion of the true parents in the next step, the pedigree reconstruction.

4.3 Pedigree Reconstruction

The likelihood of a pedigree $\mathscr{P}$ is computed as the probability of the genotypes given this pedigree. So the goal is to find the pedigree which maximizes the log-likelihood:

$$\max_{\mathscr{P} \in \mathscr{T}} L(\mathscr{P}) \quad = \quad \sum_{i=1}^{N_I} \log P(G_i | N^+(v_i))$$

Here, $P(\cdot)$ is the probability of observing the multilocus genotype G_i given the parents $N^+(v_i)$. For founders ($N^+ = \emptyset$), $\log P(\cdot)$ equals the denominator of the multilocus LOD score. This is equivalent to the assumption that all founders are unrelated. For the offspring, these probabilities are the multilocus Mendelian transition probabilities in the error model. So for vertices where $|N^+| = 1$, $\log P(\cdot)$ is the *single-parent*, when $|N^+| = 2$ the *parent-pair* LOD enumerator.

For each individual, we now sort the possible parent combinations by their probability. The maximal possible score is simply the sum of all most likely parent combinations. Our greedy algorithm works by selecting one vertex v and then adding the arcs corresponding to the most likely parent combination $N^+ \in \mathscr{H}(v)$. If the arcs introduce a directed cycle in $\mathscr{P}$, we try the second most likely parent combination and so on. If no parent-offspring relationships are known, this algorithm produces a valid pedigree, because the 'empty' parent combination (v is a founder) is always in $\mathscr{H}(v)$, which can never introduce a cycle. We proceed until all vertices are added.

For vertices with known parents, every parent combination adds at least one arc. A simple strategy is now to start with vertices where $|\mathscr{H}(v)| = 1$. Unless the "known" parent-offspring relationships are wrong, this introduces no directed cycles. Then we proceed with the remaining vertices with known parents. If this succeeds, we add the remaining vertices without known parents as described above. If not, or if the final score is not the maximal score, we use Simulated Annealing [KGV83] for the pedigree reconstruction as described in [Alm03].

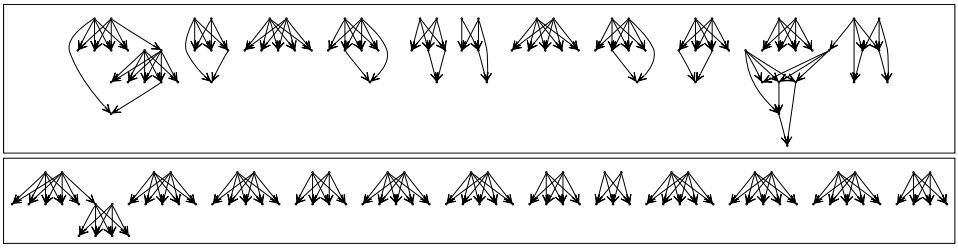


Figure 1: Reconstructed *penaeus monodon* pedigree. Without (top) and with (below) full-sib calculation. The assumed typing error rate is 0.01.

5 Results

Black Tiger Shrimp *Penaeus monodon*. Our first dataset is a microsatellite dataset of the black tiger shrimp *Penaeus monodon* [JBMW06]. The true pedigree is known from direct observation. The dataset consists of 13 families with a total number of 85 individuals (of which 59 offspring), genotyped at seven highly polymorphic loci. For ten individuals, alleles are missing at one locus. The error rate is very low, with only one observed mismatch. Figure 1 are the best pedigrees with and without full-sib heuristic (4.2) and shows that large full-sib groups greatly enhance the performance of our algorithm. The accuracy of the complete pedigree without full-sib heuristic is 87.4% in comparison to 99.58% with this heuristic. A recent publication [BWSD$^+$07] listed an accuracy rate of several sibling reconstruction methods ranging from 67.8 to 77.97 percent on the same dataset.

Simulated Data. We use the statistics of the German population [Off07] to calculate the probabilities of death, (multiple) birth and marriage at a given age for males and females. As initial population we generate 100 unrelated individuals. For the genotypes, we use the allele frequencies of 64 human microsatellites [JBCS$^+$00]. In every year, we let all individuals die, mate or marry according the corresponding probabilities. As mating partners or husbands, we only allow unrelated individuals. Married couples only mate with each other. We stop when the desired number of individuals is reached. In order to simulate typing errors, we replace the true allele with a random one. Null alleles are simulated in heterozygote genotypes by replacing the null allele with the other allele ($a_i.a_n$ becomes $a_i.a_i$). Homozygote genotypes are marked as missing, i.e., $a_n.a_n$ becomes $?.?$.

We analyzed the accuracy of our algorithm with different subsets of the simulated data, see Figure 2. If the accuracy is not 100%, then either the algorithm failed to find the maximum likelihood pedigree or there exists a valid pedigree that has a higher likelihood than the true one. Without exceptions, our optimization algorithm found a pedigree with at least the log-likelihood of the true pedigree (data not shown).

Age data is clearly the most informative prior knowledge. Known mothers and the sex of the individuals are getting less informative the more genomic data is available. This is because mothers are sampled like all individuals with a rate of 0.5 and sex requires candidate parent pairs. So for example, they do not help in difficult cases where the true

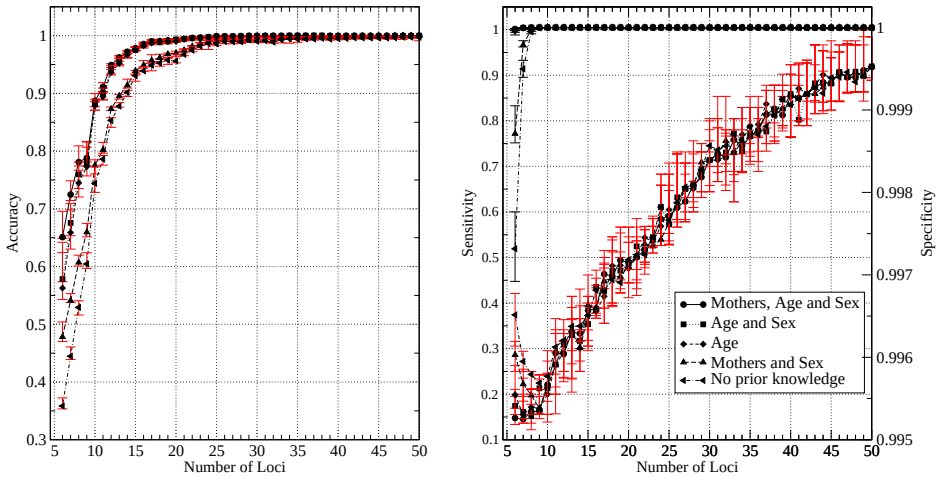


Figure 2: (Left) The accuracy of the reconstructed pedigrees is plotted as a function of the number of loci. The values are the median accuracy of ten randomly generated pedigrees of size 1000, reconstructed with different combinations of available prior knowledge. The error bars indicate the first and third quartile. The dataset has a sampling rate of 0.5 (1000 of 2000 individuals sampled) and has an overall typing error rate of 0.01. In addition, the first locus comprises one null allele ($p_n = 0.05$). The pedigree depth ranges from 5 to 9.
(Right) The sensitivity and specificity of the sibling calculation plotted again as a function of the number of loci.

parents are unsampled but a close relative (e.g. aunt or uncle) is sampled.

We also evaluated the performance of our full-sib heuristic directly. As we use this heuristic to reduce the pedigree space, we require a very small false positive rate. The sensitivity and specificity is plotted in Figure 2 (right). The critical values for this heuristic are the same in all cases, which explains that the observed sensitivity and specificity is quite independent of prior knowledge. However, as more non-siblings are tested without age data, one has to expect more false positives. Note that the second requirement, full-sibs must have common parents (4.1), gets also more significant the more genomic data is available. Hence, if age data is available or with larger amounts of genomic data, less conservative significance levels should be chosen.

6 Discussion

We have presented a fast algorithm for the pedigree reconstruction problem. The publicly available implementation is written in the C programming language and is platform-independent. It can be obtained under the GPL[1]. The genealogy of datasets with thousands of individuals is typically reconstructed in a few minutes. Due to the space constraints of

[1] `http://www.bioinf.uni-leipzig.de/Software/FRANz/`

this paper, we can only describe the core functionality of the software. Our implementation is flexible in incorporating additional data like age, sex, sampling locations, sub-pedigrees and allele frequencies. This was suggested in [Alm03] but not previously implemented in a publicly available software package. The reconstruction of large and deep pedigrees is highly accurate with only 15-20 polymorphic microsatellite loci (twice as many when age data are not available).

In [Alm03], some remaining challenges in the pedigree reconstruction problem were listed. These are the assumption that founders are unrelated, a better estimation of allele frequencies, linkage, support for typing errors or mutation, and estimation of the error of the reconstruction procedure. FRANz makes significant progress in the latter two tasks by combining the simulation procedure and the error model described in [KTM07] with the Simulated Annealing algorithm.

The error model was criticized in the literature because of its simplicity. Other programs explicitly model special kinds of errors, for example null alleles [Wan04, WCK06]. At typical error rates of 1%, however, the number of mismatching loci is low and a detailed modeling seems provide little benefit. More complex error models may be necessary for data with higher error rates, however.

Extensions of the LOD scores for linked loci when the linkage phase is known are proposed in [DRE88]. If the linkage phase and recombination rates are known with high accuracy, the incorporation of this prior information can significantly enhance the performance of the parentage assignments [DRE88]. However, in most cases the linkage phase is unknown and has to be estimated jointly. Loose linkage of a small fraction of markers should not seriously bias multilocus likelihood calculations [Mea91]. Tightly linked loci in contrast, such as neighboring SNPs, can be combined and treated as one single *pseudolocus*.

The pedigree likelihood function (4.3) is appealing because of its property being additive over the individuals. This allows very efficient construction algorithms and requires no prior information about the pedigree structure. However, if the genomic signal is low, the likelihood function will fail to construct the correct pedigree, especially when single-parents are considered. This is because the expected number of false positive single-parent arcs becomes large. Age data significantly reduces this effect. The same is true for our full-sib heuristic (4.2) in particular when large full-sib groups and both of their parents are sampled. Priors about the pedigree structure (the expected inbreeding rates, number of offspring, etc.) might further improve the performance. Information of this kind is oftentimes unknown *a priori*, however. In fact, these are parameters that one typically would like to infer from the reconstructed pedigrees.

Our incorporation of full-sib probabilities is a reaction to the concern expressed in [MT86] that non-excluded full-sibs of the offspring have on average a higher LOD score than the true father. To keep the pedigree likelihood function simple and efficient to calculate, we use only highly significant full-sibs to reduce the pedigree space. It seems possible to include more siblings than just the highly significant ones into the pedigree likelihood calculation without the risk of excluding the true parents. Since such “local” factors in the pedigree likelihood are also not very computationally intensive, we plan to explore this

avenue in future work.

Traditional parentage inference methods such as the one described in this paper have been criticized lately [HRB06]. Pedigrees are used to estimate parameters. If the genomic signal is not strong enough, many different pedigrees will have similar likelihood scores. Using only the best pedigree will thus introduce a bias. In [HRB06], it has been proposed to estimate the parameters of interest jointly with the pedigree. This, however, requires that the population's mating behaviour fits the implemented model. FRANz outputs all possible parent combinations, not only the ones of the maximum likelihood pedigree, as a starting point to investigate such a bias [DRE88, NMCP01].

Acknowledgements. We would like to thank Dean Jerry for the *P.monodon* dataset, the anonymous reviewers for many helpful comments and Elizabeth Thompson for elaborately answering our questions. This work has been supported by the European Commission NEST Pathfinder initiative on Complexity through project EDEN (Contract 043251).

References

[Alm03] A. Almudevar. A simulated annealing algorithm for maximum likelihood pedigree reconstruction. *Theor Popul Biol*, 63:63–75, Mar 2003.

[BBBE+04] A. Bonin, E. Bellemain, P. Bronken Eidesen, F. Pompanon, C. Brochmann, and P. Taberlet. How to track and assess genotyping errors in population genetics studies. *Mol. Ecol.*, 13:3261–3273, Nov 2004.

[Blo03] Michael S. Blouin. DNA-based methods for pedigree reconstruction and kinship analysis in natural populations. *Trends in Ecology & Evolution*, 18(10):503–511, October 2003.

[BW98] K.W. Broman and J.L. Weber. Estimation of pairwise relationships in the presence of genotyping errors. *Am. J. Hum. Genet.*, 63:1563–1564, Nov 1998.

[BWSD+07] T.Y. Berger-Wolf, S.I. Sheikh, B. DasGupta, M.V. Ashley, I.C. Caballero, W. Chaovalitwongse, and S.L. Putrevu. Reconstructing sibling relationships in wild populations. *Bioinformatics*, 23:49–56, Jul 2007.

[DRE88] B. Devlin, K. Roeder, and N.C. Ellstrand. Fractional paternity assignment: theoretical development and comparison to other methods. *TAG Theoretical and Applied Genetics*, 76(3):369–380, Sep 1988.

[HRB06] J.D. Hadfield, D.S. Richardson, and T. Burke. Towards unbiased parentage assignment: combining genetic, behavioural and spatial data in a Bayesian framework. *Mol. Ecol.*, 15:3715–3730, Oct 2006.

[JA03] A.G. Jones and W.R. Ardren. Methods of parentage analysis in natural populations. *Mol. Ecol.*, 12:2511–2523, Oct 2003.

[JBCS+00] L. Jin, M.L. Baskett, L.L. Cavalli-Sforza, L.A. Zhivotovsky, M.W. Feldman, and N.A. Rosenberg. Microsatellite evolution in modern humans: a comparison of two data sets from the same populations. *Ann. Hum. Genet.*, 64:117–134, Mar 2000.

[JBMW06] D.R. Jerry, Evansa B.S., Kenwayb M, and K. Wilson. Development of a microsatellite DNA parentage marker suite for black tiger shrimp Penaeus monodon. *Aquaculture*, 255(1-4):542–547, May 2006.

[KGV83] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by Simulated Annealing. *Science, Number 4598, 13 May 1983*, 220, 4598:671–680, 1983.

[KTM07] S.T. Kalinowski, M.L. Taper, and T.C. Marshall. Revising how the computer program CERVUS accommodates genotyping error increases success in paternity assignment. *Mol. Ecol.*, 16:1099–1106, Mar 2007.

[LMX06] T.H. Lin, E.W. Myers, and E.P. Xing. Interpreting anonymous DNA samples from mass disasters–probabilistic forensic inference using genetic markers. *Bioinformatics*, 22:298–306, Jul 2006.

[Mea91] Thomas R. Meagher. Analysis of Paternity within a Natural Population of Chamaelirium luteum. II. Patterns of Male Reproductive Success. *The American Naturalist*, 137(6):738–752, 1991.

[MSKP98] T.C. Marshall, J. Slate, L.E. Kruuk, and J.M. Pemberton. Statistical confidence for likelihood-based paternity inference in natural populations. *Mol. Ecol.*, 7:639–655, May 1998.

[MT86] Thomas R. Meagher and Elizabeth Thompson. The relationship between single parent and parent pair genetic likelihoods in genealogy reconstruction. *Theoretical Population Biology*, 29(1):87–106, February 1986.

[NMCP01] R. Nielsen, D.K. Mattila, P.J. Clapham, and P.J. Palsbll. Statistical approaches to paternity analysis in natural populations and applications to the North Atlantic humpback whale. *Genetics*, 157:1673–1682, Apr 2001.

[Off07] Federal Statistical Office. *Statistical Yearbook 2007 For the Federal Republic of Germany*. Number ISBN: 978-3-8246-0803-4. Federal Statistical Office, Wiesbaden, 2007.

[Pem08] J.M. Pemberton. Wild pedigrees: the way forward. *Proc. Biol. Sci.*, 275:613–621, Mar 2008.

[TH00] S.C. Thomas and W.G. Hill. Estimating quantitative genetic parameters using sibships reconstructed from marker data. *Genetics*, 155:1961–1972, Aug 2000.

[TH02] S.C. Thomas and W.G. Hill. Sibship reconstruction in hierarchical population structures using Markov chain Monte Carlo techniques. *Genet. Res.*, 79:227–234, Jun 2002.

[Tho76] E.A. Thompson. Inference of genealogical structure. *Social Science Information*, 15(477), 1976.

[TM87] E.A. Thompson and T.R. Meagher. Parental and sib likelihoods in genealogy reconstruction. *Biometrics*, 43:585–600, Sep 1987.

[VG06] J.F. Vouillamoz and M.S. Grando. Genealogy of wine grape cultivars: ”Pinot” is related to ”Syrah”. *Heredity*, 97:102–110, Aug 2006.

[Wan04] J. Wang. Sibship reconstruction from genetic data with typing errors. *Genetics*, 166:1963–1979, Apr 2004.

[WCK06] A.P. Wagner, S. Creel, and S.T. Kalinowski. Estimating relatedness and relationships using microsatellite loci with null alleles. *Heredity*, 97:336–345, Nov 2006.

Structure Local Multiple Alignment of RNA

Wolfgang Otto*[1] and Sebastian Will*[2] and Rolf Backofen[2]

[1] Bioinformatics, University Leipzig, D-04107 Leipzig
wolfgang@bioinf.uni-leipzig.de

[2] Bioinformatics, Albert-Ludwigs-University Freiburg, D-79110 Freiburg
{*will,backofen*}*@informatik.uni-freiburg.de*

Abstract: Today, RNA is well known to perform important regulatory and catalytic function due to its distinguished structure. Consequently, state-of-the-art RNA multiple alignment algorithms consider structure as well as sequence information. However, existing tools neglect the important aspect of locality. Notably, locality in RNA occurs as similarity of subsequences as well as similarity of only substructures. We present a novel approach for multiple alignment of RNAs that deals with both kinds of locality. The approach extends LocARNA by structural locality for computing all-against-all pairwise, structural local alignments. The final construction of the multiple alignments from the pairwise ones is delegated to T-Coffee. The paper systematically investigates structural locality in known RNA families. Benchmarking multiple alignment tools on structural local families shows the need for algorithmic support of this locality. The improvement in accuracy in special cases is achieved while staying competitive with state-of-the-art alignment tools across the whole Bralibase. LocARNA and its T-Coffee extended variant LocARNATE are freely available at `http://www.bioinf.uni-freiburg.de/Software/LocARNA/`.

1 Introduction

The recent discovery of the ubiquity and vast importance of regulatory and catalytic RNA in biological systems has radically changed our view on RNA [Cou02, Bar04, FW05]. This motivated a series of algorithmic developments in the area of multiple RNA alignment. RNA comparisons are challenging since both structure and sequence information have to be taken into account in order to successfully align RNAs with low sequence identities; pure sequence alignment is failing below of about 60% sequence identity. Spearheading this development are tools based on simultaneous alignment and folding like FoldAlignM [THG07], LARA [BKR07], and LocARNA [WRH+07]. However, these approaches neglect an important aspect of locality.

For RNA, one distinguishes two kinds of locality. First, similarity of RNAs can occur restricted to only corresponding subsequences; this form of locality is well known for sequence alignment. Even this locality is rarely supported by multiple alignment algorithms,

*Both authors contributed equally.

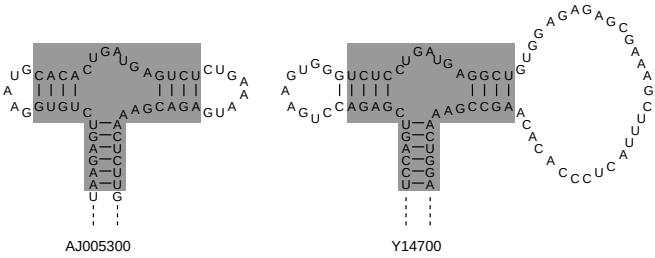


Figure 1: Two similar local substructures. Both hammerhead ribozymes $AJ005300$ and $Y14700$ differ globally. Nevertheless, they share a common functional motif (highlighted), which is structural local.

which thereby assume that the input sequences are accurately excised from their genomic context.

This assumption however does not suffice in face of the second kind of locality. Namely, RNA shows structural locality in the case where only substructures of several RNAs are similar, cf. Figure 1. Such corresponding local substructures can consist of several subsequences that are unconnected to each other at the sequence level. Then, these subsequences are connected only via the structure of the RNAs. An analogous view is that a local substructure consists of a subsequence, where certain subsequences are excluded (therefore called exclusions in the paper). For the simpler case of RNA alignment with fixed input structures, the algorithmic challenge posed by this kind of locality is solved in $O(n^5)$ [BW04].

Contribution In the paper, we show that structural locality plays an important role for RNA similarity and occurs in a number of known RNA families. To our knowledge this feature is for the first time analyzed across the Rfam, a database of known RNA family alignments[GJMM$^+$05].

Responding to this observation, we present the tool LocARNATE, which handles structural and sequence locality in the computation of multiple alignments of RNAs. To our knowledge it is the first multiple alignment approach that supports structural locality of RNAs. The paper describes the extension of the pairwise alignment algorithm of LocARNA[WRH$^+$07] by structural locality without increasing its theoretical complexity. This serves as a basis for the construction of multiple alignments, which is done here using T-Coffee[NHH00]. T-Coffee is chosen since it can do a consistency extension of the information from pairwise LocARNA alignments. Compared to a purely pogressive alignment strategy it is thereby able to avoid many of the typical mistakes. At the same time it respects the high-quality pairwise relation of the sequences derived by LocARNA.

Our theoretical results are supported 1.) by benchmarks using selected RNA sequences from the Rfam that show distinguished structural locality as well as 2.) by non-biased Bralibase 2.1 benchmarks. The Bralibase 2.1 is a compilation of true, hand-curated alignments for the purpose of assessing the accuracy of RNA alignment tools. [WMS06]

2 Preliminaries

An *(RNA) sequence* S is a word of $\Sigma = \{A, C, G, U\}$. We denote by A_i the ith symbol in A, by $A_{i..j}$ the subsequence from position i to j, and by $|A|$ the length of A. An *(RNA) structure* P *for* S is a set of *base pairs* (or *arcs*) $(i, j) \in \{1 \dots n\} \times \{1 \dots n\}$, $i < j$. A structure P is called *crossing* iff $\exists (i, i'), (j, j') \in P : i < j < i' < j'$. Otherwise it is called *non-crossing* or *nested*. In the paper, we assume that RNA structures are non-crossing. We define a partial ordering $\prec$ on pairs of natural numbers by $(i, i') \prec (j, j')$ iff $j < i < i' < j'$. Obviously, $\prec$ orders the base pairs of a structure P according to their nesting.

A *pairwise alignment* $\mathcal{A}$ *of two sequences* A *and* B is a subset of $[1..|A|] \cup \{-\} \times [1..|B|] \cup \{-\}$, where for all pairs $(i, j), (i', j') \in \mathcal{A}$ holds 1.) $i \leq i' \Rightarrow j \leq j'$ 2.) $i = i' \neq - \Rightarrow j = j'$, and 3.) $j = j' \neq - \Rightarrow i = i'$. We define the projections $\pi_1\mathcal{A} = \{i \neq - \mid \exists j : (i, j) \in \mathcal{A}\}$ and $\pi_2\mathcal{A} = \{j \neq - \mid \exists i : (i, j) \in \mathcal{A}\}$. An alignment $\mathcal{A}$ of A and B is called *global*, iff $\pi_1(\mathcal{A}) = [1..|A|]$ and $\pi_2\mathcal{A} = [1..|B|]$. A *sequence local motif of a sequence* A is a range $[i..j]$ for some $1 \leq i, j \leq |A|$. An alignment $\mathcal{A}$ of A and B is called *sequence local* iff $\pi_1\mathcal{A}$ is a sequence local motif for A and $\pi_2\mathcal{A}$ is a sequence local motif for B.

A *consensus structure* P *for an alignment* $\mathcal{A}$ *of* A *and* B is a pair (P_A, P_B) of a structure P_A for A and a structure P_B for B, such that 1.) for all $(i, j), (i', j') \in \mathcal{A}$ holds $(i, i') \in P_A$ iff $(j, j') \in P_B$, 2.) P_A contains only positions in $\pi_1\mathcal{A}$, and 3.) P_B contains only positions in $\pi_2\mathcal{A}$.

3 Locality

Structural Locality in Pairwise Alignments We distinguish sequence and structural locality. Adopting a graph theoretic view, sequence local motifs of a sequence A are sets of connected vertices in a graph $G_{\text{seq}} = (V, E)$, where $V = [1..|A|]$ and $E = \{(i, i+1) | 1 \leq i < |A|\}$. For a structure P of A, we define a *structural local motif for* A *and* P as a set of connected vertices in the *structure graph* $G_{\textit{struct}} = (V, E \cup P)$ *of* A *and* P. By this definition, structural local motifs correspond to "substructures", where the connection of bases can be either due to the backbone or due to bonds between base pairs.

An alignment $\mathcal{A}$ of two RNA sequences A and B is *structural local for consensus structure* (P_A, P_B) iff $\pi_1\mathcal{A}$ is a structural local motif for A and P_A as well as $\pi_2\mathcal{A}$ is a structural local motif for B and P_B.

To emphasize the orthogonality of sequence locality and structural locality, we require a (purely, i.e. sequence global) structural local motif for A to contain 1 and $|A|$, otherwise we may speak of a *sequence and structural local motif*. This extends to alignments.

For the later algorithmic treatment an alternative view of structural locality is required. Obviously, a structural local motif M for A and P (i.e. actually any motif $M \subseteq [1..|A|]$) is of the form $M = [i_1..i'_1] \cup \dots \cup [i_k..i'_k]$, i.e. it corresponds to a series of subsequences of A. The ranges $[i'_p + 1..i_{p+1} - 1]$ $(1 \leq p < k)$ are called *exclusions of* M, since we get M

by excluding them from the range $[i_1..i'_k]$. For an exclusion $[x..x']$ of a motif $M \subseteq [1..|A|]$ there is a base pair $(i, i') \in P, \{i, i'\} \in M$ where $(x, x') \prec (i, i')$. Denote the according to $\prec$ minimal such (i, i') as *bridge of* (x, x'). The following lemma gives an alternative characterization of structural locality, which will be used by our algorithm. An analogous statement is proven in [BW04].

Lemma 1 *A motif $M \subseteq [1..|A|]$ is structural local for A and P iff there is a bridge for each exclusion of M and each base pair in P is the bridge of at most one exclusion in M.*

Structural Locality in Multiple Alignments In contrast to our pairwise alignment definition, a multiple alignment, e.g. from Rfam, is usually given as a sequence of alignment columns. Thus it does not make explicit, which bases are locally aligned and which parts of the alignment are excluded from the structural local alignment due to their dissimilarity. However, structural locality can still be observed in such alignments.

For this purpose, multiple alignments are decomposed into their pairwise subalignments. Then, we assess structural locality by the presence of type I or type II exclusions in the pairwise alignments, which are defined as follows.

In a pairwise alignment $\mathcal{A}$, a *type I exclusion of length l and error rate e* is a subalignment (i.e. a continuous window) of l columns where 1.) in one sequence all columns contain a gap with the exception of at most $l \cdot e$ columns and 2.) no base in the l columns forms a base pair to any other base in the alignment.

A *type II exclusion in $\mathcal{A}$ of length l and error rate e* is a continuous window of l columns where 1.) more than $l \cdot e$ columns in one of the two sequences form a base pair with another base inside the window and 2.) for the other sequence, no bases inside of the window contribute to base pairs. Hence, type II exclusions correspond to the exclusion of substructures.

4 Structural Local Alignment

Based on the previous definitions, we will provide evidence for the ubiquity of structural locality in the results section. Here, we develop a structural local multiple alignment approach. The general workflow of the method is depicted in Figure 2.

Pairwise RNA Alignment We start our description by reviewing global and sequence-local pairwise alignment. [WRH+07] We compute an alignment $\mathcal{A}$ and a consensus structure $P = (P_A, P_B)$ of the given RNA sequences A and B that together maximize the score

$$\text{score}(\mathcal{A}, P) = \sum_{\substack{(i,k)\in P_A,(j,l)\in P_B \\ (i,j)\in\mathcal{A},(k,l)\in\mathcal{A}}} \tau(i,j,k,l) + \sum_{(i,j)\in\mathcal{A}_s} \sigma(A_i, B_j) - N_{\text{gap}}\gamma,$$

where N_{gap} denotes the number of gaps in $\mathcal{A}$ and $\tau(i,j,k,l)$ is the score contribution for matching the arcs (i,k) and (j,l). In LocARNA, $\tau(i,j,k,l)$ depends on the ensemble probabilities of the two arcs, as computed by McCaskill's algorithm [McC90], which is implemented in the Vienna RNA Package [HFS$^+$94]. This kind of scoring by base pair probabilities was introduced for the tool PMcomp/PMmulti [HBS04] as a much simplified scoring for Sankoff-style simultaneous alignment and folding [San85]. In LocARNA, very improbable arcs (below a given threshold) are forbidden in P, which significantly reduces the algorithmic complexity, making the approach applicable in practice. For details see [WRH$^+$07].

The score is efficiently maximized by a dynamic programming algorithm. First define a helper function

$$\mathrm{h}(M,k,l) = \max \begin{cases} M(k-1,l-1) + \sigma(A_j, B_l) \\ M(k-1,l) + \gamma \\ M(k,l-1) + \gamma \\ \max\limits_{k'l'} M(k'-1,l'-1) + D_{i\,j\,k'\,l'}. \end{cases}$$

The DP algorithm is now specified by the recursion

$$\begin{aligned} M_{i\,j}(k,l) &= \mathrm{h}(M_{i\,j},k,l) \\ D_{i\,j\,k\,l} &= M_{i\,j}(k-1,l-1) + \tau(i,j,k,l). \end{aligned}$$

Initialisation is simply by $M_{i\,j}(k,i) = M_{i\,j}(i,k) = k\gamma$. As given, the recursion computes the global alignment score. For the case of sequence local alignment, where we search the best alignment of subsequences, we modify the recursion for $i = 0$ and $j = 0$ by

$$M_{0\,0}(k,l) = \max(0, \mathrm{h}(M_{0\,0},k,l))$$

with initialization $M_{0\,0}(k,0) = M_{0\,0}(0,k) = 0$.

Pairwise Structural Local RNA Alignment Due to Lemma 1, certain exclusions are allowed in structural local alignments. Algorithmically, this distinguishes structural local alignments from sequence local or global alignments. The score is extended by adding one exclusion cost ϵ per exclusion. According to Lemma 1 (raised from motifs to alignments in a straightforward way), each exclusion in a local alignment has a bridge in the consensus structure and no two exclusions share the same bridge. This is enforced by counting the number of exclusions below each arc match in both sequences. For this purpose, we distinguish eight states, corresponding to eight different matrices. State NN means there is no exclusion for the arc match starting at (i,j). State XN means there is exactly one exclusion for this arc match in the first sequence, state NX is analogous for the second sequence, and state XX means there is exactly one exclusion in each of the sequences. In addition we introduce states for alignments that have exclusions immediately at the right

end of the first or the second sequence, which can therefore be extended. At the same time we keep track of the number of exclusions in the other sequence. This results in states ON,NO,OX,XO. The recursions are now given as follows. For $i > 0$ or $j > 0$,

$$
\begin{aligned}
M_{i\,j}^{NN}(k\,l) &= \mathrm{h}(M_{i\,j}^{NN}, k, l) \\
M_{i\,j}^{NX}(k\,l) &= \max(\mathrm{h}(M_{i\,j}^{NX}, k, l), M_{i\,j}^{ON}(k-1\,l) + \epsilon) \\
M_{i\,j}^{XN}(k\,l) &= \max(\mathrm{h}(M_{i\,j}^{XN}, k, l), M_{i\,j}^{ON}(k\,l-1) + \epsilon) \\
M_{i\,j}^{XX}(k\,l) &= \max(\mathrm{h}(M_{i\,j}^{XX}, k, l), M_{i\,j}^{ON}(k-1\,l) + \epsilon, M_{i\,j}^{NO}(k\,l-1) + \epsilon) \\
M_{i\,j}^{ON}(k\,l) &= \max(M_{i\,j}^{ON}(k-1\,l), M_{i\,j}^{NN}(k\,l)) \\
M_{i\,j}^{OX}(k\,l) &= \max(M_{i\,j}^{OX}(k-1\,l), M_{i\,j}^{NX}(k\,l)) \\
M_{i\,j}^{NO}(k\,l) &= \max(M_{i\,j}^{NO}(k\,l-1), M_{i\,j}^{NN}(k\,l)) \\
M_{i\,j}^{XO}(k\,l) &= \max(M_{i\,j}^{XO}(k\,l-1), M_{i\,j}^{XN}(k\,l)).
\end{aligned}
$$

Now, the scores for alignments enclosed by arc matches are read of these matrices as

$$D_{i\,j\,k\,l} = \max_{s \in \{NN,NX,XN,XX\}} M_{i\,j}^{s}(k-1\,l-1) + \tau(i,j,k,l).$$

Finally, the complete alignment score is obtained by the same recursion as for the global or purely sequence local case by evaluating $M_{0\,0}(k,l) = \mathrm{h}(M_{0\,0}, k, l)$ or $M_{0\,0}(k,l) = \max(0, \mathrm{h}(M_{0\,0}, k, l))$, respectively.

Note that the time complexity of $O(|A|^2|B|^2)$ and the space complexity of $O(|A||B|)$, both complexities given under the assumption of a fixed probability threshold, is not increased by supporting structural locality. In a practical implementation, the space for storing the M matrices can be limited to grow by a factor of only 4, since for the states NO,ON,OX,XO it is sufficient to store only matrix lines (ON,OX) or even single values (NO,XO) for evaluating the recursion.

The actual alignment is produced from the alignment matrices by traceback. In order to maintain the good space complexity, the M-matrices are recomputed on demand during the traceback phase; notably this does not increase the total complexity.

Finally note that, although the recursions are given for linear gap cost only, the extension to affine gap cost can be done in the way of Gotoh without increasing the complexity. The needed additional space is only linear in the lesser sequence length.

Multiple Alignment Using T-Coffee For constructing a (structural local) multiple alignment of sequences $A^{(1)}, \ldots, A^{(m)}$, we compute all pairwise (structural local) alignments as described above. From the pairwise alignments, we compile a library of alignment edges $(L_{kl})_{1 \le k,l \le m}$. L_{kl} contains an edge (i,j) with an alignment score dependent weight (between 1000 and 2000) iff in the pairwise alignment of $A^{(k)}$ and $A^{(l)}$, $A_i^{(k)}$ is aligned to $A_j^{(l)}$. All other edges get a weight of zero. This library is fed as primary library to T-Coffee. From this, T-Coffee computes an extended library by increasing the edge weights of pairwise edges that transitively fit to alignment edges to third sequences. The multiple

alignment is finally computed in a progressive fashion much like CLUSTALW, however using the extended library for scoring base similarity.

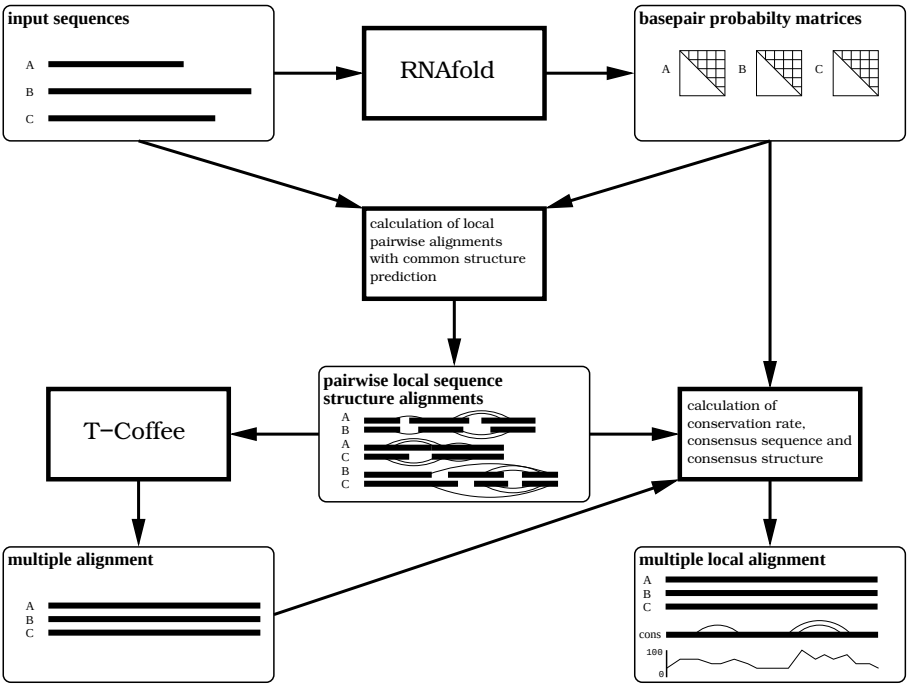


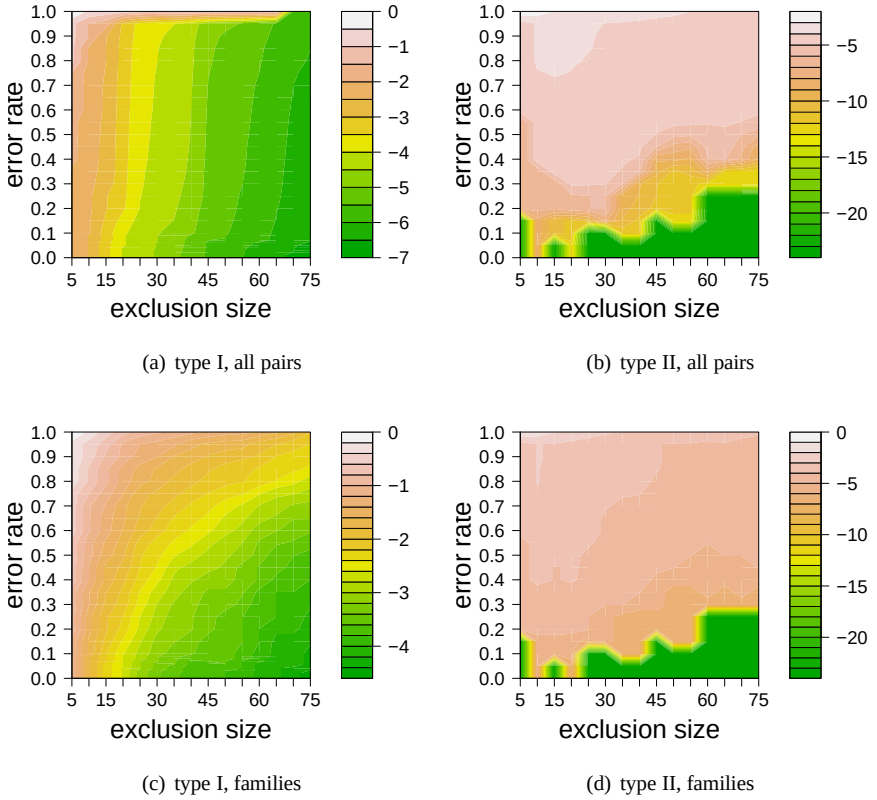
Figure 2: General workflow of the multiple alignment algorithm of LocARNATE

Local Motif of a Multiple Alignment Once a multiple alignment is constructed out of the (structural local) pairwise alignments, we can determine the structural local columns of this multiple alignment. This is done by assigning to each column a sum-of-pairs score over its pairwise alignment edges. There, each edge contributes with a weight of 1 if it got a non-zero weight in T-Coffee's primary library. As result, one gets a profile that reports a degree of locality for each column. Applying a fixed threshold, one finally extracts the local motif (subset of local columns) described by the alignment.

5 Results

Structural Locality in RNA Families In order to assess the demand for structural locality aware alignment, we analyze the occurrence of structural locality in the Rfam. We identify two reasons for structural locality. In alignments of two RNAs, type I exclusions of length l are subsequences of alignment columns where one of alignment strings consists of almost only gaps (with an error rate of e). Type II exclusions are subsequences, where

only one of the RNAs forms structure (again with error rate e). Our statistic of the Rfam seed sequences is shown in Figure 3.



(a) type I, all pairs

(b) type II, all pairs

(c) type I, families

(d) type II, families

Figure 3: Locality in the Rfam. We show the percentage of type I and type II exclusions for all pairs and for single families. Colors indicate frequency varying with exclusion size and allowed error rate.

LocARNATE: A Tool for Local Multiple Alignment Our structural locality aware multiple alignment approach for RNA, which combines an extended version of LocARNA with T-Coffee for constructing consistency based alignments, is implemented using C++ and Perl. It is available as the tool `locarnate` in the LocARNA software package.

Case Study Figure 4 gives an example for the identification of a local motif in a multiple local alignment.

Alignment Accuracy on the Bralibase The alignment accuracy of our approach is compared to two other programs Lara and FoldAlignM using the Bralibase benchmark. The

```
S1_AJ295015.1/58-1        .ACAGAGUCUGACAAA-------CCGUCACUGAAGACGUUCAA.C-UU----------------GCGUU----GAACAGAAACUCUGC
S2_AF170503.1/280-333     .GAAAGGUCUGUGCUU-------AGCACACUGACGA.GUUCCUGAAAU----------------G--------GAAC.GAAACCUUUU
S3_M83545.1/56-3          .CAUAAGUCUGGGCUA-------AGCCCACUGAUGA.GUCGCUGAAAU----------------G--------CGAC.GAAACUUAUG
S4_D00685.1/1-46          .........GCCAGACGU-G-GACCCGGCCUGAUGA.GUCCGAAA-------------------------------GGAC.GAAACAGUA.
S8_J02439.1/42-95         .UGUCCGU.AGUGGAUGU-G-UAUCCACUCUGAUGA.GUCCGAAA-------------------------------GGAC.GAAACGGAUG
S5_M17439.1/1-48          .........ACCGGAUGU-GCUUUCCGGUCUGAUGA.GUCCGUGA-------------------------------GGAC.GAAACAGGAC
S6_AJ536620.1/206-152     CC.ACCGUCGGAAAGUGU-G-CGCUUUCCCUGAUGA.GCCCAAAA-------------------------------GGGC.GAAACGGUAC
S7_Y12833.1/339-285       CC.GCUAUAUGGGGAUGU-G-UGUCCCUACUGACGA.GUUCAAAA-------------------------------GAAC.GAAAUAGUUA
S10_Y14700.1/133-53       ..UCCAGUCGAGACCUGAAG-UGGGUUUCCUGAUGA.GGCUGUGGAGAGAGCGAAA-GCUUUACUCCCACACAAGCC.GAAACUGGA.
S9_AJ247113.1/134-53      ..UCCAGUCGAGACCUGAAG-UGGGUUUCCUGACGA.GGCUGUGGAGAGAGCAAAUUGCUUUACUCCCGCACAAGCC.GAAACUGGA.
#=GC SS_cons              ..<<<<<<.<<<<<<.......>>>>>>.......<<<<.............................>>>>.>>>>>>.
#=GC conservation          .........*******-------***********.********-----------------------****.*********.
```

Figure 4: Example of a LocARNATE alignment of hammerhead ribozymes in stockholm-format. The line *#=GC conservation* marks conserved columns with a conservation rate of at least 0.5 by ***** (also highlighted in light gray) and excluded columns by - (darkgray). Note that the conserved columns correspond to the functional motive in Figure 1.

Bralibase consists of a collection of hand-curated multiple RNA alignments of 2 to 15 sequences each. We restrict the comparison to the most interesting subset of the Bralibase, namely alignments with less than 50% sequence identity. For the benchmark, one re-aligns the sequences of each such alignment with the candidate alignment tool and compares the result with the true alignment. The comparison is done by `compalignp`, as suggested for the Bralibase 2.1 benchmark [WMS06]. The resulting COMPALIGN score measures how accurately the generated alignment reproduces the given, true alignment - a score of 1.0 is optimal. This benchmark was done in the same way by Bauer et al.[BKR07], where Lara and FoldAlignM passed as the most successful sequence-structure alignment programs. The result of this test is reported in Figure 5.

An immediate, striking observation is that the tools LocARNATE and Lara seem to improve their accuracy with increasing number of sequences. The same effect is not seen for FoldAlignM, which is the only tool in this comparison that does not enjoy the consistency extension of T-Coffee. For 15 sequences, the comparably worse pairwise alignment of Lara is even outweighed by this effect and Lara is again on par with LocARNATE.

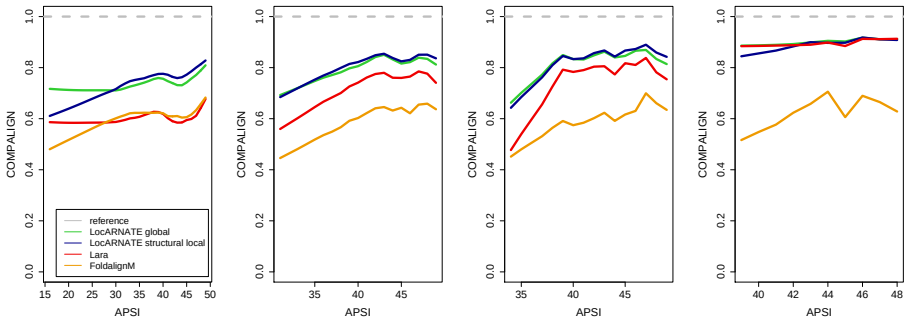


Figure 5: Benchmark on the Bralibase-fragment with $APSI < 50\%$ for alignments with 2,5,7, and 15 sequences (from left to right). The curves show the dependency between sequence identity (APSI) and alignment accuracy (COMPALIGN) for each of the four benchmarked algorithms.

Alignment Accuracy on Selected Rfam Alignments We select multiple subalignments of 7 sequences per alignment from the Rfam seed alignments. A benchmark set EI of 20 alignments with type I exclusions and a benchmark set EII with 10 type II exclusions is chosen. The sets EI (EII) are produced by each time selecting four pairwise alignments that have type I (type II) exclusions with length $l \geq 20$ ($l \geq 10$) and error rate $e \leq 0.25$ ($e \leq 0.6$), respectively. Of the eight sequences, we drop one at random. The, according to the Rfam, true alignment is obtained by projecting the corresponding Rfam family's seed alignment to the selected 7 sequences (deleting all only-gap columns). For each benchmark alignment, we align by LocARNATE with and without support of structural locality, Lara, and FoldAlignM. For each computed alignment, we obtain a COMPALIGN score by comparison with the true alignment. The results are shown in Figure 6.

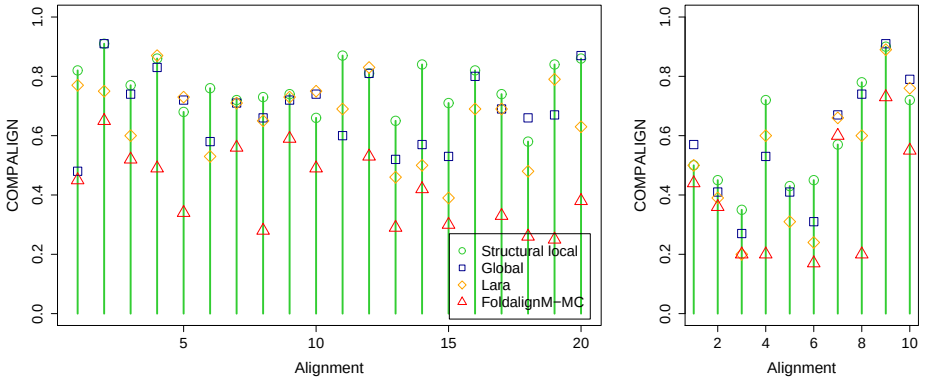


Figure 6: Benchmark on the alignment sets EI(left) and EII(right). Both sets consist of multiple alignments, each of seven sequences. EI contains type I exclusions, EII type II exclusions. The accuracy (COMPALIGN) is plotted for each single alignment and for each of the algorithms.

6 Conclusion

As we show by analysis of the whole Rfam database, structural locality is a wide spread feature of known RNA families. Structural locality is formalized by connectivity in the structure graph and via the notion of exclusions. Some families show strong structural locality, which motivates the development of special algorithmic support of this kind of locality. While current state-of-the art tools are not aware of this locality, we show that structural locality can be integrated into the tool LocARNA without increasing its complexity. By supporting this locality, the alignment accuracy for certain RNA families is increased significantly. We show by extensive benchmarks using the critical fragment of Bralibase 2.1 that the accuracy for families without obvious structural locality is not affected.

Acknowledgement Wolfgang Otto is supported by the Konrad-Adenauer-Stiftung as a scholarship holder. We thank the anonymous reviewers for their valuable comments.

References

[Bar04] David P. Bartel. MicroRNAs: genomics, biogenesis, mechanism, and function. *Cell*, 116(2):281–97, 2004.

[BKR07] Markus Bauer, Gunnar W. Klau, and Knut Reinert. Accurate multiple sequence-structure alignment of RNA sequences using combinatorial optimization. *BMC Bioinformatics*, 8:271, 2007.

[BW04] Rolf Backofen and Sebastian Will. Local Sequence-Structure Motifs in RNA. *Journal of Bioinformatics and Computational Biology (JBCB)*, 2(4):681–698, 2004.

[Cou02] Jennifer Couzin. Breakthrough of the year. Small RNAs make big splash. *Science*, 298(5602):2296–7, 2002.

[FW05] Martha J. Fedor and James R. Williamson. The catalytic diversity of RNAs. *Nat Rev Mol Cell Biol*, 6(5):399–412, 2005.

[GJMM+05] Sam Griffiths-Jones, Simon Moxon, Mhairi Marshall, Ajay Khanna, Sean R. Eddy, and Alex Bateman. Rfam: annotating non-coding RNAs in complete genomes. *Nucleic Acids Research*, 33 Database Issue:D121–4, 2005.

[HBS04] I. L. Hofacker, S. H. Bernhart, and P. F. Stadler. Alignment of RNA base pairing probability matrices. *Bioinformatics*, 20(14):2222–7, 2004.

[HFS+94] Ivo L. Hofacker, Walter Fontana, Peter F. Stadler, Sebastian Bonhoeffer, Manfred Tacker, and Peter Schuster. Fast Folding and Comparison of RNA Secondary Structures. *Monatshefte Chemie*, 125:167–188, 1994.

[McC90] J. S. McCaskill. The equilibrium partition function and base pair binding probabilities for RNA secondary structure. *Biopolymers*, 29(6-7):1105–19, 1990.

[NHH00] C. Notredame, D. G. Higgins, and J. Heringa. T-Coffee: A novel method for fast and accurate multiple sequence alignment. *Journal of Molecular Biology*, 302(1):205–17, 2000.

[San85] David Sankoff. Simultaneous solution of the RNA folding, alignment and protosequence problems. *SIAM J. Appl. Math.*, 45(5):810–825, 1985.

[THG07] Elfar Torarinsson, Jakob H. Havgaard, and Jan Gorodkin. Multiple structural alignment and clustering of RNA sequences. *Bioinformatics*, 23(8):926–32, 2007.

[WMS06] Andreas Wilm, Indra Mainz, and Gerhard Steger. An enhanced RNA alignment benchmark for sequence alignment programs. *Algorithms Mol Biol*, 1:19, 2006.

[WRH+07] Sebastian Will, Kristin Reiche, Ivo L. Hofacker, Peter F. Stadler, and Rolf Backofen. Inferring Non-Coding RNA Families and Classes by Means of Genome-Scale Structure-Based Clustering. *PLOS Computational Biology*, 3(4):e65, 2007.

Lightweight Comparison of RNAs Based on Exact Sequence-Structure Matches

Steffen Heyne, Sebastian Will, Michael Beckstette, Rolf Backofen

{heyne,will,mbeckste, backofen}@informatik.uni-freiburg.de

Albert-Ludwigs-University Freiburg,
Institute of Computer Science, Chair of Bioinformatics,
Georges-Koehler-Allee 106,
79110 Freiburg, Germany

Abstract: Specific functions of RNA molecules are often associated with different motifs in the RNA structure. The key feature that forms such an RNA motif is the combination of sequence and structure properties. In this paper we introduce a new RNA sequence-structure comparison method which maintains exact matching substructures. Existing common substructures are treated as whole unit while variability is allowed between such structural motifs.

Based on a fast detectable set of overlapping and crossing substructure matches for two nested RNA secondary structures, our method computes the longest colinear sequence of substructures common to two RNAs in $O(n^2m^2)$ time and $O(nm)$ space. Applied to different RNAs, our method correctly identifies sequence-structure similarities between two RNAs. The results of our experiments are in good agreement with existing alignment-based methods, but can be obtained in a fraction of running time, in particular for larger RNAs. The proposed algorithm is implemented in the program `expaRNA`, which is available from our website (www.bioinf.uni-freiburg.de/Software).

1 Introduction

Ribonucleic acids (RNAs) are associated to a large range of important cellular functions in living organisms. Moreover, recent findings show that RNAs can perform regulatory functions formerly assigned to proteins only. Likewise to proteins, these functions are often associated with evolutionary conserved motifs that contain specific sequence and structure properties. Examples for such regulatory RNA elements, whose function is mediated by sequence-structure motifs are selenocysteine insertion sequence (SECIS) elements [HWB96] (see Figure 1 for an example), iron-responsive elements (IRE)[HK96], different riboswitches [SP07], or internal ribosomal entry sites (IRES)[MLBM$^+$04]. Therefore, the detection of similar structural motifs in different RNAs is an important aspect for function determination and should be considered in pairwise RNA comparison methods. Although this problem is addressed in sequence-structure alignment methods, these approaches are often very time-consuming and do not necessarily preserve functionally important common substructures in the alignment [JLMZ02, JWZ95].

In this paper we propose a new lightweight, motif-based method for the pairwise comparison of RNAs. Instead of computing a full sequence-structure alignment, our approach efficiently computes a significant arrangement of sequence-structure motifs, common to two RNAs. For the

sake of algorithmic complexity and applicability in practice, we neglect higher order interactions like pseudoknots. This allows to describe sequence-structure motifs with nested RNA secondary structures, as shown in Figure 1.

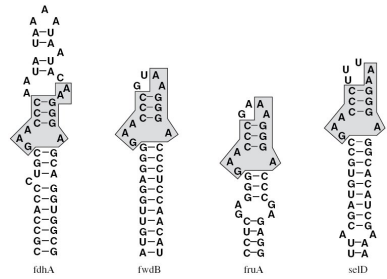


Figure 1: Putative SECIS elements in non-coding regions of *Methanococcus jannaschii* according to [WSPB97]. The indicated substructure represents a common substructure.

In [BS07] the authors presented a fast $O(nm)$ time and space algorithm for the identification of isolated common substructures for two given RNAs of lengths n and m with nested secondary structures. More precisely, their method identifies the complete, but overlapping set of exact common substructures. Our approach makes use of these common substructures and computes the longest colinear, non-overlapping sequence of substructures common to two RNAs in $O(n^2m^2)$ time and $O(nm)$ space. Herein after, we call this the LONGEST COMMON SUBSEQUENCE OF EXACT PATTERN MATCHINGS problem (LCS-EPM).

Related Work

Existing approaches addressing the sequence-structure comparison problem for RNA molecules can be distinguished by the given structural information and their representation. The standard alignment-based comparison approach employs the computation of edit distances between given RNA secondary structures [BMR95, JLMZ02]. In [Eva99] the author introduced the problem of finding the longest arc-preserving common subsequence (LAPCS). However, even for two *nested* RNA secondary structures, both problems remain NP-hard [BFRS03, LCJW02]. With some restrictions to the scoring scheme, the time complexity for determination of the edit distance can be lowered to polynomial time [JLMZ02].

If the nested secondary structure is represented as a tree, comparison methods exist for the edit distance between two ordered labeled trees [ZS89] as well as for the alignment of trees [JWZ95]. An improved version of the tree alignment method with extension to global and local forest alignments is given in [HTGK03] and implemented in the program RNAforester. The MIGAL approach extends the tree edit distance model by two new tree edit operations and is especially efficient due to its usage of different abstraction layers [AS05].

2 Exact Pattern Matchings and Longest Common Subsequences of Two RNA Secondary Structures

RNA is a macro molecule described formally by a pair $\mathcal{R} = (S, B)$ of a primary structure S and a secondary structure B. A *primary structure* S is a sequence of nucleotides $S = s_1 s_2 \dots s_n$

over the alphabet $\{A, C, G, U\}$. With $|S|$ we denote the length of sequence S. $S[i]$ indicates the nucleotide at position i in sequence S. With $S[i...j]$ we define the substring of S starting at position i until j for $1 \leq i < j \leq |S|$. A *secondary structure* B is a set of base pairs $B = \{(i, i') \mid 1 \leq i < i' \leq |S|\}$ over S, where each base takes part in at most one base pair. A secondary structure B is called *crossing* if there are two pairs $(i, i'), (j, j') \in B$ with $i < j < i' < j'$. Otherwise it is called *non-crossing* or *nested*.

For the definition of local RNA motifs, we represent an RNA $\mathcal{R} = (S, B)$ as undirected labeled graph $G = (V, E)$, called the *structure graph of* $\mathcal{R}$. Its set of vertices V is the set of positions in S, i.e. $V = \{1, \ldots, |S|\}$. Its set of edges E comprises all backbone bonds and all base pairs, i.e. $E = \{(i, i+1) \mid 1 \leq i < |S|\} \cup B$. An *RNA pattern in* $\mathcal{R}$ is a set of positions $\mathcal{P} \subseteq \{1, \ldots, |S|\}$, such that the *pattern graph for* $\mathcal{P}$ *in* G, defined as the subgraph $G' = (V', E')$ of G, where $V' = \mathcal{P}$ and $E' = \{(i, i') \in E \mid i \in \mathcal{P} \text{ and } i' \in \mathcal{P}\}$, is connected. By this definition, an RNA pattern corresponds to a local motif, i.e. a substructure that preserves the local neighborhood induced by backbone bonds and base pairs within a fixed secondary structure.

2.1 Exact Pattern Matchings of Two RNAs

In the following we consider two fixed, non-crossing RNAs $\mathcal{R}_1 = (S_1, B_1)$ and $\mathcal{R}_2 = (S_2, B_2)$. Their corresponding structure graphs are $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$, respectively. We will define an exact pattern matching as an *ordered matching of* V_1 *and* V_2, i.e. as a set $\mathcal{M} \subseteq V_1 \times V_2$, where for all $(p, q), (p', q') \in \mathcal{M}$ it holds that $p < p'$ implies $q < q'$ and $p = p'$ iff $q = q'$.

According to an ordered matching $\mathcal{M}$ of V_1 and V_2, we merge the graphs G_1 and G_2 into a *matching graph* $G_{\mathcal{M}} = (\mathcal{M}, E_{\mathcal{M}})$, where $E_{\mathcal{M}} = \{((p, q), (p', q')) \in \mathcal{M} \times \mathcal{M} \mid (p, p') \in E_1 \text{ and } (q, q') \in E_2\}$. A pair $(p, q) \in \mathcal{M}$ is called *admissible* if it satisfies the following conditions: (a) $S_1[p] = S_2[q]$ and (b) $\mathsf{STRUCT}_1(p) = \mathsf{STRUCT}_2(q)$. Here, function $\mathsf{STRUCT}_i(j)$ yields one of the three possible structural types for a nucleotide at position j in structure i: *single stranded*, *left paired*, or *right paired*. Further we want to preserve base pairs, i.e. $\forall (p, q), (p'q') \in \mathcal{PM} : (p, p') \in B_1 \Leftrightarrow (q, q') \in B_2$. Then, an *exact pattern matching* $\mathcal{PM}$ is an ordered matching where $G_{\mathcal{PM}}$ is connected, all $(p, q) \in \mathcal{PM}$ are admissible and all base pairs are preserved.

Hence, an exact pattern matching $\mathcal{PM}$ describes the matching between sets of positions in the two RNAs $\mathcal{R}_1$ and $\mathcal{R}_2$, namely the projections $\pi_1\mathcal{PM} = \{p | (p, q) \in \mathcal{PM}\}$ and $\pi_2\mathcal{PM} = \{q | (p, q) \in \mathcal{PM}\}$. Note that $\pi_1\mathcal{PM}$ and $\pi_2\mathcal{PM}$ are patterns in $\mathcal{R}_1$ and $\mathcal{R}_2$ respectively, i.e. in particular they correspond to the connected pattern graphs G_1^p and G_2^p. Note, although we claim an isomorphism on base pairs, $\mathcal{PM}$ does not necessarily describe an isomorphism on backbone edges in the pattern graphs G_1^p and G_2^p, since for $(p, q), (p', q') \in \mathcal{PM}$ where p and p' form an edge in G_1^p, q and q' do not necessarily form an edge in G_2^p. For details and proofs we refer to [BS07].

For our algorithm, we utilize only *maximal* exact pattern matchings, i.e. $\forall \mathcal{PM}' : \mathcal{PM} \subseteq \mathcal{PM}' \Rightarrow \mathcal{PM}' = \mathcal{PM}$. We abbreviate the term exact matching pattern by EPM. In the following, EPMs are always maximal. Similar to the minimal word size as e.g. used in BLAST [AMS+97], it is reasonable to consider a minimal size γ for EPMs. Hence, the set of all maximal exact pattern matchings $\mathcal{E}$ over two RNAs $\mathcal{R}_1$ and $\mathcal{R}_2$ is defined as

$$\mathbf{E}_{\gamma}^{1,2} = \{\, \mathcal{E} \mid \mathcal{E} \text{ is } \mathsf{EPM} \ \wedge \ |\mathcal{E}| \geq \gamma \,\}.$$

Note that each EPMis an arc-preserving common (but not longest common) subsequence as defined in [Eva99] for the LAPCS problem. However, the set of all EPMs is not a solution for the LAPCS problem since the combination of several EPMs is not necessarily arc-preserving. Since EPMs have in addition the above described properties, the detection of all EPMs is a computationally easy problem, compared to LAPCS, which is NP-complete even for nested sequences [BFRS03]. Using the dynamic programming approach described in [BS07], the set of all EPMs can be found in $O(nm)$ time and $O(nm)$ space, making this approach applicable for fast sequence-structure comparisons.

Now recall that each EPM is maximal. This implies that any two exact pattern matchings are disjoint and therefore a pair $(p, q) \in \mathcal{E} \in \mathbf{E}_{\gamma}^{1,2}$ is unique in $\mathbf{E}_{\gamma}^{1,2}$ and part of at most one EPM. Of course, two EPMs can overlap in one RNA and even in both RNAs. But this overlapping case implies that one exact pattern matching has to match to another region in the other RNA. The number of EPMs contained in $\mathbf{E}_{\gamma}^{1,2}$ is bounded by $n \cdot m$, with $n = |S_1|$ and $m = |S_2|$.

$\mathbf{E}_{\gamma}^{1,2}$ can be seen as a "library" of all common motifs between two RNAs, that can be utilized for a pairwise comparison method. In the following we describe the main aspects of our method based on common substructures. The EPMs in $\mathbf{E}_{\gamma}^{1,2}$ differ in their size and shape as well as in their structural positions in both RNAs. Taking two or several of these substructures into account they probably overlap or cross each other (see Figure 2). Clearly, a meaningful subset of common substructures excludes overlapping and crossing patterns. This guarantees that the backbone order of matched nucleotides as well as base pairs of the given RNAs are preserved. Compatible EPMs are called *non-crossing*.

Figure 2 shows an example of a possible set $\mathbf{E}_{\gamma}^{1,2}$. A "good" subset to describe the similarity between the two RNAs would probably exclude the EPMs indicated in red.

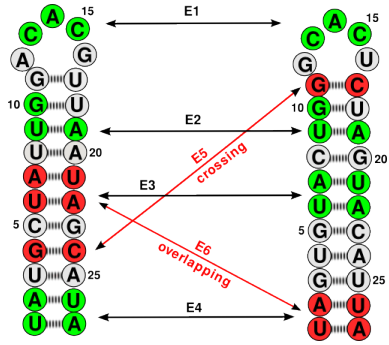


Figure 2: A possible set $\mathbf{E}_{\gamma}^{1,2}$ for two RNAs $\mathcal{R}_1, \mathcal{R}_2$. The set $\{\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3, \mathcal{E}_4\}$ can be used for a comparison, whereas $\{\mathcal{E}_5, \mathcal{E}_6\}$ should be excluded. $\mathcal{E}_5$ is crossing $\mathcal{E}_2$ and $\mathcal{E}_3$ whereas $\mathcal{E}_6$ is overlapping with $\mathcal{E}_3$ in $\mathcal{R}_1$ and with $\mathcal{E}_4$ in $\mathcal{R}_2$. Note, that not all possible EPMs are indicated.

2.2 A Global Comparison Approach: The Longest Common Subsequence of Exact Pattern Matchings (LCS-EPM)

The formulation of LCS-EPM is motivated by the fact that different RNA secondary structures share similar structural elements. Examples are shown in our result section for the comparison of thermodynamically folded as well as experimentally verified secondary structures. The knowl-

edge of such a "common core" of identical substructures in two RNAs is interesting for different tasks.

For our global approach we are interested in a *maximal* possible arrangement of substructures shared by two RNAs. If the motives are given in the form of exact pattern matchings, we call this the LCS-EPM problem. Basically, we search for a maximal combination of EPMs that form a common subsequence. Note that albeit the problem shares some similarity with LAPCS, it is restricted in such a way that an efficient solution is possible.

Formally, LCS-EPM is defined as follows. Given two nested RNAs $\mathcal{R}_1$, $\mathcal{R}_2$ and a set of exact pattern matchings $\mathbf{E}_\gamma^{1,2}$over these two RNAs, LCS-EPM is the problem of finding the longest common subsequence of S_1 and S_2 which preserves the exact pattern matchings in $\mathbf{E}_\gamma^{1,2}$; i.e. finding a mapping $\mathcal{M}_{\mathsf{EPM}} \subseteq V_1 \times V_2$ of maximal length such that:

1. for each pair $(p, q) \in \mathcal{M}_{\mathsf{EPM}}$ there exists one EPM in $\mathbf{E}_\gamma^{1,2}$:
   $\forall (p, q) \in \mathcal{M}_{\mathsf{EPM}} : \exists \mathcal{E} \in \mathbf{E}_\gamma^{1,2}$with $(p, q) \in \mathcal{E}$ and $\mathcal{E} \subseteq \mathcal{M}_{\mathsf{EPM}}$
2. $\mathcal{M}_{\mathsf{EPM}}$ is a bijective mapping and preserves the order of the nucleotides:
   $\forall (p, q), (p', q') \in \mathcal{M}_{\mathsf{EPM}} : p = p' \iff q = q', p < p' \iff q < q'$

Condition one claims that for any matched nucleotide, there exists one EPM in $\mathbf{E}_\gamma^{1,2}$. In addition, condition one includes that the complete EPM is part of $\mathcal{M}_{\mathsf{EPM}}$. The second condition ensures that the found subsequence is a common subsequence, i.e. a sequence which preserves the backbone order. Arcs or base pairs are induced by the EPMs itself.

2.2.1 Boundaries and Holes

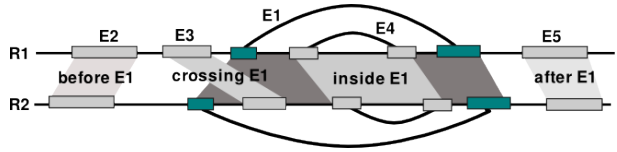


Figure 3: Ordering of exact pattern matchings relative to EPM $\mathcal{E}_1$ (indicated in green and dark gray). The cases *before, inside* and *after* do not violate the non-crossing condition. Only EPM $\mathcal{E}_3$ crosses $\mathcal{E}_1$. Note that an arc denotes a base pair within an EPM.

Our algorithm works by combining compatible EPMs. Given a single EPM of a library of EPMs, the relative order of the other EPMs can be distinguished as given in Figure 3. Formally, this is defined via the bounds and holes of a single EPM.

Bounds of EPMs The nucleotide positions of a pattern $\mathcal{P}$ of size k can be written as an increasing sequence. Similarly, an EPM $\mathcal{E}$ of size k over two RNAs is given with its corresponding patterns $\mathcal{P}_1$ in $\mathcal{R}_1$ and $\mathcal{P}_2$ in $\mathcal{R}_2$ and their increasing sequences $\mathcal{P}_1 = \langle p_1, p_2, ..., p_k \rangle$ and $\mathcal{P}_2 = \langle q_1, q_2, ..., q_k \rangle$.

In the view of the secondary structure, the elements (p_1, p_k) and (q_1, q_k) determine the outside borders of the EPM. Therefore we call them *outside-bounds* and write them as $\mathsf{OUT}_\mathcal{E} = \big\langle (p_1, p_k), (q_1, q_k) \big\rangle$. In the view of an arc-annotated sequence, we call (p_1, q_1) *left-outside-bounds* and (p_k, q_k) *right-outside-bounds* and denote them as $\mathsf{LEFT}_\mathcal{E}$ and $\mathsf{RIGHT}_\mathcal{E}$.

If an EPM contains base pairs, the structural shape is more complex and the outside-bounds are not sufficient to describe all structural borders. If not all enclosed nucleotides of a base pair are part of the EPM, then there exist two positions in each RNA that form an additional structural border *inside* the range of the outside-bounds. In addition, if a pattern contains several independent base pairs (e.g. in a multi-loop), there can be several such inside borders. The set of all such borders is called *inside-bounds* and is defined as $\mathsf{IN}_{\mathcal{E}} = \{\langle (p_i, p_{i+1}), (q_j, q_{j+1}) \rangle \mid p_{i+1} > p_i + 1 \Leftrightarrow q_{j+1} > q_j + 1\}$. Note, that *outside-bounds* always exists, whereas the set *inside-bounds* can be empty. For example, suppose an EPM that comprises only unbound nucleotides or a complete hairpin inclusive the closing bond. If an EPM consists of only one base pair in each sequence, then inside and outside bounds are identical. With the superscript index for the RNA we retrieve the bounds for a single RNA. For example $\mathsf{LEFT}^1_{\mathcal{E}} = p_1$.

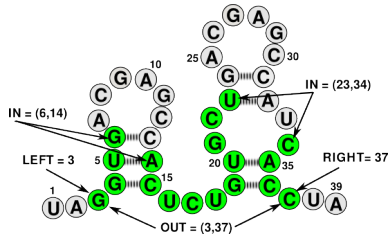


Figure 4: A pattern of an EPM in one RNA (green nucleotides). The different bounds are indicated.

Holes Holes are directly related to inside-bounds and describe the subsequences which are not part of the subsequence $S_i[\mathsf{LEFT}^i, \mathsf{RIGHT}^i]$ of an EPM. For a given EPM $\mathcal{E}$ with its set of inside-bounds $\mathsf{IN}_{\mathcal{E}}$, the set of holes with minimal size γ is defined as $\mathsf{HOLES}_{\mathcal{E}} = \{\ \langle (l^1, r^1), (l^2, r^2) \rangle \mid r^1 \geq l^1 + \gamma \ \wedge \ r^2 \geq l^2 + \gamma\ \}$. For each $h \in \mathsf{HOLES}_{\mathcal{E}}$ there exists a pair of inside-bounds with $\langle (l^1 - 1, r^1 + 1), (l^2 - 1, r^2 + 1) \rangle \in \mathsf{IN}_{\mathcal{E}}$. Clearly, a hole defines a substring $S_1[l^1, r^1]$ in the first RNA and a substring $S_2[l^2, r^2]$ in the second RNA. With γ we refer to the same size as indicated by $\mathbf{E}^{1,2}_{\gamma}$.

According to the length of the induced subsequences $S_i[l^i, r^i]$, we can sort all holes in one RNA. Let $h_i \in \mathsf{HOLES}_{\mathcal{E}_i}$ and $h_j \in \mathsf{HOLES}_{\mathcal{E}_j}$ two holes for any two $\mathcal{E}_i, \mathcal{E}_j \in \mathbf{E}^{1,2}_{\gamma}$. We define a partial ordering $h_i \preceq_{\mathsf{HOLES}} h_j$ in $\mathcal{R}_1$ if and only if h_i is of smaller size than h_j or of equal size in $\mathcal{R}_1$, i.e. $h_i \preceq_{\mathsf{HOLES}} h_j \Longleftrightarrow (r^1_i - l^1_i) \leq (r^1_j - l^1_j)$

2.2.2 Algorithm to Solve LCS-EPM

The crucial point and the main difference to alignment-based approaches as well as the LAPCS problem is that we have to treat a common substructure as whole unit. Therefore the final mapping has to include all pairs (p, q) of an EPM. Moreover, we want to compute the longest colinear sequence of EPMs which does not contain any crossing and overlapping EPMs.

The overall solution for LCS-EPM is constructed with a bottom-up approach from the comparison of substructures. This in principle requires a four-dimensional matrix, denoted as $D(i, j, k, l)$. Here the indices i, j refer to a substring $S_1[i, j]$ and the indices k, l to a substring $S_2[k, l]$, respectively. However, we can restrict ourselves to two-dimensional matrices using our notions of bounds and holes for an exact pattern matching $\mathcal{E}$ (see below). Finding non-crossing regions

relative to an EPM is achieved as follows: all nucleotides before $\mathsf{LEFT}_{\mathcal{E}}$, i.e $S_i[1, \mathsf{LEFT}^i_{\mathcal{E}} - 1]$, as well as all nucleotides after the $\mathsf{RIGHT}_{\mathcal{E}}$, i.e. $S_i[\mathsf{RIGHT}^i_{\mathcal{E}} + 1, |S_i|]$ fulfill the non-crossing condition. This means that any EPM with its outside-bounds $\mathsf{OUT}_{\mathcal{E}}$ in these regions is non-crossing relative to the considered EPM. Similar we handle EPMs that contain base pairs with the introduced notion of $\mathsf{HOLES}_{\mathcal{E}}$. All nucleotides inside these bounds are non-crossing, i.e. all EPMs which have outside-bounds within these regions satisfy the inside condition for non-crossing.

The recursion scheme for a dynamic programming algorithm is as follows. Any $\mathcal{E}$ is handled only once at its right-outside-bound $\mathsf{RIGHT}_{\mathcal{E}}$. The score of $\mathcal{E}$ is composed of the score *before* $\mathcal{E}$, given at the position $\mathsf{LEFT}_{\mathcal{E}} - 1$, plus the size of $\mathcal{E}$ itself, denoted by the function ω, plus possible scores of inside-bounds, given recursively by the computation of $\mathsf{HOLES}_{\mathcal{E}}$. This last recursion describes possible substructures and would lead to filling up a four-dimensional matrix. An improvement is achieved by ordering all holes according to the above introduced partial ordering $\preceq_{\mathsf{HOLES}}$. The recursion starts with one of the smallest holes and the remaining holes are computed in the order induced by $\preceq_{\mathsf{HOLES}}$. Hence, all necessary matrix entries exist, if an EPM with a hole is considered. Thus, only a two-dimensional matrix is necessary which leads directly to a quadratic space complexity. If two holes are of the same size, they can be treated in any order.

Suppose a given hole $h = \langle (l^1, r^1), (l^2, r^2) \rangle$, the following recursion scheme works for any $l^1 \leq j \leq r^1$ and $l^2 \leq l \leq r^2$. The best score is computed from treating the whole sequence as hole. With a standard traceback technique the set of EPMs that form the LCS-EPM are found.

$$\mathbf{D}(j,l) = \max \begin{cases} \mathbf{D}(j-1,l) \\ \mathbf{D}(j,l-1) \\ \mathbf{D}(i-1,k-1) + \mathbf{S}_{\mathcal{E}}, \\ \quad \text{if } \exists \mathcal{E} \in \mathbf{E}^{1,2}_{\gamma} \text{ with } \mathsf{RIGHT}_{\mathcal{E}} = (j,l) \text{ and} \\ \qquad \mathsf{LEFT}_{\mathcal{E}} = (i,k), i \geq l^1, k \geq l^2 \end{cases}$$

$$\mathbf{S}_{\mathcal{E}} = \omega(\mathcal{E}) \; + \sum_{h \in \mathsf{HOLES}_{\mathcal{E}}} \mathbf{D}(r^1, r^2) \quad \text{with } h = \langle (l^1, r^1), (l^2, r^2) \rangle$$

Complexity: The lengths of the sequences are $|S_1| = n$, $|S_2| = m$. The time complexity depends primarily on the number of holes. The set $\mathbf{E}^{1,2}_{\gamma}$ contains maximal $n \cdot m$ different holes which is estimated with $O(nm)$. The proof is omitted. For each hole, we fill a two-dimensional matrix with a size of at most $|S_1[l^1, r^1]| \leq |S_1| = n$ and $|S_2[l^2, r^2]| \leq |S_2| = m$. Consequently, for all holes we need $O(n^2 m^2)$ time as worst case complexity. For real RNAs, a more appropriate time complexity can be given as $O(H \cdot nm)$ with H as the number of holes, since $H \ll n \cdot m$. This also explains the fast running times of our examples. The space complexity is estimated with $O(nm)$ because the score of each hole is added to its EPM and the filled matrix is then discarded.

We summarize the complexity of solving the LCS-EPM problem as follows. Given two nested RNAs $\mathcal{R}_1 = (S_1, B_1)$ and $\mathcal{R}_2 = (S_2, B_2)$. The problem to determine the longest common subsequence of exact pattern matchings (LCS-EPM), including computation of $\mathbf{E}^{1,2}_{\gamma}$, is solvable in total $O(n^2\, m^2)$ time and $O(nm)$ space.

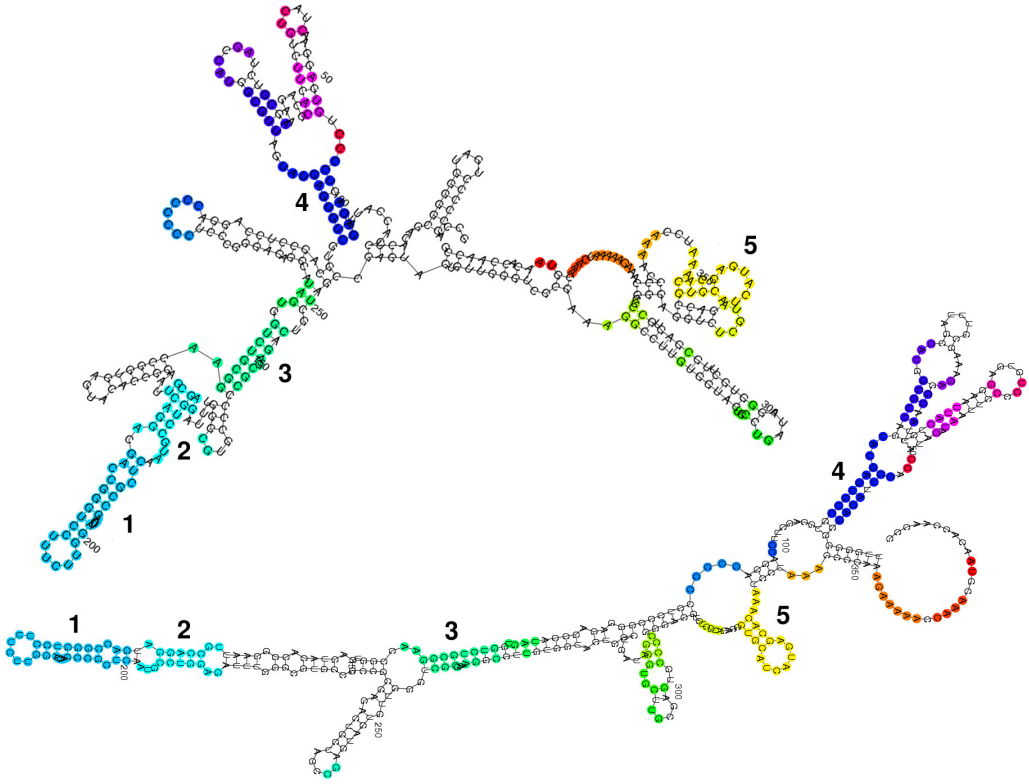


Figure 5: LCS-EPM approach applied to two Hepatitis C virus IRES RNAs. The colored nucleotides represent the found LCS-EPM with a length of 175 bases. Each EPM is shown in a different color. The numbers indicate the five largest EPMs from $\mathbf{E}_{\gamma}^{1,2}$. GenBank: D45172 (upper RNA), AF165050 (lower RNA)

3 Results

We implemented the algorithm for finding the longest common subsequence of exact RNA patterns in the tool **`expaRNA`** (exact pattern alignment of RNA). The algorithm to determine all EPMs was obtained from [BS07]. In order to analyze the performance of our approach, we have chosen two pairs of RNAs: **a)** Two IRES RNAs from Hepatitis C virus, which belong both to the Rfam family HCV_IRES for internal ribosomal entry sites (IRES) [GJMM$^+$05]. GenBank: AF165050 (bases 1-379) and D45172 (bases 1-391). The secondary structures were found via `RNAfold` [HFS$^+$94]. **b)** Two 16S rRNAs. The first RNA is from *Escherichia coli* and is 1541 bases long. The second RNA is from *Dictyostelium discoideum* and is 1551 bases long (GenBank codes: J01859 and D16466). The secondary structures were taken from the Comparative RNA Web (CRW) site [CSS$^+$02].

Table 1 shows the results for both pairs of RNAs. The structures with the indicated LCS-EPM can be seen in Figure 5 for the IRES RNAs and in Figure 6 for the 16S rRNAs. These figures are produced from `expaRNA` by interacting with the Vienna RNA Package [HFS$^+$94]. For the IRES RNAs, the numbers mark the five largest EPMs from the set $\mathbf{E}_{\gamma}^{1,2}$ and refer to the

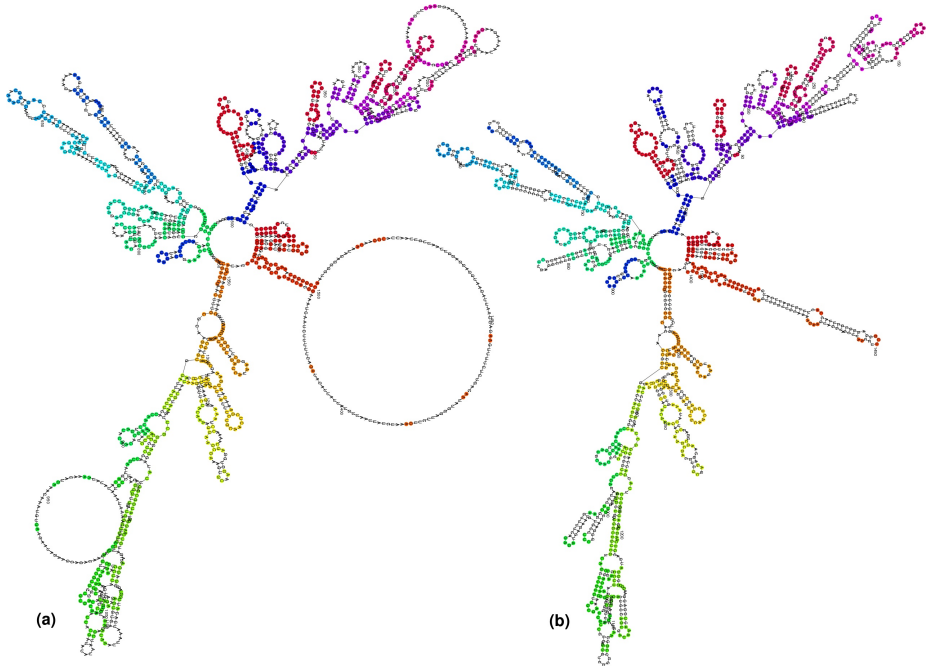


Figure 6: LCS-EPM approach applied two 16S ribosomal RNAs. The colored nucleotides represent the found LCS-EPM with a length of 875 bases. Each EPM is shown in a different color. (a) *D. discoideum* 16S rRNA (D16466), (b) *E. coli* 16S rRNA (J01859)

manually marked EPMs from the paper [BS07]. Our solution for LCS-EPM includes all of them automatically. An interesting detail is, for example, the included small blue hairpin in the top structure between number three and four. In the bottom RNA, this hairpin is opposite to the small yellow stem with number five, whereas in the top structure this stem is situated in another region. The 16S rRNAs comparison shows significant similarities in nearly all stem and loop regions. Note, for both examples the set $\mathbf{E}_{\gamma}^{1,2}$ was computed with $\gamma = 2$.

For the comparison of the results we have chosen `RNA_align` and `RNAforester`. The first method computes sequence structure alignments according to the general edit distance algorithm [JLMZ02]. The `RNAforester` program from [HTGK03] is build upon the tree alignment algorithm for ordered trees from [JWZ95] and extends it to calculate forest alignments. A comparison of these methods with our approach is possible on the number of common realized alignment edges. Therefore, we have first computed the alignments for both RNA pairs. Next, we have extracted from these alignments all positions with exact sequence structure matchings and determined the intersections with LCS-EPM. Note, the time for `expaRNA` in Table 1 includes the time to determine all EPMs for the two IRES RNAs $(0.44s)$ and for the two 16S rRNAs $(1.2s)$. The sequence coverage rate is averaged over both RNAs.

	IRES RNAs			16S rRNAs		
	#matches	coverage	time	#matches	coverage	time
expaRNA	175	45%	$0.97s$	875	57%	$16.9s$
RNA_align	192	50%	$62.1s$	861	56%	$1h\,35m$
RNAforester	128	33%	$5.41s$	847	55%	$7m\,25s$

comparison	IRES RNAs #common matches	16S rRNAs #common matches
expaRNA & RNA_align	159 (82.8%)	688(79.9%)
expaRNA & RNAforester	103 (80.5%)	700(82.6%)

Table 1: Comparison of the number of found exact matchings by LCS-EPM and two alignment methods. In the lower part, *#common matches* defines the number of identical matched nucleotides of expaRNA and the other methods.

4 Conclusion

We have developed a new algorithm for the pairwise sequence-structure comparison of RNAs and implemented it in the program expaRNA. Our approach utilizes common substructures for the detection of global similarities between two RNAs. We have applied the presented dynamic programming algorithm to two Hepatitis C virus IRES RNAs and two 16S ribosomal RNAs. In comparison to existing alignment methods, our approach found about 80% of their found exact matching edges. This also supports our assumption that "good" alignments realize a large number of common substructures. In addition, a complete gapped global alignment can be easily calculated, if the found LCS-EPM are used as anchor constraints. The impressive performance of expaRNA, in particular for large RNA molecules may allow its application as a fast prefiltering method for time-consuming RNA sequence-structure comparison approaches. This would allow genome-wide application of these methods.

5 Acknowledgment

This work has been supported by the Federal Ministry of Education and Research (BMBF grant 0313921 FORSYS/FRISYS) and the German Research Foundation (DFG grant BA 2168/2-1 SPP 1258).

References

[AMS+97] 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 Research*, 25(17):3389–402, 1997.

[AS05] Julien Allali and Marie-France Sagot. A new distance for high level RNA secondary structure comparison. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2(1):3–14, 2005.

[BFRS03] Guillaume Blin, Guillaume Fertin, Irena Rusu, and Christine Sinoquet. RNA sequences and the EDIT(NESTED,NESTED) problem. Technical Report RR-IRIN-03.07, IRIN, Université de Nantes, 2003.

[BMR95] V. Bafna, S. Muthukrishnan, and R. Ravi. Computing similarity between RNA strings. In *Proc. 6th Symp. Combinatorial Pattern Matching*, pages –16, 1995.

[BS07] Rolf Backofen and Sven Siebert. Fast Detection of Common Sequence Structure Patterns in RNAs. *Journal of Discrete Algorithms*, 5(2):212–228, 2007.

[CSS+02] J. J. Cannone, S. Subramanian, M. N. Schnare, J. R. Collett, L. M. D'Souza, Y. Du, B. Feng, N. Lin, L. V. Madabusi, K. M. Muller, N. Pande, Z. Shang, N. Yu, and R. R. Gutell. The Comparative RNA Web (CRW) Site: an online database of comparative sequence and structure information for ribosomal, intron, and other RNAs: Correction. *BMC Bioinformatics*, 3(1):15, 2002.

[Eva99] Patricia Anne Evans. *Algorithms and Complexity for Annotated Sequence Analysis*. PhD thesis, University of Alberta, 1999.

[GJMM+05] Sam Griffiths-Jones, Simon Moxon, Mhairi Marshall, Ajay Khanna, Sean R. Eddy, and Alex Bateman. Rfam: annotating non-coding RNAs in complete genomes. *Nucleic Acids Research*, 33 Database Issue:D121–4, 2005.

[HFS+94] Ivo L. Hofacker, Walter Fontana, Peter F. Stadler, Sebastian Bonhoeffer, Manfred Tacker, and Peter Schuster. Fast Folding and Comparison of RNA Secondary Structures. *Monatshefte Chemie*, 125:167–188, 1994.

[HK96] M. W. Hentze and L. C. Kuhn. Molecular control of vertebrate iron metabolism: mRNA-based regulatory circuits operated by iron, nitric oxide, and oxidative stress. *Proc. Natl. Acad. Sci. USA*, 93(16):8175–82, 1996.

[HTGK03] Matthias Höchsmann, Thomas Töller, Robert Giegerich, and Stefan Kurtz. Local Similarity in RNA Secondary Structures. In *Proceedings of Computational Systems Bioinformatics (CSB 2003)*, page 159. IEEE Computer Society, 2003.

[HWB96] A. Huttenhofer, E. Westhof, and A. Bock. Solution structure of mRNA hairpins promoting selenocysteine incorporation in Escherichia coli and their base-specific interaction with special elongation factor SELB. *RNA*, 2(4):354–66, 1996.

[JLMZ02] Tao Jiang, Guohui Lin, Bin Ma, and Kaizhong Zhang. A General Edit Distance between RNA Structures. *Journal of Computational Biology*, 9(2):371–88, 2002.

[JWZ95] T. Jiang, J. Wang, and K. Zhang. Alignment of trees - an alternative to tree edit. *Theoretical Computer Science*, 143(1):137–148, 1995.

[LCJW02] Guohui Lin, Zhi-Zhong Chen, Tao Jiang, and Jianjun Wen. The longest common subsequence problem for sequences with nested arc annotations. *J. Comput. Syst. Sci.*, 65(3):465–480, 2002.

[MLBM+04] Yvan Martineau, Christine Le Bec, Laurent Monbrun, Valerie Allo, Ing-Ming Chiu, Olivier Danos, Herve Moine, Herve Prats, and Anne-Catherine Prats. Internal Ribosome Entry Site Structural Motifs Conserved among Mammalian Fibroblast Growth Factor 1 Alternatively Spliced mRNAs. *Mol Cell Biol*, 24(17):7622–35, 2004.

[SP07] Alexander Serganov and Dinshaw J. Patel. Ribozymes, riboswitches and beyond: regulation of gene expression without proteins. *Nat Rev Genet*, 8(10):776–90, 2007.

[WSPB97] R. Wilting, S. Schorling, B. C. Persson, and A. Böck. Selenoprotein Synthesis in Archaea: Identification of an mRNA Element of Methanococcus jannaschii Probably Directing Selenocysteine Insertion. *Journal of Molecular Biology*, 266(4):637–41, 1997.

[ZS89] Kaizhong Zhang and Dennis Shasha. Simple Fast Algorithms for the Editing Distance Between Trees and Related Problems. *SIAM J. Comput.*, 18(6):1245–1262, 1989.