GI-Edition

Lecture Notes in Informatics

Ivo Grosse, Steffen Neumann,
Stefan Posch, Falk Schreiber and
Peter Stadler (Editors)

German Conference on Bioinformatics 2009

28th to 30th September 2009
Martin Luther University Halle-Wittenberg,
Germany

Proceedings

Ivo Grosse, Steffen Neumann, Stefan Posch, Falk Schreiber
and Peter Stadler (Editors)

German Conference on Bioinformatics 2009

28th to 30th September 2009
Martin Luther University Halle-Wittenberg, Germany

Gesellschaft für Informatik 2009

Lecture Notes in Informatics (LNI) - Proceedings
Series of the German Informatics society (GI)

Volume P-XXX

ISBN XXXXX
ISSN 1617-5468

Volume Editors
Prof. Dr. Ivo Große
Martin Luther University Halle-Wittenberg, Germany
Email: ivo.grosse@informatik.uni-halle.de
Dr. Steffen Neumann
Leibniz-Instituts für Pflanzenbiochemie (IPB) Halle, Germany
Email: steffen.neumann@ipb-halle.de
Prof. Dr. Stefan Posch
Martin Luther University Halle-Wittenberg, Germany
Email: stefan.posch@informatik.uni-halle.de
Prof. Dr. Falk Schreiber
Martin Luther University Halle-Wittenberg, Germany sowie
Leibniz-Institut für Pflanzengenetik und
Kulturpflanzenforschung (IPK) Gatersleben, Germany
Email: falk.schreiber@informatik.uni-halle.de
Prof. Dr. Peter F. Stadler
Universität Leipzig, Germany
Email: peter.stadler@bioinf.uni-leipzig.de

printed by Köllen Druck+Verlag GmbH, Bonn

Preface

This volume contains papers presented at the German Conference on Bioinformatics, GCB 2009, held in Halle (Saale), Germany, September 28-30, 2009. The German Conference on Bioinformatics 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) and the German Society of Chemical Technique and Biotechnology (Dechema) in cooperation with the German Society for Biochemistry and Molecular Biology (GBM).

Six leading scientists were invited to give keynote lectures to the conference. Svante Pääbo spoke on "A Neandertal Perspective on Human Origins", David Fell on "Building and Analyzing Genome-Scale Models of Metabolism", and Ewan Birney on "Ensembl and ENCODE: Understanding our genome and its variation". The focus of GCB 2009 — applying bioinformatics approaches to the field of plant science — is reflected in the keynotes "Phenotyping of Plants: Quantification of Structure and Function — Concepts and Infrastructure" by Ulrich Schurr and "Challenges of utilizing plant genetic resources" by Andreas Graner. The lecture by Theo van Hintum on "The role of bioinformatics in a global attempt to fight hunger" was given open to the general public.

The scientific program comprised 22 contributed talks presenting 18 regular and four short papers. These were selected from a total of 47 submissions after review by the program committee All regular papers and one short paper are collected in this proceedings. The remaining short papers and the 128 poster abstracts accepted to be presented at the poster session are published in a separate volume.

We like to thank all program committee members and all local organizers and helpers for the efforts. Thanks are also due to all contributing to and participating in GCB 2009 and the sponsors for their financial support of the conference. Special thanks to Matthias Hübenthal for compiling these proceedings.

August 2009,

Ivo Große, Martin Luther University Halle-Wittenberg
Steffen Neumann, IPB Halle
Stefan Posch, Martin Luther University Halle-Wittenberg
Falk Schreiber, University of Halle-Wittenberg
& IPK Gatersleben, Germany
Peter F. Stadler, University Leipzig

Program committee

Rolf Backofen . . . . . Albert-Ludwigs-University Freiburg, Germany
Sebastian Böcker . . . . . Friedrich-Schiller-University Jena, Germany
Thomas Dandekar . . . . . University of Wuerzburg Würzburg, Germany
Dirk Drasdo . . . . . INRIA Paris-Rocquencourt, France
Dmitrij Frishman . . . . . Technical University of Munich, Germany
Georg Fuellen . . . . . University of Greifswald, Germany
Robert Giegerich . . . . . Bielefeld University, Germany
Jan Gorodkin . . . . . University of Copenhagen, Denmark
Ivo Große . . . . . Martin-Luther-University of Halle-Wittenberg, Germany
Arndt von Haeseler . . . . . Center for Integrative Bioinformatics Vienna, Austria
Ivo Hofacker . . . . . University of Vienna, Austria
Dirk Holste . . . . . Vienna, Austria
Janet Kelso . . . . . MPI for Evolutionary Anthropology Leipzig, Germany
Ina Koch . . . . . Technical University of Applied Sciences Berlin, Germany
Oliver Kohlbacher . . . . . Tuebingen University, Germany
Jan Korbel . . . . . New Haven
Hans-Peter Lenhof . . . . . Saarland University Saarbrücken, Germany
Jens Léon . . . . . Bonn, Germany
David Marshall . . . . . Scottish Crop Research Institute Dundee, United Kingdom
Irmtraud Meyer . . . . . Vancouver, Canada
Burkhard Morgenstern . . . . . University of Göttingen, Germany
Axel Mosig . . . . . PICB Shanghai, China
Vince Moulton . . . . . University of East Anglia Norwich, United Kingdom
Steffen Neumann . . . . . Leibniz-Institut für Pflanzenbiochemie Halle, Germany
Sascha Ott . . . . . University of Warwick, United Kingdom
Yves van de Peer . . . . . Ghent University, Belgium
Stefan Posch . . . . . Martin Luther University Halle-Wittenberg, Germany
Stephen Rudd . . . . . Turku
Uwe Scholz . . . . . IPK Gatersleben, Germany
Dietmar Schomburg . . . . . Technische Universität Braunschweig, Germany
Falk Schreiber . . . . . MLU Halle-Wittenberg Halle/IPK Gatersleben, Germany
Michael Schröder . . . . . TU Dresden, Germany
Stefan Schuster . . . . . University of Jena, Germany
Joachim Selbig . . . . . University Potsdam, Germany
Korbinian Strimmer . . . . . University of Leipzig, Germany
Andrew Torda . . . . . University of Hamburg, Germany
Martin Vingron . . . . . MPI für molekulare Genetik Berlin, Germany
Bernd Weisshaar . . . . . Bielefeld University, Germany
Edgar Wingender . . . . . University of Goettingen, Germany
Ralf Zimmer . . . . . LMU Munich, Germany

Supporting scientific societies

Gesellschaft für Biochemie und Molekularbiologie e.V. (GBM)
http://www.gbm-online.de/

Gesellschaft für Chemische Technik und Biotechnologie e.V. (DECHEMA)
http://www.dechema.de/en/start_en.html

Gesellschaft für Informatik e.V. (GI), Fachgruppe 4.0.2
http://www.cebitec.uni-bielefeld.de/groups/fg402/

Non-profit sponsors

Leibniz Institute of Plant Biochemistry (IPB) Halle
http://www.ipb-halle.de/

Leibniz Institute of Plant Genetics and Crop Plant Research (IPK) Gatersleben
http://www.ipk-gatersleben.de/

Martin Luther University Halle-Wittenberg
http://www.uni-halle.de/

Commercial sponsors

BIOBASE GmbH
http://www.biobase-international.com/

ClusterVision BV
http://www.clustervision.com/

Genomatix Software GmbH
http://www.genomatix.de/

Illumina, Inc.
http://www.illumina.com/

Kapelan GmbH
http://www.kapelan-bioimaging.com/

KWS SAAT AG
http://www.kws.de/

Lehmanns Fachbuchhandlung GmbH
http://www.lob.de/

MEGWARE Computer GmbH
http://www.megware.com/

Microsoft Deutschland GmbH
http://www.microsoft.de

NVIDIA Corporation
http://www.nvidia.de/

SunGene GmbH
http://www.sungene.business.t-online.de/

TraitGenetics GmbH
http://www.traitgenetics.de/

transtec AG
http://www.transtec.de/

Table of Contents

2D Projections of RNA folding Landscapes

Ronny Lorenz, Christoph Flamm, Ivo L. Hofacker
{ronny, xtof, ivo}@tbi.univie.ac.at

Institute for Theoretical Chemistry
University of Vienna, Währingerstraße 17, 1090 Wien, Austria

Abstract: The analysis of RNA folding landscapes yields insights into the kinetic folding behavior not available from classical structure prediction methods. This is especially important for multi-stable RNAs whose function is related to structural changes, as in the case of riboswitches. However, exact methods such as *barrier tree* analysis scale exponentially with sequence length. Here we present an algorithm that computes a projection of the energy landscape into two dimensions, namely the distances to two reference structures. This yields an abstraction of the high-dimensional energy landscape that can be conveniently visualized, and can serve as the basis for estimating energy barriers and refolding pathways. With an asymptotic time complexity of $\mathcal{O}(n^7)$ the algorithm is computationally demanding. However, by exploiting the sparsity of the dynamic programming matrices and parallelization for multi-core processors, our implementation is practical for sequences of up to 400 nt, which includes most RNAs of biological interest.

1 Introduction

Structure formation of RNA molecules is crucial for the function of non-coding RNAs (ncRNAs) as well as for coding mRNAs with regulatory elements like riboswitches and attenuators. Some RNAs possess distinct meta-stable structures with different biological activity. A prime example are riboswitches that regulate gene expression depending on the presence or absence of a small ligand molecule. The pathogenity of viral agents like viroids is achieved by distinct meta stable structures of their 'genome'. While efficient RNA folding algorithms such as `mfold` [Zuk89] or the Vienna RNA package [HFS$^+$94] can be used to compute most equilibrium properties of an RNA molecule, they provide little information on folding dynamics. In this case one has to resort to either stochastic simulation of the folding process [FHMS$^+$01, IS00, GFW$^+$08] or analysis of the energy landscape based on enumeration or sampling. In particular the `barriers` program [FHSW02] is used to find all local minima in the energy landscape and their connecting transition states and energy barriers. The algorithm is based on a complete enumeration of all low-energy conformations [WFHS99] in the landscape and therefore scales exponentially with sequence length. In contrast, the `paRNAss` tool [GHR99] relies on sampling structures and clustering in order to detect multi-stable RNAs but gives no information on energy barriers.

The dynamic programming (DP) approach to RNA folding can also be extended to obtain

more information on the energy landscape: Cupal et al. [CHS96] proposed an algorithm that computes the density of states, i.e. the number of structures that fall into a particular energy bin, by extending the usual DP table into a third dimension corresponding to the energy bins. The `RNAbor` algorithm [FMC07], uses the base pair distance to a reference structure as the additional dimension and computes the optimal secondary structure as well as partition function for each distance class (δ-neighborhood).

Both approaches can be viewed as a one-dimensional projection of the high dimensional energy landscape, which however results into a drastic loss of information. Here, we describe a related method that employs the distances to two reference structures in order to compute a 2D projection which retains enough information to predict qualitative folding behavior and is easy to visualize. In particular, we define a κ, λ - *neighborhood* to be all secondary structures s with $d_{\text{BP}}(s_1, s) = \kappa$ and $d_{\text{BP}}(s_2, s) = \lambda$, where $d_{\text{BP}}(s_a, s_b)$ is the base pair distance of s_a and s_b, and proceed to compute minimum free energy (MFE) structure, as well as partition function and Boltzmann weighted structure samples for each κ, λ - *neighborhood* .

2 Methods

2.1 Minimum free energy algorithm

In the following we will write (i, j) to denote a base pair between the ith and jth nucleotide. A secondary structure s is regarded as a set of base pairs, the *base pair distance* between two structures is defined as $d_{\text{BP}}(s_1, s_2) = |s_1 \cup s_2| - |s_1 \cap s_2|$ and equals the number of base pairs present in either but not both structures. We will write $s[i, j] = \{(p, q) \in s : i \leq p < q \leq j\}$, to identify the substructure on the sequence interval $[i, j]$. $E(s)$ denotes the free energy of structure s.

For reference, we reproduce below the classic recurrences for MFE folding, which we will extend to the κ, λ - *neighborhood* in the following section. Note that the recursions employ an unambiguous decomposition of secondary structures as implemented in the Vienna RNA package [HFS+94].

$$
\begin{aligned}
F_{i,j} &= \min \left\{ F_{i,j-1}, \min_{i<k\leq j} F_{ik} + C_{k+1,j} \right\} \\
C_{i,j} &= \min \left\{ \mathcal{H}(i,j), \min_{i<k<l<j} C_{kl} + \mathcal{I}(i,j;k,l), \min_{i<u<j} M_{i+1,u} + \hat{M}_{u+1,j-1} + a \right\} \\
M_{i,j} &= \min \left\{ \min_{i<u<j} (u-i-1)c + C_{u+1,j} + b, \min_{i<u<j} M_{i,u} + C_{u+1,j} + b, M_{i,j-1} + c \right\} \\
\hat{M}_{i,j} &= \min \left\{ \hat{M}_{i,j-1} + c, C_{ij} + b \right\}
\end{aligned}
\tag{1}
$$

The upper triangular matrices $F_{i,j}$, $C_{i,j}$, $M_{i,j}$ and $\hat{M}_{i,j}$ contain the optimal folding energy on the sequence interval $[i, j]$, optimal energy given that (i, j) form a pair, given that i and j

reside in a multi-loop, and for multi-loop components with exactly one stem in the interval $[i, j]$, respectively. $\mathcal{H}(i, j)$ denotes the energy of a haipin-loop closed by (i, j), $\mathcal{I}(i, j, p, q)$ the energy of an interior-loop closed by (i, j) and (p, q). The parameters a, b, and c contain the penalties for closing a multi-loop, for adding a multi-loop component, and enlarging a multi-loop by one unpaired base. We use the energy parameters as tabulated by the Turner group [MSZT99]. After filling the matrices the MFE structure is found by backtracking in the usual manner.

2.2 Minimum free energy κ, λ-neighbors

For a given RNA sequence $\mathcal{S}$ and two fixed reference structures s_1 and s_2, the MFE version of the κ, λ - *neighborhood* algorithm computes energetically optimal structures $s_{\text{opt}}^{\kappa,\lambda} \in S^{\kappa,\lambda}$ where $S^{\kappa,\lambda} = \{s \mid d_{\text{BP}}(s_1, s) = \kappa \ \wedge \ d_{\text{BP}}(s, s_2) = \lambda\}$ is the κ, λ - *neighborhood* of reference structure s_1 and s_2. We extend the recursions (1) such that for each entry of the energy matrices F,C,M and $\hat{M}$ the optimal energy contribution of substructures $s[i, j]$ with $d_{\text{BP}}(s_1[i, j], s[i, j]) = \kappa$ and $d_{\text{BP}}(s_2[i, j], s[i, j]) = \lambda$ are computed. This leads to two additional dimensions in the energy matrices denoted by $F^{\kappa,\lambda}$,$C^{\kappa,\lambda}$,$M^{\kappa,\lambda}$ and $\hat{M}^{\kappa,\lambda}$. Since closing base pairs may lead to an increase of the base pair distance to both reference structures s_1 and s_2, additional decomposition constraints have to be introduced in the recurrences.

A hairpin loop closed by (i, j), for example, contributes to $C_{i,j}^{\kappa,\lambda}$ only if the substructure $s[i, j]$ consisting of the single pair $\{(i, j)\}$ only has distances κ and λ to the two substructures $s_1[i, j]$ and $s_2[i, j]$, respectively. Thus, we introduce the shorthand

$$\mathfrak{H}(i, j, \kappa, \lambda) = \begin{cases} \mathcal{H}(i, j) & \text{if } d_{\text{BP}}(s_1[i, j], \{(i, j)\}) = \kappa,\ d_{\text{BP}}(s_2[i, j], \{(i, j)\}) = \lambda \\ \infty & \text{else} \end{cases} \tag{2}$$

For non-hairpin loops, we introduce five terms $\delta_1^x - \delta_5^x$, where the superscript x is either 1 or 2, denoting the reference structure.

$$\begin{aligned} \delta_1^x(i, j) &= d_{\text{BP}}(s_x[i, j], s_x[i, j-1]) && (3) \\ \delta_2^x(i, j, u) &= d_{\text{BP}}(s_x[i, j], s_x[i, u-1] \cup s_x[u, j]) && (4) \\ \delta_3^x(i, j, p, q) &= d_{\text{BP}}(s_x[i, j], \{(i, j)\} \cup s_x[p, q]) && (5) \\ \delta_4^x(i, j, u) &= d_{\text{BP}}(s_x[i, j], \{(i, j)\} \cup s_x[i+1, u] \cup s_x[u+1, j-1]) && (6) \\ \delta_5^x(i, j, u) &= d_{\text{BP}}(s_x[i, j], s_x[u, j]) && (7) \end{aligned}$$

Each of the δ in eqs. (3-7) covers a distinct case in the energy minimization recursions, and denotes the minimal distance to the reference structure incurred when decomposing a substructure into two parts (since base pairs in the reference structure crossing the decomposition splitting positions j, u, p and q must be opened). For example $\delta_1^x(i, j)$ equals 1, if j is paired in the structure interval $s_x[i, j]$ of the reference structure s_x and 0 otherwise.

Decompositions into more than one substructure lead to additional combinatorial possibilities. They are taken into account by minimizing over $(\omega, \hat{\omega})$ pairs, where the sum $(\omega + \hat{\omega})$

reflects the residual of the base pair distance between the substructures and the references. Thus, the recursions to compute $E(s_{\text{opt}}^{\kappa,\lambda}) = F_{1,n}^{\kappa,\lambda}$ are:

$$
\begin{aligned}
F_{i,j}^{\kappa,\lambda} &= \min \left\{ \begin{array}{l} F_{i,j-1}^{\kappa-\delta_1^1(i,j),\lambda-\delta_1^2(i,j)}, \\ \min\limits_{i\le u<j} \min\limits_{\substack{\omega_1+\hat{\omega}_1=\kappa-\delta_2^1(i,j,u) \\ \omega_2+\hat{\omega}_2=\lambda-\delta_2^2(i,j,u)}} F_{i,u-1}^{\omega_1,\omega_2} + C_{u,j}^{\hat{\omega}_1,\hat{\omega}_2} \end{array} \right. \\
C_{i,j}^{\kappa,\lambda} &= \min \left\{ \begin{array}{l} \mathfrak{H}(i,j,\kappa,\lambda), \\ \min\limits_{i<p<q<j} \left\{ C_{p,q}^{\kappa-\delta_3^1(i,j,p,q),\lambda-\delta_3^2(i,j,p,q)} + \mathcal{I}(i,j,p,q) \right\}, \\ \min\limits_{i<u<j} \min\limits_{\substack{\omega_1+\hat{\omega}_1=\kappa-\delta_4^1(i,j,u) \\ \omega_2+\hat{\omega}_2=\lambda-\delta_4^2(i,j,u)}} \left\{ M_{i+1,u}^{\omega_1,\omega_2} + \hat{M}_{u+1,j-1}^{\hat{\omega}_1,\hat{\omega}_2} + a \right\} \end{array} \right. \\
M_{i,j}^{\kappa,\lambda} &= \min \left\{ \begin{array}{l} M_{i,j}^{\kappa-\delta_1^1(i,j-1),\lambda-\delta_1^2(i,j)} + c \\ \min\limits_{i\le u<j} \left\{ (u-i)\cdot c + C_{u,j}^{\kappa-\delta_5^1(i,j,u),\lambda-\delta_5^2(i,j,u)} + b \right\}, \\ \min\limits_{i\le u<j} \min\limits_{\substack{\omega_1+\hat{\omega}_1=\kappa-\delta_2^1(i,j,u) \\ \omega_2+\hat{\omega}_2=\lambda-\delta_2^2(i,j,u)}} \left\{ M_{i,u-1}^{\omega_1,\omega_2} + C_{u,j}^{\hat{\omega}_1,\hat{\omega}_2} + b \right\}, \end{array} \right. \\
\hat{M}_{i,j}^{\kappa,\lambda} &= \min \left\{ \begin{array}{l} C_{i,j}^{\kappa,\lambda} + b \\ \hat{M}_{i,j-1}^{\kappa-\delta_1^1(i,j),\lambda-\delta_1^2(i,j)} + c, \end{array} \right. \qquad (8)
\end{aligned}
$$

2.3 Time and memory complexity

Regarding the time complexity of the algorithm, a contribution of $\mathcal{O}(n^3)$, where n denotes RNA sequence length is implicit due to the underlying MFE folding algorithm. The additional degrees of freedom of the multi-loop decompositions in $C_{i,j}^{\kappa,\lambda}$ and $M_{i,j}^{\kappa,\lambda}$ increase the complexity by a factor of $\kappa \cdot \lambda$. The extension of the dynamic programming matrices by two further dimensions κ and λ additionally requires quadratically more effort. If the maximum distance values of κ and λ is limited to $\kappa \le d_1$ and $\lambda \le d_2$, the time complexity becomes $\mathcal{O}(n^3 \cdot d_1^2 \cdot d_2^2)$. Since the maximum number of base pairs on a sequence of length n is $\sim \frac{n}{2}$, the maximum achievable base pair distance between any two structures is bounded by n. Thus, the total asymptotic time complexity of the κ, λ - *neighborhood* algorithm results in $\mathcal{O}(n^7)$ for any distance boundaries d_1 and d_2.

A similar argument holds for the memory complexity which is $\mathcal{O}(n^2 \cdot d_1 \cdot d_2) = \mathcal{O}(n^4)$. Thus, the memory increase compared to regular MFE folding is $d_1 \cdot d_2 \le n^2$.

2.4 Partition function of the κ, λ - *neighborhood*

A modification of algorithm (8) to compute the partition function

$$Q^{\kappa,\lambda} = \sum_{s_x \in S^{\kappa,\lambda}} e^{-E(s_x)/kT} \tag{9}$$

for each κ, λ - *neighborhood* according to the algorithm of McCaskill et al. [McC90] is straight forward. The energy contributions are Boltzmann weighted and all sums/minimizations are replaced by products/sums. This can be done, as the recursions (8) perform unique decompositions and therefore already constitute a partitioning.

Since clustering of the complete secondary structure space into κ, λ - *neighborhoods* is a partitioning too, $\sum_{\kappa,\lambda} Q^{\kappa,\lambda} = Q$, where $Q = \sum_s e^{-E(s)/kT}$ is the partition function of the complete ensemble of all secondary structures.

The Boltzmann probabilities of a κ, λ - *neighborhood* in the complete ensemble and for a structure $s_x \in S^{\kappa,\lambda}$ inside a κ, λ - *neighborhood* become

$$P(S^{\kappa,\lambda}) = \frac{Q^{\kappa,\lambda}}{Q} \quad \text{and} \quad P(s_x \in S^{\kappa.\lambda}) = \frac{e^{-E(s_x)/kT}}{Q^{\kappa,\lambda}} \tag{10}$$

Stochastic backtracking yields a Boltzmann weighted sample of representative structures.

2.5 Sparse matrix approach and Parallelization

Some properties of the κ, λ - *neighborhood* can be used to improve the runtime and reduce memory requirements. Due to the definition of the κ, λ - *neighborhood* of two structures s_1 and s_2, there exist combinations of κ, λ distance pairs which do not contribute to any solution. For example, there is no κ, λ-neighbor with $\kappa + \lambda < d_{\text{BP}}(s_1, s_2)$. An increase or decrease of the base pair distance of a structure s to one of the reference structures implicitly changes the base pair distance to the other reference. In particular, if $d_{\text{BP}}(s_1, s_2) =$ even (resp. odd), then $\kappa + \lambda =$ even (resp. odd). This checkerboard-like pattern of the κ, λ - *neighborhood* roughly halves the number of entries in the extended dimensions κ and λ actually needed for the calculations. Furthermore, the maximum distance d_{max} to any reference structure in any substructure $s[i, j]$ of length $m = j - i + 1$ is constrained to $d_{\text{max}} < m$. These observations introduce sparsity in the dynamic programming matrices. Hence, two-dimensional matrices $F, C, M, \hat{M}$ with lists of triples, containing energy E, distance κ and distance λ at each matrix entry can be used. By iterating over the list instead of all κ, λ combinations, impossible structure formations are avoided.

Further runtime improvements can be obtained through parallelization by noting that all entries of the matrices F, C, M and $\hat{M}$ with $j - i =$ const. can be computed concurrently if the matrices are filled in diagonal order, see [FHS00].

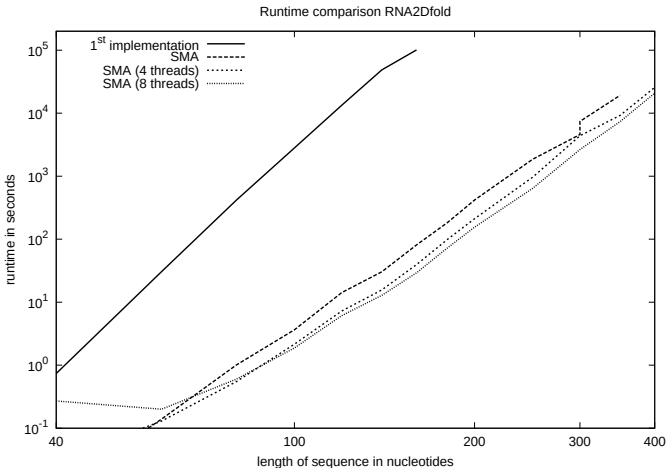


Figure 1: Runtimes of MFE calculation for the complete κ, λ - *neighborhood* . Timings are given for naïve approach (1^{st} implementation) and the sparse matrix approach (SMA) using 1, 4 and 8 threads on a dual quad-core Intel® Xeon® E5450 @3.00GHz with 32GB RAM. Runtimes are means of 15 random sequences. Reference structures used were the MFE structure and the open chain. With 8 processor cores, a sequence of 400 nt can be processed in about $5.8h$.

3 Results

3.1 Implementation

The partition function as well as the MFE version of the κ, λ - *neighborhood* algorithm was implemented in `ISO C` and will be available as a stand-alone program `RNA2Dfold` in one of the next releases of the Vienna RNA Package. A release candidate is available from `http://www.tbi.univie.ac.at/~ronny/RNA/`. The implementation provides most of the command line options of `RNAfold` such as different dangling end models and temperature. Given an RNA sequence $\mathcal{S}$ and two reference structures s_1 and s_2, `RNA2Dfold` computes for each κ, λ - *neighborhood* the MFE structure $s_{\text{opt}}^{\kappa,\lambda}$ and its free energy, the probabilities $P(S^{\kappa,\lambda})$, $P(s_{opt}^{\kappa,\lambda} \in S^{\kappa.\lambda})$, the probability of $s_{opt}^{\kappa,\lambda}$ in the complete ensemble and the Gibbs free energy $\Delta G^{\kappa,\lambda}$. The maximum values d_1 and d_2 with $\kappa \leq d_1$ and $\lambda \leq d_2$ can be specified by the user.

For parallelization we used *OpenMP* which allows efficient use of modern multi-core systems while requiring only small changes to the serial version of the source code. The performance gain from exploiting sparsity as well as parallelization is demonstrated in Fig. 1. The resulting speedups for 4 and 8 cores were 2.0 and 2.9, respectively. On modern multi-core systems `RNA2Dfold` can easily compute the MFE structures and partition functions for all κ, λ - *neighborhoods* for RNA sequences up to about 400 nt. This length range covers functional RNAs such as riboswitches and viroids.

3.2 2D Projection of the energy landscape

The probability densities and partition functions calculated by `RNA2Dfold` can be used for several secondary structure space analysis. One of the possible applications of the κ, λ - *neighborhood* algorithm is the prediction of metastable structure states and the detection of bi-stable RNA switches. Typically, the MFE structure is used as the first reference structure s_1. A meta-stable state, suitable as second reference structure s_2, can be obtained e.g. from a first run of `RNA2Dfold` using the open chain as second reference, and selecting s_2 from the $s_{\text{opt}}^{\kappa,\lambda}$. We note that this provides an alternative to the `paRNAss` approach for detecting RNA switches that avoids sampling errors. Computing the κ, λ - *neighborhood* of s_1 and s_2 and plotting the MFE values, probability densities and/or the Gibbs free energy of the partitioned landscape as a two dimensional height map reveals a qualitative picture of the roughness of the landscape. In the examples of Fig. 2 both RNAs can be clearly recognized as bi-stable switches. Molecules with more than 2 long-lived meta-stable states should exhibit additional minima in the interior of the height map. Furthermore, the height map yields a lower bound on the energy barrier between s_1 and s_2, and indicates the difficulty of refolding from s_1 to s_2 and vice versa.

3.3 A heuristic for finding non direct refolding paths

Most existing approaches utilize heuristics that consider only direct (minimal length) refolding paths between two states [MH98, FHMS$^+$01]. Since direct paths allow no detours, potentially stabilizing base pairs which are not in either of the two ground states cannot be formed. This can lead to intermediate structures with energetically unfavorable loop motifs and thus unnecessarily high energy barriers. In contrast, a refolding path with guaranteed minimal barrier can be obtained from the `barriers` program [FHSW02]. Since the approach is based of exhaustive enumeration of the energy landscape, it is limited to short sequences, typically less than a 100 nt.

The κ, λ - *neighborhood* can be used as base for various heuristics estimating the refolding path and energy barrier. Note however that taking the representatives $s_{\text{opt}}^{\kappa,\lambda}$ from a series of adjacent κ, λ-neighbors does usually not yield a continuous path of adjacent structures. Nevertheless, the height map already provides a lower bound for the height of the transition state and therefore for the energy barrier too. Direct path heuristics perform poorly when the two structures are far apart. Therefore, a natural extension is to construct an intermediate structure s_m, termed *mesh-point*, thus splitting the path construction problem between the two reference structures s_1 to s_2 into two path constructions from s_1 to s_m and from s_m to s_2. The problem is of course to find suitable mesh points, and the κ, λ - *neighborhoods* turn out to be an excellent starting point for this. The `Pathfinder` algorithm given below (3.1) connects such mesh points using the direct path heuristic from [FHMS$^+$01]. The method produces indirect paths, since the mesh points need not lie on a shortest path between s_1 and s_2.

After computing the κ, λ - *neighborhood* of the start (s_1) and target (s_2) structure, we test

Algorithm 3.1 Pseudo-code of the Pathfinder(s_a, s_b, iter) algorithm where s_a is the start structure, s_b is the stop structure, iter is the maximal number of iterations

```
MeshpointHeap ← ∅ /* initialize min-order mesh-point heap */
bestpath ← DirectPath(s_a, s_b) /* get best refolding path so far */
while Meshpoints available ∧ |MeshpointHeap| < m do
    s ← Meshpoint structure /* sample a mesh point structure */
    path ← DirectPath(s_a, s) + DirectPath(s, s_b)
    if Barrier(path) < Barrier(bestpath) then
        insert(MeshpointHeap, (s, path, Barrier(path)))
    end if
end while
if iter > 0 then
    for 0 ... m do
        (s, path) ← pop(MeshpointHeap)
        path ← Pathfinder(s_a, s, iter − 1) + Pathfinder(s, s_b, iter − 1)
        if Barrier(path) < Barrier(bestpath) then
            bestpath ← path
        end if
    end for
else
    (s, path) ← pop(MeshpointHeap)
    bestpath ← path
end if
return bestpath
```

as mesh-points all MFE structures $s_{\text{opt}}^{\kappa,\lambda}$ where $\kappa+\lambda \leq \gamma$ with a constant γ. This constraint limits the maximal deviation from a direct path and allows an adjustable exploration of the underlying energy landscape. Clearly, it is possible to recursively subdivide the problem further if required (see Pseudocode). With this simple approach the `Pathfinder` algorithm is able to find refolding paths with energy barriers very close or identical to those of an exhaustive search using the `barriers` program [FHSW02]. Results for an artificially designed RNA switch of 45 nt length revealed a barrier height of 10.7 kcal/mol (see Fig. 2A) which is the same as found with a barrier tree analysis. In contrast to that, a direct path generated according to [FHMS$^+$01] predicts an energy barrier of 13.33 kcal/mol. As mentioned before, the heightmap already provides a lower bound for the energy barrier. Here, direct paths are bounded by at least 13.3 kcal/mol, while indirect paths are bounded by 10.0 kcal/mol. Refolding between the aptamer- and non-aptamer fold of the *add*-riboswitch [RLGM07] (Fig. 2B) also shows the same energy barrier of 6.77 kcal/mol for both, Pathfinder and barrier tree analysis, while a direct path exhibits 7.28 kcal/mol.

4 Conclusion

We introduced a method for a unique partitioning of the RNA secondary structure space, in which structures are lumped together according to their base pair distances to two reference structures. In effect, this provides a 2D projection of the high-dimensional folding space. To overcome the high time complexity of $\mathcal{O}(n^7)$ our implementation exploits the sparseness of the dynamic programming matrices as well as *OpenMP* parallelization. The

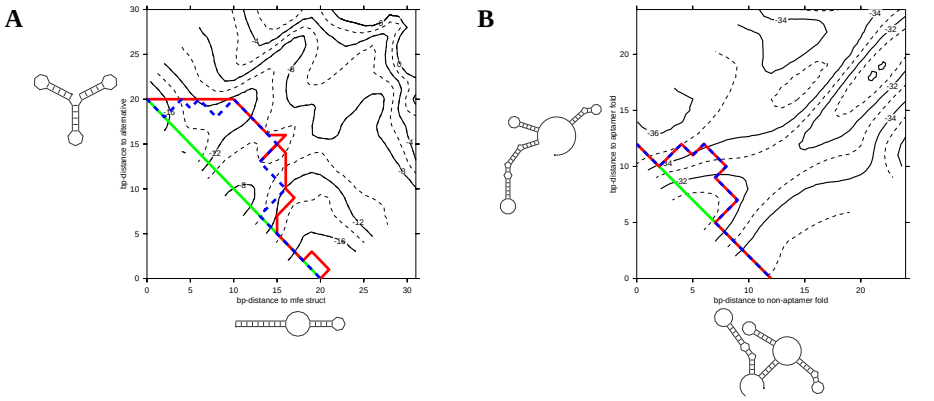


Figure 2: Gibbs free energy height map of all κ, λ - *neighborhoods* and projection of the refolding paths generated by `barriers` (red line) and the `Pathfinder` (blue dashed line) without recursive refinement. Mesh-points are taken from the κ, λ - *neighborhood* of both reference structures. **A**: MFE- and alternative structure of an artificial RNA switch with sequence `GGGCGCGGUUCGCCCUCCGCUAAAUGCGGAAGAUAAAUUGUGUCU` and meta stable structure conformations `(((((.....)))))((((((.....))))))((((((.....))))))` *(MFE structure)* and `((((((((((.....((((((.....))))))....))))))))))` *(alternative structure)*. **B**: Aptamer- and non-aptamer fold of an *add*-riboswitch [RLGM07]. In contrast to direct paths (straight diagonal green line), the `Pathfinder` solution is as good as the (optimal) solution generated by `barriers` in both cases. In **B**, identical refolding paths are obtained for `Pathfinder` and `barriers` analysis.

resulting program is fast enough to treat RNA molecules up to 400 nt which covers most biologically interesting cases such as riboswitches and viroids.

The κ, λ - *neighborhoods* provide both a qualitative picture of the energy landscape, as well as a convenient starting point for more detailed exploration. As an example we show that it can be used to suggest excellent intermediate nodes for the construction of refolding paths, resulting in a fast heuristic that often gives optimal results. Such heuristics are needed e.g. for kinetic folding strategies like `Kinwalker` [GFW$^+$08]. Furthermore, the height maps could provide the starting point for methods that recognize RNA switches or for coarse grained folding simulations.

Acknowledgments

This work has been funded, in part, by the Austrian GEN-AU projects “bioinformatics integration network III” and “non coding RNA”.

References

[CHS96] J. Cupal, I.L. Hofacker, and P.F. Stadler. Dynamic programming algorithm for the density of states of RNA secondary structures. *Computer Science and Biology*, 96(96):184–186, 1996.

[FHMS+01] C. Flamm, I.L. Hofacker, S. Maurer-Stroh, F. Stadler, and M. Zehl. Design of multistable RNA molecules. *RNA*, 7:254–265, 2001.

[FHS00] Martin Fekete, Ivo L. Hofacker, and Peter F. Stadler. Prediction of RNA base pairing probabilities using massively parallel computers. *J. Comp. Biol.*, 7:171–182, 2000.

[FHSW02] Christoph Flamm, Ivo L. Hofacker, Peter F. Stadler, and Michael T. Wolfinger. Barrier Trees of Degenerate Landscapes. *Z. Phys. Chem.*, 216:155–173, 2002.

[FMC07] E. Freyhult, V. Moulton, and P. Clote. Boltzmann probability of RNA structural neighbors and riboswitch detection. *Bioinformatics*, 23(16):2054, 2007.

[GFW+08] M. Geis, C. Flamm, M.T. Wolfinger, A. Tanzer, I.L. Hofacker, M. Middendorf, C. Mandl, P.F. Stadler, and C. Thurner. Folding kinetics of large RNAs. *Journal of Molecular Biology*, 379(1):160–173, 2008.

[GHR99] R. Giegerich, D. Haase, and M. Rehmsmeier. Prediction and visualization of structural switches in RNA. In *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, page 126, 1999.

[HFS+94] I.L. Hofacker, W. Fontana, P.F. Stadler, S. Bonhoeffer, M Tacker, and P. Schuster. Fast Folding and Comparison of RNA Secondary Structures. *Monatshefte f. Chemie*, 125:167–188, 1994.

[IS00] H Isambert and E D Siggia. Modeling RNA folding paths with pseudoknots: application to hepatitis delta virus ribozyme. *Proc Natl Acad Sci U S A*, 97(12):6515–20, Jun 2000.

[McC90] J. S. McCaskill. The equilibrium partition function and base pair binding probabilities for RNA secondary structure. *Biopolymers*, 29(6-7):1105–1119, 1990.

[MH98] S.R. Morgan and P.G. Higgs. Barrier heights between ground states in a model of RNA secondary structure. *Journal of Physics A-Mathematical and General*, 31(14):3153–3170, 1998.

[MSZT99] D.H. Mathews, J. Sabina, M. Zuker, and D.H. Turner. Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure. *Journal of Molecular Biology*, 288(5):911–940, 1999.

[RLGM07] R. Rieder, K. Lang, D. Graber, and R. Micura. Ligand-induced folding of the adenosine deaminase A-riboswitch and implications on riboswitch translational control. *Chembiochem*, 8(8), 2007.

[WFHS99] S. Wuchty, W. Fontana, I. L. Hofacker, and P. Schuster. Complete suboptimal folding of RNA and the stability of secondary structures. *Biopolymers*, 49(2):145–165, February 1999.

[Zuk89] M. Zuker. On finding all suboptimal foldings of an RNA molecule. *Science*, 244(4900):48–52, April 1989.

Graph-Kernels for the Comparative Analysis of Protein Active Sites

Thomas Fober*, Marco Mernberger*, Ralph Moritz, Eyke Hüllermeier
Department of Mathematics and Computer Science
Marburg University, Germany
{thomas,mernberger,moritz,eyke}@mathematik.uni-marburg.de

Abstract: Graphs are often used to describe and analyze the geometry and physicochemical composition of biomolecular structures, such as chemical compounds and protein active sites. A key problem in graph-based structure analysis is to define a measure of similarity that enables a meaningful comparison of such structures. In this regard, so-called kernel functions have recently attracted a lot of attention, especially since they allow for the application of a rich repertoire of methods from the field of kernel-based machine learning. Most of the existing kernel functions on graph structures, however, have been designed for the case of unlabeled and/or unweighted graphs. Since proteins are often more naturally and more exactly represented in terms of node-labeled and edge-weighted graphs, we propose corresponding extensions of existing graph kernels. Moreover, we propose an instance of the substructure fingerprint kernel suitability for the analysis of protein binding sites. The performance of these kernels is investigated by means of an experimental study in which graph kernels are used as similarity measures in the context of classification.

1 Introduction

The functional analysis of proteins is a key research problem in the life sciences and a main prerequisite for resolving the proteome and interactome of living cells, tissues and organisms. Since improved technology has led to an increased number of known protein structures, structure-based prediction of protein function has now become a viable alternative to classical sequence-based prediction methods. In fact, structure-based approaches complement sequence-based methods in a reasonable way, as it is well-known that functional similarity does not necessarily come along with sequence similarity [GMB96].

Prediction of protein function can be seen as a classification problem. In machine learning, a large repertoire of classification methods has been developed, most of them relying, in one way or the other, on a kind of similarity measure between the objects to be classified. What is needed, therefore, is a measure of similarity between protein structures. More specifically, our focus in this paper will be on the special case of *protein binding sites* derived from crystal structures. To model such structures in a formal way, we resort to a graph representation which is able to capture the most important geometrical and physicochemical properties of a binding site.

For a long time, graphs have been used in chemoinformatics for the modeling of chemical

compounds [BJ00]. In bioinformatics, they are becoming more and more important, too, due to their general versatility in modeling complex structures such as proteins or interaction networks [BL04]. It is hence not surprising that a number of methods has been developed for comparing graphs representing protein structures (e.g. [JIDG03; WHKK07; FMKH09]), and for computing related similarity measures, for example based the concepts of maximum (minimum) common subgraph (supergraph) [RGW02; RW02] or graph edit distance [NB07].

In this context, so-called *kernel functions* (on graphs) have attracted increasing attention in recent years[Gär03]. Here, the term 'kernel' refers to a class of functions that fulfill certain mathematical properties and can typically be interpreted as similarity measures. These functions are especially attractive as they can be used as a 'plug-in' for every kernel-based machine learning method. In other words, as soon as a kernel function has been defined on a certain class of objects, the related domain becomes amenable to these methods.

The random walk kernel [Gär03] and the shortest path kernel [Bor07] are among the most prominent graph kernels that have been used in the fields of bio- or chemoinformatics. However, as they have originally been defined for unweighted graphs, they are not immediately applicable to the case of graphs modeling protein binding sites. In fact, as will be explained in more detail in Section 2, binding sites are more naturally modeled in terms of graphs with node labels and edge weights, and a representation ignoring labels and weights would come along with an unacceptable loss of information. In Section 3, we therefore extend the aforementioned kernel functions to the case of node-labeled and edge-weighted graphs. Besides, we make use of the *substructure fingerprint representation* [FHZ06] to define a class of kernels for protein binding sites. An experimental comparison of these graph kernels will be presented in Section 4 and discussed in Section 5.

2 Modeling Protein Binding Sites

To model protein binding sites as graphs, we build upon CavBase [SHK01; SKK02], a database developed for the purpose of identifying and extracting putative protein binding sites from structural data deposited in the protein database (PDB) [BWF$^+$00]. CavBase detects putative binding sites as cavities on the surface of proteins by using the LIGSITE algorithm [HRB97]. The geometry of a protein binding site is internally represented by a set of *pseudocenters*, spatial points that represent the physico-chemical properties of a surface patch within the binding site. Currently, CavBase uses seven types of pseudocenters (donor, acceptor, donor-acceptor, pi, aromatic, aliphatic and metal) that account for different types of possible interactions between residues of the binding site and the substrate of the protein. These pseudocenters are derived from the amino acid composition of the binding site.

As a natural way to model such structures, we make use of node-labeled and edge-weighted graphs. Nodes correspond to pseudocenters and are thus labeled with the pseudocenter type. On average, a graph representation of a binding pocket has around 100 nodes, though graphs with several hundred nodes and some extremes with thousands of nodes do exist.

Edges are weighted by the Euclidean distance between the pseudocenters and thus capture the geometry of the binding site. To reduce the complexity of the representation and increase algorithmic efficiency, we use an approximate representation in which edges exceeding a certain length are ignored; in this regard, a threshold of 11 Angström has proved to be a reasonable choice [FMKH09]. Despite this approximation, our representation will produce graphs that are rather dense, as approximately 20 percent of all pairs of nodes are connected by an edge. Consequently, the graphs have a large number of cycles. Indeed, a cycle-free representation will normally not be able to reproduce the geometry of a binding site in an accurate way. As will be seen later on, this property leads to problems for certain types of kernel functions.

Formally, a node-labeled and edge-weighted graph will be denoted by $G = (V, E, l_V, l_E)$, where V is a finite set of nodes and $E \subseteq V \times V$ a set of edges. Moreover, $l_V : V \to \mathcal{L}_V$ is a function that maps each node to one among a finite set of labels $\mathcal{L}_V$. Likewise, $l_E : E \to \mathbb{R}_+$ is a mapping that assigns weights to edges. We define the size of a graph in terms of its number of nodes $|V|$. The adjacency matrix of a graph G will be denoted by A.

We note that, since our edges are undirected, it would be more correct to use a subset instead of a tuple representation. For convenience, however, we stick to the simpler tuple notation, with the implicit understanding that $(u, v) \in E$ implies $(v, u) \in E$ and $l_E((u, v)) = l_E((v, u))$.

3 Kernels for Node-Labeled and Edge-Weighted Graphs

Let $\mathcal{G}$ be a set of objects, in our case graphs. A $\mathcal{G} \times \mathcal{G} \to \mathbb{R}$ mapping k is called kernel if it is symmetric and positive definite, that is, $k(x, y) = k(y, x)$ for all $x, y \in \mathcal{G}$ and

$$\sum_{i,j=1}^{m} c_i c_j k(x_i, x_j) \geq 0$$

for all $m \in \mathbb{N}$, $\{c_1, \ldots, c_m\} \subseteq \mathbb{R}$, and $\{x_1, \ldots, x_m\} \subseteq \mathcal{G}$.

A generic way to define similarity measures for complex objects, such as graphs, is to use decomposition techniques, that is, to decompose a complex object into a set of simple substructures of a specific type, and to reduce the comparison to the level of these substructures. The idea is that, for such substructures, the definition of adequate similarity measures is less difficult and, hopefully, the computation more efficient. Therefore, graph kernels often belong to the class of *R-convolution kernels*, a special type of kernel especially suitable for composite objects in a discrete space. Generally, an R-convolution kernel $k : \mathcal{G} \times \mathcal{G} \to \mathbb{R}$ can be expressed in the following from:

$$k(G, G') = \sum_{g \in R^{-1}(G)} \sum_{g' \in R^{-1}(G')} \kappa(g, g') \ , \tag{1}$$

where $R^{-1}(G)$ denotes a decomposition of G into substructures, and κ is a kernel defined on such substructures. In the following, we consider specific instances of (1).

3.1 Random Walk Kernels

Random walk kernels were introduced in [Gär03] for unweighted graphs. Roughly speaking, they decompose a graph into sequences of nodes generated by random walks, and count the number of identical random walks that can be found in two graphs. Thus, the random walk kernel is an R-convolution kernels with substructures given by paths. In the following, we present an extension of these kernels to the case of edge-weighted graphs.

Interestingly, to compute a graph kernel, it is not necessary to sample random walks. Instead, one can exploit an important property of the adjacency matrix A of a graph G, namely that $[A^n]_{i,j}$ is the number of paths of length n from node i to node j; here, A^n denotes the n-th power of A. Let $G_\times = G \times G'$ be the product graph of the graphs G and G', where the node and the edge set of $G_\times$ are defined as follows:

$$\begin{aligned} V_\times &= \{\, (v_i, v'_j) \,|\, v_i \in V,\, v'_j \in V',\, l_V(v_i) = l_V(v'_j) \,\} \\ E_\times &= \left\{\, \big((v_i, v'_j), (v_k, v'_l)\big) \in V_\times \times V_\times \mid \|l_E(v_i, v_k) - l_E(v'_j, v'_l)\| \leq \epsilon \,\right\} \end{aligned}$$

Since $[A^n_\times]_{i,j}$ now corresponds to the number of equal paths of length n from node i to node j that occur in G as well as in G', the product graph $G_\times$ allows one to calculate $k(G, G')$ by performing simple matrix-operations. The requirement that node labels and edge weights have to match along two paths is implicitly encoded in the definition of the product graph (namely by the restriction to node pairs with $l_V(v_i) = l_V(v'_j)$ and edges with $\|l_E(v_i, v'_j) - l_E(v_k, v'_l)\| \leq \epsilon$); this idea was already used by [BOS+05], albeit only for discrete edge labels. The similarity of the graphs G and G', considering all equal paths of length 1 to ∞, is finally given by

$$k_{RW}(G, G') = \sum_{i,j=1}^{|V_\times|} \left[\sum_{k=0}^{\infty} \lambda_k \cdot A_\times^k \right]_{i,j} , \tag{2}$$

where λ_k is a factor that guarantees convergence of the series. For certain choices of λ, the above series can be calculated in a simple way. Choosing $\lambda_k = (1/a)^k$, with $a \geq max_{v \in V_\times}\{\text{degree}(v)\}$, leads to the geometrical series, and (2) reduces to

$$k_{RW_{geo}}(G, G') = \sum_{i,j=1}^{|V_\times|} \left[(I - \lambda \cdot A_\times)^{-1}\right]_{i,j} , \tag{3}$$

where I is the unit matrix. Choosing $\lambda_k = \frac{\beta^k}{k!}$ leads to the exponential series and to

$$k_{RW_{exp}}(G, G') = \sum_{i,j=1}^{|V_\times|} \left[e^{\beta \cdot A_\times}\right]_{i,j} .$$

Since the product graph is of quadratic size and matrix inversion has cubic complexity, the complexity of the random walk kernel is $\mathcal{O}(M^6)$, with $M = \max\{|V|, |V'|\}$.

3.2 Shortest Path Kernels

The random walk kernel considers an extremely large number of substructures (paths). Intuitively, this may not only come with a high computational complexity but also produce a certain redundancy. To reduce the number of substructures, Borgwardt [BK05] proposed to consider only the shortest paths between two nodes, an idea which leads to the shortest path kernel. Again, we propose an extension of this kernel to the case of edge-weighted graphs.

For two nodes $v_i, v_j \in G$, let $sp(v_i, v_j)$ denote the length of the shortest path (sum of edge weights on the path) between these nodes, and let

$$SP(v_i, v_j) \;=\; (\{l_V(v_i), l_V(v_j)\}, sp(v_i, v_j)) \;\; .$$

Thus, a path is represented by its length and the labels of the start and the end node (while the node labels in-between are ignored). A simple kernel on substructures of this type is the identity (Dirac kernel):

$$\kappa_{path}(SP(v_i, v_j), SP(v_k, v_l)) = \begin{cases} 1 & \text{if } SP(v_i, v_j) = SP(v_k, v_l) \\ 0 & \text{else} \end{cases} \;\; .$$

Since testing equality is of course not reasonable for real-valued edge lengths, we assume these lengths to be discretized (into bins of length δ).

Now, we can define the generalized shortest path kernel as follows:

$$k_{SP}(G, G') = \frac{1}{C} \sum_{v_i, v_j \in V} \sum_{v_k, v_l \in V'} \kappa_{path}(SP(v_i, v_j), SP(v_k, v_l)) \;\; ,$$

where $C = \frac{1}{4}(|V|^2 - |V|) \cdot (|V'|^2 - |V'|)$ is a normalizing factor that guarantees $0 \leq k_{SP}(G, G') \leq 1$.

To analyze the complexity of the shortest path kernel, assume $|V| = |V'| = M$. The computation of all shortest paths can be done using the Floyd-Warshall [Flo62] algorithm in time $\mathcal{O}(M^3)$. The results are stored in a shortest path matrix, in which the entry at position (i, j) gives the cost of the shortest path from node i to node j. We consider in a pairwise way all paths in both shortest path matrices and compare them using κ_{path} which needs time $\mathcal{O}(1)$. Since there are M^4 comparisons to perform, the shortest path kernel needs time $\mathcal{O}(M^4)$.

3.3 Fingerprint Kernels

A very simple type of kernel, which has nevertheless been applied successfully for learning on structured data such as molecules [FHZ06], is based on the idea of mapping a structured object to a fingerprint vector of fixed length first, and to compare these vectors afterward. Typically, each entry in this vector informs about the presence or absence of a specific substructure (pattern).

In our case, we consider as substructures all non-isomorphic graphs of size 3. Assuming n distinct node and k distinct edge labels, there exist

$$N(n,k) = \binom{n}{3} \cdot k^3 + n(n-1) \cdot k \cdot \binom{k+1}{2} + n \cdot \binom{k+2}{3}$$

substructures of this type, which can be verified by means of a case distinction: (i) All three node labels are distinct: There are $\binom{n}{3}$ possibilities to choose 3 distinct labels from a set of n labels. Moreover, since edges are ordered uniquely in this case, there exist k^3 possibilities for the edge labels. (ii) Two node labels are equal and different from the third: There are $n(n-1)$ possibilities to choose the two labels, one for the identically labeled nodes and one for the other. Assuming an arbitrary ordering on the nodes and edges, an isomorphism can switch the equally labeled nodes so that the ordering of two edges will change, too. To map isomorphic graphs uniquely, we sort the edges, which leads to only $k \cdot \binom{k+1}{2}$ possible edge combinations. (iii) All nodes have equal label: An isomorphism can reorder all nodes in this case. Therefore, to obtain a unique representation of the possible graphs, all edges must be sorted according to their label. Thus, there are n possible node labels and $\binom{k+2}{3}$ edge combinations.

For a graph G, let

$$f_G \;=\; \big(G \sqsupseteq t_1, G \sqsupseteq t_2, \ldots, G \sqsupseteq t_{N(n,k)}\big) \in \{0,1\}^{N(n,k)}$$

where $\{t_1, \ldots, t_{N(n,k)}\}$ is the set of all non-isomorphic subgraphs of size 3, numbered in an arbitrary but fixed order. The predicate $G \sqsupseteq t_i$ tests whether t_i is contained in G and, by convention, returns 1 if it evaluates to `true` and 0 otherwise. To compare two graphs G and G' in terms of their respective fingerprint vectors f_G and $f_{G'}$, different kernels can be used. The simplest approach is to look for the Hamming distance of the two vectors, which leads to

$$k_{FPH}(G,G') = \frac{1}{N(n,k)} \sum_{i=1}^{N(n,k)} \kappa_\delta([f_G]_i, [f_{G'}]_i) \;\;, \tag{4}$$

where $[f_G]_i$ denotes the i-th entry in the vector f_G, and κ_δ is the Dirac kernel (i.e., $\kappa_\delta(x,y) = 1$ if $x = y$ and $= 0$ if $x \neq y$). As a potential disadvantage of this approach, note that it does not only reward the co-occurrence of a substructure in both graphs, but also the simultaneous absence: If the i-th pattern neither occurs in G nor in G', then $\kappa_\delta([f_G]_i, [f_{G'}]_i) = \kappa_\delta(0,0) = 1$, which may not be desirable. An alternative measure avoiding this problem is the well-known Jaccard coefficient:

$$k_{FPJ}(G,G') = \frac{\sum_{i=1}^{N(n,k)} \min([f_G]_i, [f_{G'}]_i)}{\sum_{i=1}^{N(n,k)} \max([f_G]_i, [f_{G'}]_i)} \;\;. \tag{5}$$

Our current implementation of the fingerprint approach is a naive one, in which testing the presence of a substructure in a graph G has complexity $O(M^3)$, with $M = |V|$ the number of nodes in G. Thus, the overall complexity of computing $k(G,G')$ is $O(N(n,k) \cdot M^3)$, with $M = \max(|V|, |V'|)$. Of course, more efficient implementations are possible, for example based on the use of hashing techniques [WKHK04].

4 Experimental Evaluation

In our experiments, we compared the graph kernels discussed in the previous section, namely the random walk kernel (RW) using (3) with a given by the maximum size of the graphs in the data set (plus 1), the shortest path kernel (SP), and the fingerprint kernel based on (4) and (5), respectively (FPH and FPJ). Moreover, to get an idea of their absolute performance, we additionally included two state-of-the-art methods for comparing protein binding sites in terms of their similarity. Both approaches are based on the concept of a *graph alignment* that has recently been introduced in [WHKK07]. The first method (GA) is the original algorithm proposed in the same paper, which is based on a heuristic (greedy) optimization strategy. The second method (GAVEO) makes use of evolutionary optimization techniques to compute a graph alignment [FMKH09]. Both methods need a number of parameters, which we defined as recommended in [WHKK07]. For the kernel methods, we set the parameter ϵ (tolerance for edge length comparison) to 0.2.

The assessment of a similarity measure for biomolecular structures, such as protein binding sites, is clearly a non-trivial problem. In particular, since the concept of similarity by itself is rather vague and subjective, it is difficult to evaluate corresponding measures in an objective way. To circumvent this problem, we propose to evaluate similarity measures in an indirect way, namely by means of their performance in the context of nearest neighbor (NN) classification. The underlying idea is that, the better a similarity measure is, the better he predictive performance we expect from an NN classifier using this measure for determining similar cases.

4.1 Data

We selected two classes of binding sites that bind, respectively, to NADH or ATP. This gives rise to a binary classification problem: Given a protein binding site, predict whether it binds NADH or ATP. More concretely, we compiled a set of 355 protein binding pockets representing two classes of proteins that share, respectively, ATP and NADH as a cofactor. To this end, we used CavBase to retrieve all known non-redundant ATP and NADH binding pockets that were co-crystallized with the respective ligand. Subsequently, we reduced the set to one cavity per protein, thus representing the enzymes by a single binding pocket to ensure that no identical binding pockets are present in our data set. As protein ligands adopt different conformations due to their structural flexibility, it is likely that the ligands in our data set are bound in completely different conformations, hence the corresponding binding pockets do not necessarily share much structural similarity. To ensure a minimum level of similarity, we therefore utilized the ligand information available for these binding pockets, as these structures where all co-crystallized with the corresponding ligand. Using the Kabsch algorithm [Kab76], we calculated the root mean squared deviation (RMSD) between pairs of ligand structures and combined all proteins whose ligands yielded a RMSD value below a threshold of 0.4, thus ensuring that the ligands are roughly oriented in the same way. This value was chosen as a trade-off between data set size and similarity. Eventually, we thus obtained a two-class data set comprising 214 NADH-binding proteins and

141 ATP-binding proteins.

4.2 Results

The performance of the different methods, using a simple k-nearest neighbor classifier ($k = 1, 3, 5, 7, 9$) for prediction, is summarized in Table 1. More specifically, the table shows the percentage of correct classifications in a leave-one-out cross validation: For each structure, a class prediction is derived from its k nearest neighbors (in terms of the respective similarity measure) by means of majority voting, and the prediction is compared with the true class.

method	RW	SP	FPH	FPJ	GA	GAVEO
k = 1	0.597	0.606	0.828	0.842	0.766	0.789
k = 3	0.597	0.628	0.839	0.882	0.718	0.766
k = 5	0.597	0.634	0.839	0.873	0.724	0.780
k = 7	0.608	0.625	0.819	0.859	0.718	0.786
k = 9	0.608	0.634	0.814	0.836	0.713	0.766

Table 1: Classification rates of a k-nearest-neighbor classifier in a leave-one-out cross validation using different values of k and different similarity measures: random walk kernel (RW), shortest path kernel (SP), fingerprint kernel (FPH, FPJ), and graph alignment (GA, GAVEO).

Table 2 shows the average time complexity of the methods, namely the time needed for a single pairwise comparison of two structures. These numbers have been determined by averaging over 1000 comparisons with randomly chosen structures.

method	RW	SP	FP	GA	GAVEO
runtime	65.51 ± 89.07	9.75 ± 97.77	2.05 ± 3.66	74.24 ± 85.61	> 5 min

Table 2: Average runtime (in seconds) of the different methods for a single pairwise comparison.

We investigated the behavior of the best approach FPJ more in detail. A critical parameter of this approach is k, the number of distinct edge labels, that influence strongly the number $N(n, k)$ of graphs of size three. Obviously the runtime will decrease if k is becoming smaller since there are less comparisons to perform. A remaining question is, if as a consequence thereof the accuracy is also decreasing. To investigate this we varied the granularity (discretized edge weights into bins of length δ) and measured the accuracy and runtime for the whole leave-one-out procedure. As can be seen in figure 1 the runtime is a strictly decreasing curve as already prognosticated. However, the benefit of a lower runtime is redeemed by a lower accuracy. Nevertheless, the runtime decreases much faster than the accuracy so that for a fast screening of a database higher δ values can be used. We do not recommend to use smaller δ values since the runtime is growing exponentially with decreasing δ.

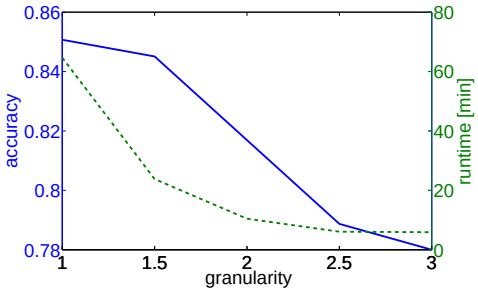


Figure 1: Runtime and accuracy w.r.t. δ; the dotted line illustrates the runtime, the solid line the accuracy.

5 Discussion and Conclusion

The results convey are relatively clear picture: The fingerprint kernels perform best, the random walk and shortest path kernel worst, and the graph alignment methods are in-between. The overall best results are achieved by the Jaccard-variant of the fingerprint kernel. In terms of efficiency, the fingerprint kernels are superior, too (despite the naive implementation). Thus, this type of kernel is clearly of high interest in the context of comparing protein binding sites.

The poor performance of the random walk and shortest path kernels can possibly be attributed to their characteristics as R-convolution kernels. In general, the 'all-against-all' comparison of substructures performed by kernels of this type appears to be problematic for diverse objects with a large number of substructures. In the random walk kernel, nodes and edges can appear more than once in a random walk, a problem known as *tottering*. This problem becomes especially severe in the presence of many cycles within a graph, a property which, as mentioned earlier, our graph descriptors of protein binding sites will inevitably exhibit. The shortest path kernel avoids tottering but has another problem known as *halting*: As it only looks at shortest paths, it tends to be dominated by a large number of paths with very few nodes. As we consider graphs representing geometric constraints within a binding pocket, this is likely to result in a loss of information.

The strong performance of the fingerprint kernel suggests to elaborate on this approach in more detail. In fact, the approach presented in this paper is rather simple and can be extended in different ways. First, substructures other than subgraphs of size 3 might be considered, even though our experience so far has shown that this class of patterns is able to capture considerable information while still being manageable in terms of complexity. Second, the fingerprint vectors could be constructed (and compared) in a more sophisticated way. For example, instead of just indicating the presence or absence of a pattern, one may count its number of occurrences and then apply similarity measures for frequency vectors. Besides, as mentioned earlier, the approach can be implemented in a much more efficient way.

References

[BJ00] Horst Bunke and Xiaoyi Jiang. Graph matching and similarity. *Intelligent systems and interfaces*, 15:281 – 304, 2000.

[BK05] K. M. Borgwardt and H. P. Kriegel. Shortest-path kernels on graphs. In *International Conference on Data Mining*, pages 74–81, Houston, Texas, 2005.

[BL04] Johannes Berg and Michael Lässig. Local graph alignment and motif search in biological networks. *Proceedings of the National Academy of Sciences of the United States of America*, 101(41):14689–14694, 2004.

[Bor07] K. M. Borgwardt. *Graph Kernels*. PhD thesis, Ludwig-Maximilians-Universität München, Germany, 2007.

[BOS+05] Karsten Borgwardt, Cheng Soon Ong, Stefan Schönauer, S. V. N. Vishwanathan, Alex J. Smola, and Hans-Peter Kriegel. Protein function prediction via graph kernels. *Bioinformatics*, 21(21):i47 – i56, 2005.

[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 Research*, 28:235–242, 2000.

[FHZ06] N. Fechner, G. Hinselmann, and A. Zell. Implicitly Defined Substructure Fingerprints for Support Vector Machines. In *German Conference on Chemoinformatics*, 2006.

[Flo62] R. W. Floyd. Algorithm 97: Shortest path. *Communications of the ACM*, 5(6):345, 1962.

[FMKH09] Thomas Fober, Marco Mernberger, Gerhard Klebe, and Eyke Hüllermeier. Evolutionary Construction of Multiple Graph Alignments for the Structural Analysis of Biomolecules. *Bioinformatics*, 2009.

[Gär03] Thomas Gärtner. A survey of kernels for structured data. *SIGKKD Explorations*, 5(1):49 – 58, 2003.

[GMB96] J. F. Gibrat, T. Madej, and S. H. Bryant. Surprising Similarities in Structure Comparison. *Current Opinion in Structural Biology*, 6(3):377–385, 1996.

[HRB97] M. Hendlich, F. Rippmann, and G. Barnickel. LIGSITE: Automatic and efficient detection of potential small molecule-binding sites in proteins. *Journal of Molecular Graphics and Modelling*, 15:359–363, 1997.

[JIDG03] M. Jambon, A. Imberty, G. Deleage, and C. Geourjon. A New Bioinformatic Approach to Detect Common 3 D Sites in Protein Structures. *Proteins Structure Function and Genetics*, 52(2):137–145, 2003.

[Kab76] Wolfgang Kabsch. A solution of the best rotation to relate two sets of vectors. *Acta Crystallographica*, 32:922–923, 1976.

[NB07] Michael Neuhaus and Horst Bunke. *Briding the Gap between Graph Edit Distance and Kernel Machines*. World Scientific, New Jersey, 2007.

[RGW02] J.W. Raymond, E.J. Gardiner, and P. Willett. Heuristics for Similarity Searching of Chemical Graphs Using a Maximum Common Edge Subgraph Algorithm. *Jorunal of Chemical Information and Computer Sciences*, 42(2):305–316, 2002.

[RW02] J. Raymond and P. Willett. Maximum common subgraph isomorphism algorithms for the matching of chemical structures. *Journal of Computer-Aided Molecular Design*, 16(7):521–533, 2002.

[SHK01] S. Schmitt, M. Hendlich, and G. Klebe. From Structure to Function: A New Approach to Detect Functional Similarity among Proteins Independent from Sequence and Fold Homology. *Angewandte Chemie International Edition*, 40(17):3141 – 3144, 2001.

[SKK02] S. Schmitt, D. Kuhn, and G. Klebe. A New Method to Detect Related Function Among Proteins Independent of Sequence and Fold Homology. *Journal of Molecular Biology*, 323(2):387–406, 2002.

[WHKK07] N. Weskamp, E. Hüllermeier, D. Kuhn, and G. Klebe. Multiple Graph Alignment for the Structural Analysis of Protein Active Sites. *IEEE Transactions on Computational Biology and Bioinformatics*, 4(2):310–320, 2007.

[WKHK04] N. Weskamp, D. Kuhn, E. Hüllermeier, and G. Klebe. Efficient Similarity Search in Protein Structure Databases: Improving Clique-Detection through Clique-Hashing. *Bioinformatics*, 20(10):1522–1526, 2004.

Aligning Protein Structures Using Distance Matrices and Combinatorial Optimization

Inken Wohlers* Lars Petzold† Francisco S. Domingues‡ Gunnar W. Klau*

Abstract: Structural alignments of proteins are used to identify structural similarities. These similarities can indicate homology or a common or similar function. Several, mostly heuristic methods are available to compute structural alignments.

In this paper, we present a novel algorithm that uses methods from combinatorial optimization to compute provably optimal structural alignments of sparse protein distance matrices. Our algorithm extends an elegant integer linear programming approach proposed by Caprara *et al.* for the alignment of protein contact maps. We consider two different types of distance matrices with distances either between C_α atoms or between the two closest atoms of each residue. Via a comprehensive parameter optimization on HOMSTRAD alignments, we determine a scoring function for aligned pairs of distances. We introduce a negative score for non-structural, purely sequence-based parts of the alignment as a means to adjust the locality of the resulting structural alignments.

Our approach is implemented in a freely available software tool named PAUL (Protein structural Alignment Using Lagrangian relaxation). On the challenging SISY data set of 130 reference alignments we compare PAUL to six state-of-the-art structural alignment algorithms, DALI, MATRAS, FATCAT, SHEBA, CA, and CE. Here, PAUL reaches the highest average and median alignment accuracies of all methods and is the most accurate method for more than 30% of the alignments. PAUL is thus a competitive tool for pairwise high-quality structural alignment.

1 Introduction

Background. Structural alignments of proteins help identify structural similarities. They are used to detect homologous proteins, to identify common structural elements, and to determine protein function. Frequently, the function of a protein is defined by its three-dimensional structure, and protein structure is often more conserved during evolution than protein sequence. Therefore, structural alignment is especially useful to detect remotely homologous proteins with low sequence identity, which lie either in the twilight zone [Doo86] of 20% to 35% sequence identity or in the midnight zone [Ros97] of less than 20% sequence identity. Furthermore, structural alignment is applied to identify new protein folds or to map protein structures to already established folds. Detected structural similarities are used effectively for functional annotation [YYS+04].

There are two established approaches to compute protein structural alignments: minimiz-

*CWI, P.O. Box 94079, 1090 GB Amsterdam, Netherlands, {inken.wohlers,gunnar.klau}@cwi.nl
†Freie Universität Berlin, 14195 Berlin, Germany
‡Max Planck Institute for Informatics, 66123 Saarbrücken, Germany

ing the root mean square deviation (RMSD) of rigid body superposition and maximizing the score for an assignment of distance matrix rows and columns. A popular heuristic algorithm of the second type is DALI [HS93]. DALI scans in a first step protein distance matrices for similar distance patterns by computing a similarity score for aligning fragments of six residues. In a second step combinations of non-overlapping fragments are repeatedly chosen in a random fashion. Each set of fragments makes up an alignment and is evaluated using a scoring function. Finally the alignment with highest score is reported. Several other algorithms also aim at finding good combinations of aligned fragment pairs, *e.g.*, CE [SB98] and FATCAT (flexible structure alignment by chaining aligned fragment pairs allowing twists) [YG03]. Other methods like MATRAS (Markovian transition of structure evolution) [Kaw03] match in a first step secondary structure elements and compute an alignment on atomic level in a second step. A sequence-order independent approach to compute alignments is geometric hashing, which is applied by CA [BFNW93]. The method SHEBA (structural homology by environment-based alignment) [JL00] compares in a first step lists of primary, secondary and tertiary structure characteristics and then improves the initial alignment using weighted RMSD. Further state-of-the-art approaches are SSAP [TO89], which is based on double dynamic programming, PPM [CBZ08], a method that minimizes the cost of morphing one structure into the other, TM-ALIGN [ZS05] that maximizes the TM-score, and PROTDEFORM [RSWD09] and MATT [MBC08], which align proteins in a flexible fashion. Furthermore, structural alignments have also been computed by aligning protein contact maps [CCI$^+$04].

Contribution. In this paper, we present a structural alignment approach based on combinatorial optimization. In our approach, which builds upon an algorithm for the alignment of protein contact maps by Caprara *et al.* [CCI$^+$04], we align sparse distance matrices. We compute an alignment by maximizing a function that scores aligned distances. We tailor our method specifically towards high-quality pairwise alignments. In order to efficiently use the elegant integer linear programming approach of [CCI$^+$04] we determine a suitable distance threshold and scoring function, decrease the number of variables in the integer linear program and add a parameter that scores non-structural, purely sequence-based parts of the alignment in order to balance global against local alignment. We optimize our method for C_α distance matrices as well as for all-atom distance matrices that contain the minimum distance between any pair of atoms of two residues. In this study we investigate which distances should be included in the integer linear program in order to increase the accuracy of pairwise alignments. We did not optimize for speed—this issue will be dealt with in future work. Our approach is implemented in the freely available software tool PAUL (protein structural alignment using Lagrangian relaxation). We evaluate PAUL on the challenging SISY data set [MDL07] comparing it to six state-of-the-art structural alignment tools. PAUL reaches higher average and median alignment accuracies than any of the other methods.

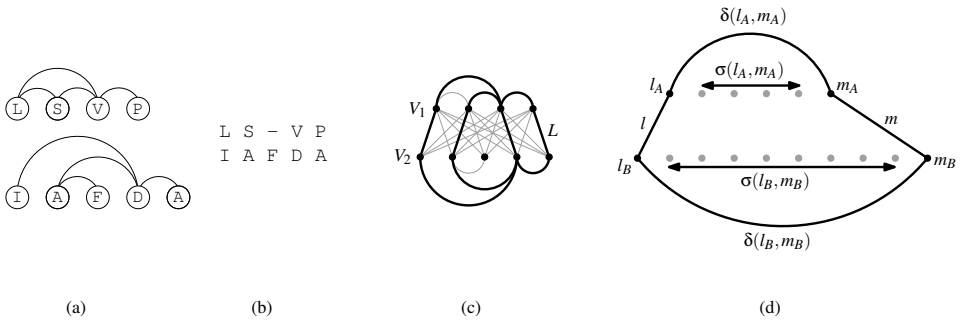


Figure 1: Maximum contact map overlap problem. (a) Two protein contact maps. (b) Alignment of the two proteins. (c) Corresponding solution in the graph problem. Alignments are characterized by non-crossing matches, or *traces*, in the complete bipartite alignment graph $(V_1 \cup V_2, L)$ [Kec93]. Here, vertices in V_1 and V_2 denote the residues of the two proteins, resp., and L is the complete set of alignment edges, *i.e.*, $L = \{(i, j) \mid i \in V_1, j \in V_2\}$. The displayed trace (bold alignment edges) maximizes the contact map overlap, in the example there are three shared contacts (also shown in bold). (d) A pair of aligned distances. Functions $\delta(\cdot, \cdot)$ and $\sigma(\cdot, \cdot)$ denote the distance between two residues with respect to their three-dimensional coordinates and sequential position, respectively.

2 Methods

Combinatorial approach to structural alignment. In [CCI+04], the authors give an algorithm to compute pairwise alignments of two protein structures that maximize the number of common contacts. Two residues of a protein are in contact if they are in some sort of chemical interaction, *e.g.*, by hydrogen bonding. However, a simple distance criterion is used: whenever the distance between two residues is below a predefined threshold, the residues are considered to be in contact. Caprara *et al.* have introduced the *maximum contact map overlap* problem and have given an integer linear programming (ILP) formulation. They propose to solve the ILP using an elegant Lagrangian relaxation approach.

The underlying ILP formulation relies on a reformulation of the structural alignment problem as a graph problem. Figure 1 explains the relation. In their ILP approach, Caprara *et al.* introduce binary variables x_l for each alignment edge $l \in L$ and binary variables y_{lm} for each potentially shared common contact represented by the two alignment edges l and m. The binary variables indicate the presence or absence of the corresponding objects in the solution. The authors express the set of feasible solutions using linear inequalities and integrality constraints involving x and y and find the largest set of common contacts using the objective function $\max \sum_{(l,m)\in\binom{L}{2}} y_{lm}$. For a detailed description, refer to [CCI+04].

We extend the approach by Caprara *et al.* by replacing the rigid contact definition and taking into account the three-dimensional and sequential distances between the residues in order to align inter-residue distances. Let (l, m) be a pair of aligned distances of two proteins A and B with $l = (l_A, l_B)$ and $m = (m_A, m_B)$, see also Fig. 1(d). We use two distance measures $\delta(\cdot, \cdot)$ and $\sigma(\cdot, \cdot)$ that denote the distance between two residues with respect to their three-dimensional coordinates and sequential position, resp., and are

now able to align inter-residue distances instead of contacts. To this end, we replace the objective function $\max \sum_{(l,m)\in\binom{L}{2}} y_{lm}$ by

$$\max \sum_{(l,m)\in\binom{L}{2}} w_{lm} y_{lm} + \sum_{l\in L} c\, x_l \quad \text{, where} \tag{1}$$

$$w_{lm} = \begin{cases} \max\{0, \theta_R - \Delta_{lm}\} & \Delta_{lm} \leq \Delta_t \text{ and } \Gamma_{lm} \leq \Gamma_t \\ 0 & \text{otherwise} \quad , \end{cases} \tag{2}$$

with $\Delta_{lm} = |\delta(l_A, m_A) - \delta(l_B, m_B)|$ and $\Gamma_{lm} = |\sigma(l_A, m_A) - \sigma(l_B, m_B)|$. Here, θ_R, Δ_t, Γ_t, and c are constant parameters. Our choices of (1) and (2) are motivated by the following considerations.

1. A scoring function for pairs of inter-residue distances from two proteins A and B should be symmetric with respect to the order of A and B. This is achieved by taking absolute values.
2. Aligning similar distances is more preferable than aligning dissimilar ones, *i.e.*, the contribution of a pair of aligned distances to the objective function should decrease with increasing difference Δ_{lm}. Inspired by the *rigid similarity* measure introduced by Holm and Sander in their paper [HS93] on DALI, we use the term $\theta_R - \Delta_{lm}$ to score pairs of distances. Parameter θ_R modulates the score such that in one extreme example slight differences in distance cause great differences in score and in the other extreme example all combinations of distances have the same score—this second scoring is identical to the contact map overlap. See also Fig. 2(b).
3. Analogous to contact map alignment, the overall time and space complexity of our method is $O(|E_A|, |E_B|)$, where E_A and E_B are the numbers of distances in the ILP from protein A and B respectively. The major restriction of solving protein structural alignment to provable optimality is therefore the high demand of computational resources. In principle, each pair of distances has to be considered explicitly in the ILP, leading to $\binom{n_A}{2}\binom{n_B}{2}$ y-variables, where n_A and n_B are the number of residues in proteins A and B, resp. The Lagrangian relaxation approach is highly sensitive to the number of y-variables, making it practically infeasible to include all pairs of distances. Therefore, we consider only such distance combinations that are likely to denote structural similarity between the two proteins and derive a distance threshold d_t and a threshold for distance differences Δ_t. We find that sequential distance differences Γ_{lm} of aligned distances are typically low, but are aware that by applying a threshold Γ_t we neglect distances between different secondary structure elements that are divided by an insertion or deletion greater than Γ_t. Therefore we do not apply a threshold Γ_t in this study. Applying thresholds leads to a large number of variables y_{lm} with $w_{lm} = 0$. Due to the nature of the ILP formulation, we can safely omit variables with zero coefficients.
4. Due to the structure of the ILP we do not have the possibility to penalize aligned distances by using negative scores w_{lm}. Therefore we penalize parts of the alignment without structural conservation by giving each alignment edge a negative score c. Thus, alignment edges will only be chosen if they contribute significantly to multiple pairs of aligned distances with large weight. This prevents the alignment of residues that do not

indicate sufficient structural similarity. If we decrease this penalty, we can tune PAUL towards local alignment by concentrating on structurally extremely similar parts while neglecting less similar parts. On the other side we can tune PAUL also towards rigorous global alignment by increasing the penalty or setting it to zero.

Implementation. We implemented the novel structural alignment algorithm as the freely available package PAUL within the C++ software library PLANET LISA [K$^+$]. PAUL supports different input formats, *e.g.*, PDB files, lists of pre-selected distances or complete distance matrices. Distance matrix representations are currently built internally in either of two modes: based on the distances between the C_α atoms of residues or based on the minimum distance between each pair of atoms of residues. While the alignment of C_α distances aims at finding equal or similar protein backbone conformations, the alignment of all-atom distances shows similar residue interactions in the two proteins. Beside the type of distances, scoring function parameters can also be chosen. In this way additional information about the pair of proteins that are aligned can be incorporated. For instance, the penalty c can be adjusted to favour global or local alignment. By default, the optimized scoring function parameters for C_α and all-atom distances reported in this paper are used.

Experimental setup for parameter setting, optimization, and evaluation. To determine good and robust parameters for the distance difference threshold Δ_t, the steepness θ_R, and the penalty c we use structure-based alignments from the Homologous Structure Alignment Database (HOMSTRAD, Oct 2008 release) [MDBO98]. As these alignments are manually curated by experts, we consider them as gold standard reference alignments. From HOMSTRAD we consider only protein families with exactly two members from the twilight or midnight zone of sequence identities below 35%. Hereby we define sequence identity as the number of identically aligned residues divided by the total number of aligned residues. We optimize the parameters on a training set of 200 alignments and evaluate them on a test set that consists of the remaining 102 alignments. We measure the quality of the results computed by structural alignment algorithms in terms of the achieved alignment accuracy, which is the number of correctly aligned residues divided by the number of aligned residues in the reference alignment.

In a preprocessing step we compute histograms of aligned distances over the training set alignments. Fig. 2(a) displays the results for C_α distances. For all-atom distances the distribution is similar, but shifted to smaller distances. For close distances of less than 12Å we observe distinct peaks for certain aligned distances. These peaks represent typical distances within secondary structure elements and within super-secondary structures. The histograms help identify distance thresholds for C_α and all-atom distance matrices that are qualitatively equivalent in terms of overall number of distances included in the ILP as well as in terms of inclusion of biological features. We optimized parameters for the distance thresholds $d_t \in \{7.5$Å, 8Å,$\ldots$, 10Å$\}$ and $d_t \in \{5$Å, 5.5Å,$\ldots$,7Å$\}$ for C_α and all-atom distance matrices, resp.

We carry out a parameter sweep in order to optimize the scoring function parameters θ_R, Δ_t and c. We use 7 nodes equipped each with two quad core 2.33 GHz Intel Xeon processors and 8 GB of main memory running 64 bit Linux. On each node we compute 4 PAUL alignments in parallel using OpenMP. We choose a maximum time limit of 90 CPU s and a maximum number of $1\,000$ Lagrange iterations for each computation. In a first

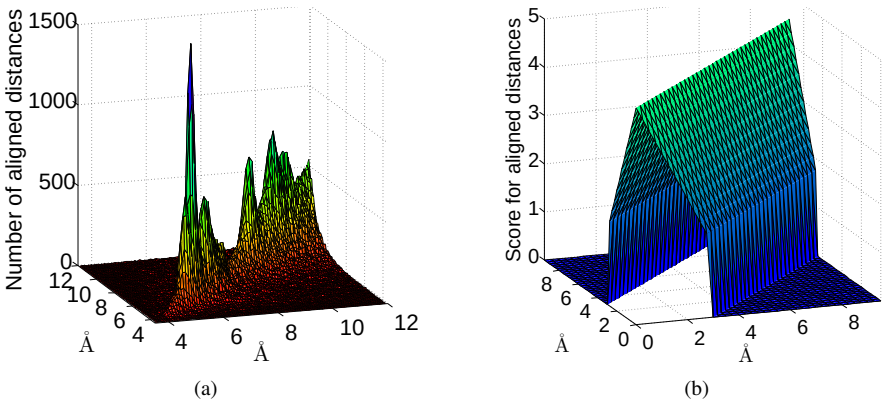


Figure 2: (a) Number of aligned C_α distances over all HOMSTRAD training set alignments. (b) Scoring function for C_α distance matrices ($d_t = 9.5$, $\theta_R = 4.5$ and $\Delta_t = 3$).

broad sweep we choose 10 values for the steepness of the scoring function, θ_R, in such a way that the angle is divided equally. We then refine the sweep by focusing on an interval of good parameter values and equally divide this interval again, obtaining 10 more values for θ_R. For the maximum distance difference we evaluate $\Delta_t \in \{1.5, 2, \ldots, d_t - 1.5\}$ for C_α matrices and $\Delta_t \in \{0.5, 1, \ldots, d_t - 0.5\}$ for all-atom matrices. The sequence penalty ranges over $c \in \{0, -\frac{1}{2}\theta_R, \ldots, -3\theta_R\}$. Since aligning two identical distances receives a maximum score of θ_R, the range of penalty values covers the cases of aligning two residues if they maintain at least 0, 1, 2, or 3 aligned distances of maximum score. We compute the average alignment accuracy for each parameter set (θ_R, Δ_t, c), resulting in more than 150 evaluations of the full training set for each distance threshold. We then apply 10-fold cross-validation in order to assess the performance of PAUL on the HOMSTRAD training set alignments. We use the best parameter set for C_α and all-atom distances resp. to align the HOMSTRAD test set alignments and compare PAUL performance to DALI.

Experimental setup of computational study. We use the parameters optimized on the HOMSTRAD data set to compare PAUL with the state-of-the-art structural alignment programs DALI, MATRAS, FATCAT, SHEBA, CA and CE on a second, distinct data set, the SISY set [MDL07, L^+]. This set is assembled from SISYPHUS [APHM07], a manually curated database for alignments of proteins with non-trivial relationships. It consists of 130 very diverse reference alignments: the lengths of the protein chains vary greatly, from 32 up to 1 283 residues, as do the lengths of the number of aligned residues, from 17 to 372. For aligning the SISY set we use a maximum runtime of 30 minutes per alignment, in order to exploit the benefit of using a high distance threshold. Note that, depending on the pair of proteins, the actual runtime in which we observe improvements is usually a lot shorter (see HOMSTRAD), but in order to proof the optimality of a solution a longer runtime is needed. However, in terms of speed our method is not yet competitive to others, therefore we did not compare runtimes.

	PAUL	MATRAS	DALI	FATCAT	SHEBA	CA	CE
average %	**72.93**	71.83	69.44	62.40	59.43	51.45	50.13
median %	**92.48**	91.42	90.96	78.30	84.41	59.55	57.43

Table 1: Results on the SISY data set. Average and median alignment accuracies of different state-of-the-art structural alignment algorithms. Overall best values denoted in bold.

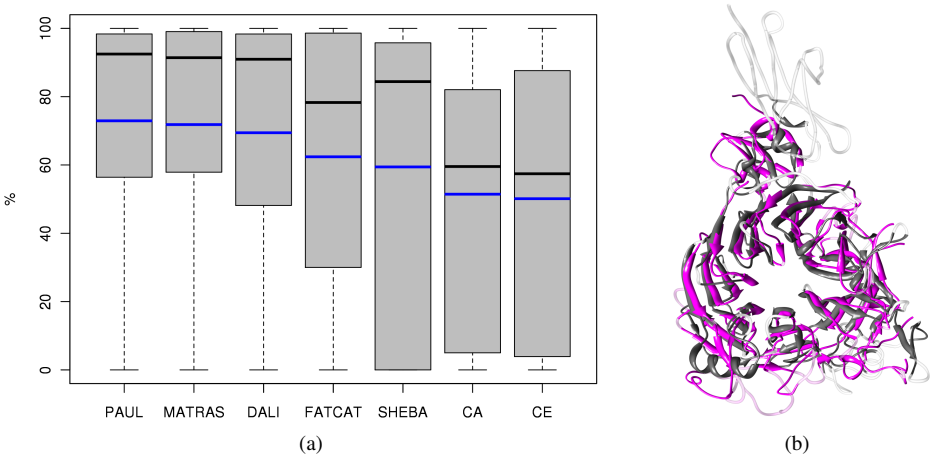
3 Results

Optimized scoring function. The best distance thresholds for C_α distance matrices were 8Å with an alignment accuracy of 87.56% followed by 8.5Å with 87.34% and closely followed by 9.5Å with 87.30%. The corresponding optimized parameters are similar, for $d_t = 9.5$ they are $\theta_R = 4.5$, $\Delta_t = 3$ and $c = -4.5$. For all-atom distance matrices the best parameters are $d_t = 5.5$, $\theta_R = 3.5$, $\Delta_t = 3$ and $c = -1.75$ with an alignment accuracy of 87.23%. Although the parameters vary over a large range, the resulting optimized parameters are of the same order of magnitude for all distance thresholds d_t, with alignment accuracies between 86.77 and 87.56% for C_α distances and between 85.17 and 87.23 for all-atom distances. The result of the 10-fold cross-validation over all distance thresholds and parameter sets amounts to 86.76% for C_α and 86.42% for all-atom distances, compared to 85.08% alignment accuracy achieved by DALI. For a visualization of the optimized scoring function for C_α distances refer to Fig. 2(b).

We test our optimized parameters on the HOMSTRAD test set. For C_α distance matrices PAUL reaches an average alignment accuracy of 85.86% for $d_t = 8$, of 85.49% for $d_t = 8.5$, and of 86.56% for $d_t = 9.5$; all-atom distance matrices with $d_t = 5.5$ reach 86.22%. The alignment accuracy for C_α distances and $d_t = 9.5$ is slightly higher than the average alignment accuracy of DALI alignments, which amounts to 86.32%. Based on these results, we decide to use C_α distance matrices with $d_t = 9.5$ for the evaluation on the SISY data set.

Results on SISY data set. We investigate the alignment accuracy in terms of percentages of correctly aligned residues on the more challenging SISY data set and compare PAUL's performance to six other state-of-the-art structural alignment algorithms. Table 1 contains the average and median alignment accuracies for the set of 130 alignments. Fig. 3(a) shows the distributions of the percentages of alignment accuracies for PAUL and each of the other structural alignment methods using box-and-whisker plots. Fig. 3(b) visualizes a difficult SISY alignment, for which PAUL outperforms the other structural alignment methods.

We observe that PAUL alignments shows higher average and median accuracy than any other method. Furthermore, according to two-sided Wilcoxon signed-rank tests with paired observations, PAUL matches the SISY gold standard alignments significantly better than SHEBA, CA, and CE. Additionally, we investigate the correlation between alignment accuracy values using Pearson correlation coefficients. These are around 0.5 for any pair of methods and are thus generally low, whereas the correlation between PAUL and MATRAS has a Pearson correlation coefficient of 0.56 and between PAUL and DALI of 0.49.



(a) (b)

Figure 3: (a) Box-and-whisker plots display median and quartiles of the distributions of percentages of alignment accuracies for the SISY set for PAUL, MATRAS, DALI, FATCAT, SHEBA, CA and CE. Additionally, blue lines denote the average alignment accuracies. (b) PAUL alignment of semaphorin 4D, PDB 1olz chain A, (grey) with hepatocyte growth factor, PDB 1shy chain B, (purple). Protein lengths are 621 and 499 residues, resp. The proteins are oriented according to the optimal superposition of the matching residues given by PAUL. The alignment given by PAUL is mostly correct, with an alignment accuracy of 94.74%; the other methods generate alignments with lower accuracy (DALI 86.84%, FATCAT 81.58%, MATRAS 57.89%, SHEBA 10.53%, CA 2.63%, CE 0%).

4 Discussion

We suggest a novel structural alignment algorithm that is based on aligning small inter-residue distances using techniques from combinatorial optimization. By considering each combination of distances explicitly in our integer linear program we are able to solve the structural alignment problem on single-residue level and potentially to optimality without applying heuristics. This has several advantages. First of all, only a method that provides provably optimal alignments with respect to the scoring function allows to question, assess, and validate the underlying model, which is, in the case of structural alignment, the measure that evaluates structural similarity. Provably optimal solutions allow to attribute poor alignments to the measure of structural similarity that we maximize. For heuristic methods, however, a corrupt alignment might be suboptimal and then may have to be attributed to a poor search algorithm.

In order to be able to handle the combinatorial complexity in an explicit, non-heuristic manner, we have to restrict our method to sparse distance matrices and accept a significantly longer runtime than other, heuristic methods. PAUL's running time highly depends on protein length and protein similarity and may vary significantly. Therefore, in terms of improving the running time and estimating the status of the solution process, a lot of work still needs to be done. However, using the SISY set, we show that our scoring function and problem formulation is capable of finding difficult similarities, and on the HOM-

STRAD data set we show that this can also be done in shorter time scales, because PAUL achieves higher alignment accuracies than DALI—on the training set, as determined by cross-validation, as well as on the test set. Furthermore, we demonstrate that by aligning only small inter-residue distances, we still can compute alignments as good as or better than alignments computed by DALI, a heuristic structural alignment method that aligns complete inter-residue distance matrices.

There are two aspects that influence the performance of PAUL. Firstly, this is the suitability of scoring function parameters. Optimal or close to optimal parameters are of the same order of magnitude for different distance thresholds d_t. This denotes a common distance difference Δ_t, at which a majority of pairs of distances is structurally non-significant as well as a common preference for differentiated scoring of aligned distances, denoted by θ_R. The second aspect is the distance threshold d_t itself and the resulting computation time. Including more distances in the problem description renders the computation of a good alignment gradually more difficult, inefficient and thus time-consuming. This effect has to be counterbalanced by a gain of accuracy in describing protein structure, which leads to an overall higher alignment accuracy. Remarkably, different distance thresholds d_t and thus different numbers of distances in the ILP led to similar alignment accuracies on HOMSTRAD alignments. Therefore, we find that higher distance thresholds increase alignment accuracy, however, only when combined with a significantly longer runtime. In order to assess the importance of the penalty c we use different penalties to compute SISY alignments. We find that PAUL almost always finds an alignment as good as the best alignment from any of the other six methods. Therefore PAUL almost never fails due to algorithmic problems, but the balance between global and local alignment is crucial.

On the challenging SISY set, PAUL reaches the highest average and median alignment accuracies. This illustrates the soundness of our approach and its capability to detect structural similarity even in difficult cases. An example is given in Fig. 3(b). For more than 30% of the alignments PAUL achieves the maximum alignment accuracy that is reached by the seven structural alignment methods. In addition to its good performance, PAUL computes 21 alignments to provable optimality and thus with maximum score with respect to the scoring function. On the SISY set PAUL alignment accuracies correlate poorer to DALI than to MATRAS alignment accuracies, despite the common approach of aligning inter-residue distances. This might be attributed to qualitatively different scoring functions, to the fact that PAUL aligns only sparse distance matrices, and to the restriction of DALI to compute scores based on fragments and not on single-residue level. PAUL as well as DALI alignments benefit from a high degree of flexibility, because the approach of aligning distances instead of computing rigid superpositions allows to detect similarities of high RMSD, for which other algorithms need to introduce twists. The results on the SISY set thus demonstrate that PAUL is a beneficial tool for high-quality alignments, on its own as well as when complementing other structural alignment methods.

Acknowledgements. We thank Peter Lackner for providing the SISY data set and the results of the structural alignment methods with which we compare PAUL and Ingolf Sommer for valuable comments and initiation of the authors' cooperation. This work has been partly supported by DFG grant KL 1390/2-1. Computational experiments were sponsored by the NCF for the use of supercomputer facilities, with financial support from NWO.

References

[APHM07] A Andreeva, A Prlić, T J Hubbard, and A G Murzin. SISYPHUS–structural alignments for proteins with non-trivial relationships. *Nucleic Acids Res*, 35(Database issue):253–259, Jan 2007.

[BFNW93] O Bachar, D Fischer, R Nussinov, and H Wolfson. A computer vision based technique for 3-D sequence-independent structural comparison of proteins. *Protein Eng*, 6(3):279–288, Apr 1993.

[CBZ08] G Csaba, F Birzele, and R Zimmer. Protein structure alignment considering phenotypic plasticity. *Bioinformatics*, 24(16):98–104, Aug 2008.

[CCI+04] A Caprara, R Carr, S Istrail, G Lancia, and B Walenz. 1001 optimal PDB structure alignments: integer programming methods for finding the maximum contact map overlap. *J Comput Biol*, 11(1):27–52, 2004.

[Doo86] R F Doolittle. *Of URFs and ORFs: a primer on how to analyze derived amino acid sequences*. University Science Books, Mill Valley, CA, USA, 1986.

[HS93] L Holm and C Sander. Protein structure comparison by alignment of distance matrices. *J Mol Biol*, 233(1):123–138, Sep 1993.

[JL00] J Jung and B Lee. Protein structure alignment using environmental profiles. *Protein Eng*, 13(8):535–543, Aug 2000.

[K+] G W Klau et al. `http://planet-lisa.net`. Accessed 21 May 2009.

[Kaw03] T Kawabata. MATRAS: A program for protein 3D structure comparison. *Nucleic Acids Res*, 31(13):3367–3369, Jul 2003.

[Kec93] J D Kececioglu. The maximum weight trace problem in multiple sequence alignment. In *Proc 4th Annual Symposium on Combinatorial Pattern Matching (CPM 93)*, volume 684 of *LNCS*, pages 106–119. Springer–Verlag, 1993.

[L+] P Lackner et al. `http://biwww.che.sbg.ac.at/RSA/`. Accessed 8 May 2009.

[MBC08] M Menke, B Berger, and L Cowen. Matt: local flexibility aids protein multiple structure alignment. *PLoS Comput Biol*, 4(1), Jan 2008.

[MDBO98] K Mizuguchi, C M Deane, T L Blundell, and J P Overington. HOMSTRAD: a database of protein structure alignments for homologous families. *Protein Sci*, 7(11):2469–2471, Nov 1998.

[MDL07] G Mayr, F S Domingues, and P Lackner. Comparative analysis of protein structure alignments. *BMC Struct Biol*, 7:50–50, 2007.

[Ros97] B Rost. Protein structures sustain evolutionary drift. *Fold Des*, 2(3):19–24, 1997.

[RSWD09] J Rocha, J Segura, R C Wilson, and S Dasgupta. Flexible structural protein alignment by a sequence of local transformations. *Bioinformatics*, 25(13):1625–1631, Jul 2009.

[SB98] I N Shindyalov and P E Bourne. Protein structure alignment by incremental combinatorial extension (CE) of the optimal path. *Protein Eng*, 11(9):739–747, Sep 1998.

[TO89] W R Taylor and C A Orengo. Protein structure alignment. *Journal of Molecular Biology*, 208(1):1–22, Jul 1989.

[YG03] Y Ye and A Godzik. Flexible structure alignment by chaining aligned fragment pairs allowing twists. *Bioinformatics*, 19 Suppl 2:246–255, Oct 2003.

[YYS+04] A F Yakunin, A A Yee, A Savchenko, A M Edwards, and C H Arrowsmith. Structural proteomics: a tool for genome annotation. *Curr Opin Chem Biol*, 8(1):42–48, Feb 2004.

[ZS05] Y Zhang and J Skolnick. TM-align: a protein structure alignment algorithm based on the TM-score. *Nucleic Acids Res*, 33(7):2302–2309, 2005.

Self-taught learning for classification of mass spectrometry data: a case study of colorectal cancer

Theodore Alexandrov
Center for Industrial Mathematics (ZeTeM), University of Bremen,
Bibliothekstr. 1, D-28209 Bremen, Germany, theodore@math.uni-bremen.de

Abstract: Mass spectrometry is an important technique for chemical profiling and is a major tool in proteomics, a discipline interested in large-scale studies of proteins expressed by an organism. In this paper we propose using a sparse coding algorithm for classification of mass spectrometry serum protein profiles of colorectal cancer patients and healthy individuals following the so-called self-taught learning approach. Being applied to the dataset of 112 spectra of length 4731 bins, the sparse coding algorithm represents each of them by means of less then ten prototype spectra. The classification of spectra is done as in our previous study on the same dataset [ADM+09], using Support Vector Machines evaluated by means of the double cross-validation. However, the classifiers take as input not discrete wavelet coefficients but the sparse coding coefficients. Comparing the classification results with reference results, we show that providing the same total recognition rate, the sparse coding-based procedure leads to higher generalization performance. Moreover, we propose using the sparse coding coefficients for clustering of mass spectra and demonstrate that this approach allows one to highlight differences between the cancer spectra.

1 Introduction

Mass spectrometry (MS) is an important technique for chemical profiling and is a major tool in proteomics, a discipline interested in large-scale studies of proteins expressed by an organism. In medicine, MS-based proteomics contributes to clinical research by identification of biomarker proteins related to a disease, e.g. produced by a tumor tissue or by the immune system in response to a disease. Since 2002, when it was first proposed to classify cancer patients and healthy individuals based on MS protein profiles, researchers have shown an increased interest in application of mass spectrometry for biomarker detection.

Given a sample of blood, urine or serum, an MS instrument produces a high dimensional histogram-like spectrum. The peaks of the spectrum express chemical compounds with high concentrations. The spectra for different groups of subjects are collected (e.g. cancer patients and control individuals groups) and a quality of classification is studied. If a successful classification is possible, one is interested in interpreting peaks which are used in the classification and in identifying proteins corresponding to those peaks.

In [ADM+09], we investigated the use of Discrete Wavelet Transformation (DWT) together with Support Vector Machines (SVM) for classification of spectra of colorectal cancer patients and healthy individuals. First, we calculated wavelet coefficients for each

spectrum. Then statistically different coefficients were classified using SVM. Along with standard DWT we exploited APPDWT ("approximation DWT"), a modified DWT where only approximation coefficients were used. The classification results proved that this type of DWT outperforms the standard DWT. APPDWT can be interpreted as dictionary representation of a spectrum, where the dictionary is constructed by translating and shifting a wavelet scaling function.

Recently, [LBRN06] introduced a sparse coding (SC) algorithm which, given a set of vectors, learns in an unsupervised manner a sparse basis for optimal linear representation of the original vectors. Note that the basis can be overcomplete and its elements are not necessarily orthonormal, i.e., formally speaking, it is not a basis but a dictionary. In [ASKS09] we demonstrated that being applied to MS data, the SC algorithm allows one to pick class-relevant peaks. For this aim, we improved the original SC algorithm replacing l_1-regularization with an elastic-net regularization (combination of l_1- and l_2-regularization terms), for more details see [ASKS09] and [AKL$^+$09].

Later, [RBL$^+$07] proposed using the SC algorithm for classification, calling their approach "self-taught learning" as features used in classification are learned from the data. In this paper we follow this approach, classifying mass spectra of colorectal cancer patients and healthy individuals. The improved version [AKL$^+$09] of the SC algorithm is used. For the classification the same scheme as in [ADM$^+$09] is applied, but instead of DWT (APPDWT) coefficients we exploit the coefficients of the basis learned using the SC algorithm.

Our procedure of classification of mass spectra is as follows. First, given a set of spectra of different classes, we apply the SC algorithm producing a set of few basis vectors and a matrix of coefficients representing each original spectrum in the basis learned. We call each basis vector a prototype spectrum. Use of SC coefficients for MS data processing is promising because peaks of different width can be extracted. In the ideal case, any peak or combination of peaks which take place in sufficiently many spectra and represents a sizable contribution to a large portion of the dataset, will be represented using a SC coefficient. For each original spectrum we build a feature vector consisting of its coefficients. Second, the feature vectors are classified using SVM where the evaluation is done by mean of the double cross-validation, for more details see [ADM$^+$09].

In Section 2 we concisely describe the data investigated, as well as the sparse coding algorithm and the classification scheme used. In Section 3.1 we present the results of SC algorithm. Then, in Section 3.2, we show the classification results and compare them with the reference results of [ADM$^+$09]. Moreover, in Section 3.3 we provide a closer look at the SC results and propose clustering the spectra based on the SC coefficients. Section 4 concludes the paper.

2 Methods

2.1 Mass spectrometry data

The dataset used in this paper consists of matrix-assisted laser desorption ionization time-of-flight (MALDI-TOF) serum protein profiles of colorectal cancer patients and healthy individuals, first published in [dNMO$^+$06]. Colorectal cancer is one of the most common malignancies and remains a principal cause of cancer-related morbidity and mortality. Diagnosing colorectal cancer still requires a sensitive test relaying on easily accessible body fluids, like serum. After a preprocessing of spectra and outliers removal, described in [ADM$^+$09], we have 64 cancer and 48 control spectra of length 16331 points covering an mz (mass-over-charge) domain of 960–11163 Da.[1] For this paper we took only a part of the whole mz domain, namely 1100–3000 Da, which contains the most significant peaks for the cancer discrimination according to [dNMO$^+$06] and [ADM$^+$09]. A part of a spectrum in this domain consists of 4731 points. The final data is shown in Fig. 1

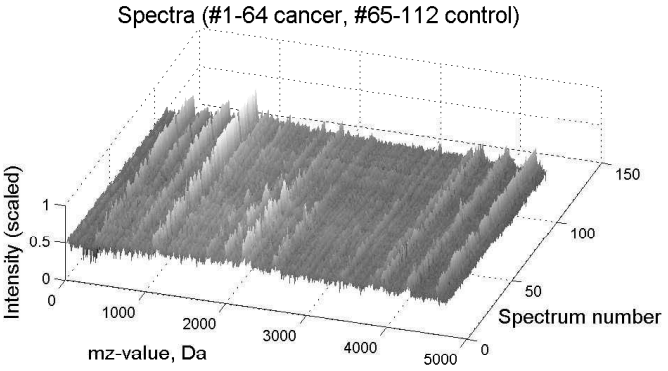


Figure 1: 64 cancer (with numbers 1-64) and 48 control mass spectrometry protein profiles.

2.2 Improved Sparse Coding algorithm with elastic-net regularization

For each original spectrum the SC algorithm calculates its coefficients in a basis expansion, where the basis vectors are learned from the data as follows.

We suppose that the dataset consists of R spectra of length L which belong to D classes ($D \ll R$) each characterized by common peaks at the same positions and with similar heights. Given a matrix $\mathbf{X} \in \mathbb{R}^{L \times R}$ with spectra in columns, the improved SC algorithm with an elastic-net regularization term represents each spectrum (a column of $\mathbf{X}$) in a self-

[1] The data is available at `http://www.math.uni-bremen.de/~theodore/MALDIDWT`.

taught sparse basis solving the following optimization problem:

$$\min_{\mathbf{B},\mathbf{S}} \quad \tfrac{1}{2}\,||\mathbf{X}-\mathbf{BS}||_{\mathrm{F}}^2 + \alpha \textstyle\sum_j ||S_j||_1 + \frac{\beta}{2}\sum_j ||S_j||^2\,, \tag{1}$$

$$\text{subject to } ||B_j||^2 \le \gamma, \tag{2}$$

with respect to a matrix $\mathbf{B} \in \mathbb{R}^{L\times L}$ of basis vectors and a matrix $\mathbf{S} \in \mathbb{R}^{L\times R}$ of the corresponding coefficients, where $||\cdot||_{\mathrm{F}}$ is the matrix Frobenius norm, $||\cdot||_1$ is the vector l_1-norm, and $||\cdot||$ is the standard euclidean norm; S_j and B_j denote the j-th column of $\mathbf{S}$ and $\mathbf{B}$, respectively. The hyperparameters of the optimization problem are the l_1-regularization parameter α, l_2-regularization parameter β and the boundary on the basis vectors norm γ.

The minimization problem (1) is solved in two steps. First, we learn the coefficients $\mathbf{S}$ keeping the basis fixed using the Feature Sign Search (FSS) algorithm minimizing (1) for a fixed $\mathbf{B}$, then for the learned coefficients we optimize the basis $\mathbf{B}$ using the Lagrange dual. For more details, see [LBRN06]. For motivation of using the elastic-net regularization instead of the original l_1-regularization, see [AKL$^+$09] and [ASKS09].

Finally, for each column X_j of $\mathbf{X}$ we have its sparse representation in the basis $\mathbf{B}$ with only a few basis vectors B_j ($j \in \mathcal{I}$) corresponding to non-zero rows S_j with indices $\mathcal{I}$.

2.3 Classification using Support Vector Machines with double cross-validation

After the SC algorithm produced a matrix $\mathbf{B}$ of basis vectors and a matrix $\mathbf{S}$ of coefficients, we classified the spectra where for each spectrum its coefficients (that is j-th column S_j of $\mathbf{S}$ for j-th spectrum) are used as features. The classification was performed using Support Vector Machine (SVM) of type C-SVM with the gaussian kernel with two-level grid search for the hyperparameters σ (the width of the gaussian kernel) and C (the C-SVM regularization parameter). The tested values are $2^{-4:2:16}$ (a grid with values from 2^{-4} to 2^{16} with a step 2^2) for σ and $2^{-4:2:12}$ for C at the first grid search level and $2^{-1:1:1}$ for both σ and C at the second level of grid search used for refinement. The simultaneous parameters selection and classifiers assessment was done by means of the double cross-validation (double CV) with the leave-one-out cross-validation (i.e. 112-fold) used for the outer loop and 10-fold cross-validation used for the inner loop, again as in [ADM$^+$09]. In this setting, the i-th step of the double CV scheme consists of two stages: (1) the choice of hyperparameters is done using 10-fold CV on all but the i-th spectrum optimizing CV recognition rate (the ratio of spectra correctly classified in CV), (2) a classifier with the chosen hyperparameters is trained using all but the i-th spectrum and applied to the i-th spectrum excluded at the first.

The following characteristics were calculated after the outer loop classification: total recognition rate or TRR (the ratio of correctly classified spectra), specificity, and sensitivity. Moreover, following [BST99] and [ADM$^+$09], we considered the number of support vectors (SV) as a measure of generalization performance of classifiers. The values of these characteristics have been compared with corresponding values reported in [ADM$^+$09], where the same dataset is used (except for the mz-domain as explained in section 2.1).

3 Results

3.1 Sparse coding representation

We applied to the matrix $\mathbf{X} \in \mathbb{R}^{4371\times 112}$ with spectra in columns the improved SC algorithm with an elastic-net penalty term with different values of parameters α (from 5 to 100 with a step 5) and γ (from 500 to 2500 with a step 500). The used value of the parameter corresponding to the l_2-penalty was $\beta = 10^{-10}$, which was selected as small as possible, as recommended in [AKL$^+$09].

For each pair of parameters (α, γ) we calculated the matrices $\mathbf{B}$ and $\mathbf{S}$ of basis vectors and corresponding coefficients. Recall that only basis vectors with indices $\mathcal{I}$ corresponding to non-zero rows of the coefficients matrix $\mathbf{S}$ are considered. In the following we refer to the computed basis vectors as the prototype spectra because each original spectrum is a linear combination of the basis vectors with weights equal to the corresponding coefficients. Fig. 2 shows the numbers of prototype spectra (sizes of $\mathcal{I}$) for all pairs of α and γ. As

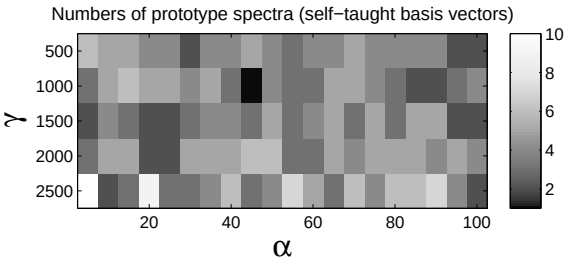


Figure 2: The numbers of prototype spectra for all pairs of α and γ considered.

can be seen from Fig. 2, for all parameters considered a spectrum can be represented by only a small number of prototype spectra (from 2 to 10). Once, for $\alpha = 45$ and $c = 1000$, only one prototype spectrum is produced. Interestingly, though it is natural to expect that the number of prototype spectra increases as α decreases (because α is a multiplier of the sparsity term), this effect can be hardly observed.

In the following we consider the results of the SC algorithm for $\alpha = 10$ and $\gamma = 1000$ selected as producing the best classification results (presented later in Section 3.2). Fig. 3 shows the five prototype spectra computed for these parameters. Fig. 4 depicts the non-zero rows of the matrix $\mathbf{S}$ (each row is normalized to have values from zero to one). One can visually observe that the 4-th and 5-th rows highly discriminate cancer (the first 64) and control (the last 48) spectra since their values are visually grouped into two clusters: corresponding to spectra with numbers 1-64 and 65-112. To confirm this observation and to evaluate the separation efficiency of the produced coefficients, we plot a Principal Components Analysis (PCA) score plot, see Fig. 4 which shows clear though not ideal separation between two classes. Here PCA is used only for visualization. In next section we present close to perfect classification results achieved using SVM. A PCA score plots

scores of the second principal component against scores of the first principal component and is often used for visualization of high-dimensional data. Fig. 4 demonstrates that the computed coefficients after a linear PCA-transformation allows one to clearly separate the groups of cancer and control individuals. This confirms the potential of using sparse coding coefficients for classifying cancer and control spectra.

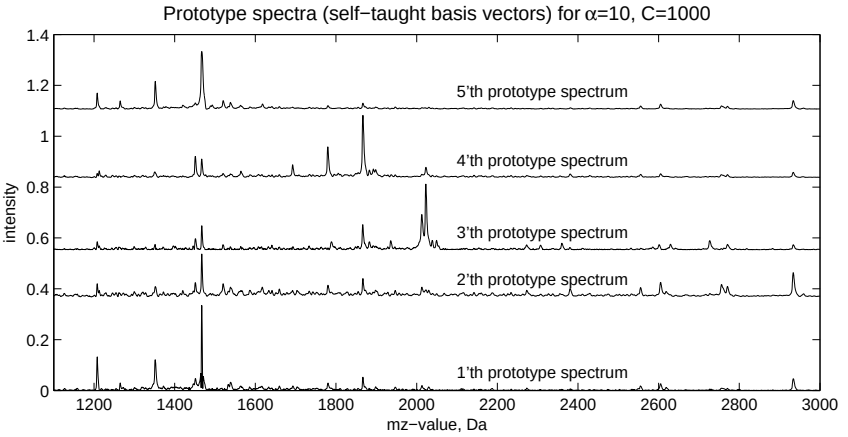


Figure 3: Prototype spectra (self-taught basis vectors) corresponding to non-zero coefficients extracted for $\alpha = 10$, $\gamma = 1000$, shifted in intensity for better visualization.

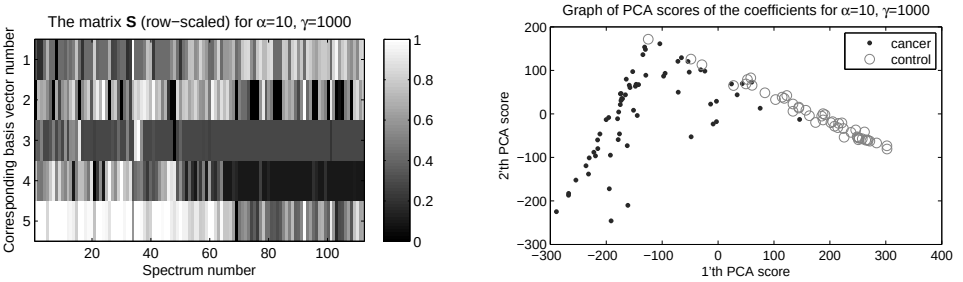


Figure 4: Left: non-zero coefficients (the matrix $\mathbf{S}$) used for representation of original spectra in the basis depicted on Fig. 3; right: a score plot showing the data (as usual, mean-corrected) projection onto the first two principal components.

3.2 Classification results

For each pair of the sparse coding parameters α and γ, we applied the SVM classification where the SVM hyperparameters selection and the classifiers assessment is done using the

		α						
		5	10	15	20	25	30	35
γ	500	94.64	93.75	93.75	90.18	88.39	90.18	90.18
	1000	92.86	**97.32**	91.07	87.50	89.29	89.29	91.07
	1500	90.18	91.07	90.18	89.29	90.18	89.29	90.18
	2000	92.86	96.43	94.64	91.07	87.50	91.07	91.07
	1500	93.75	91.07	87.50	95.54	91.96	92.86	91.07

Table 1: Total recognition rates for different α and γ for SVM classifiers using sparse coding coefficients, calculated through the double cross-validation. The best value (97.32%) is shown in bold.

double cross-validation, as described in Section 2.3 and, more detailed, in [ADM$^+$09]. The computed total recognition rates for all pairs of α and γ as well as the numbers of support vectors used are shown in Fig. 5.

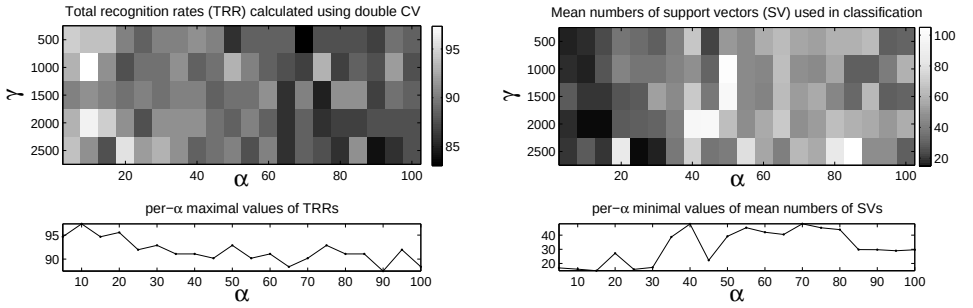


Figure 5: Classification results for different pairs of α and γ for SVM classifiers using sparse coding coefficients, calculated using the double cross-validation. Left: Total recognition rates; right: mean numbers of support vectors.

First, the achieved TRRs are quite high in comparison with the reference results. The best TRR is higher than the results of classification using the reduced-rank Linear Discriminant Analysis also evaluated using the double CV (92.6%) reported by [dNMO$^+$06] and is as hight as the results of the same classification procedure but applied on the DWT coefficients (97.3%) reported by [ADM$^+$09].

Although results presented in Fig. 5 are quite variable, there is a noticeable trend of decreasing TRR and increasing the mean number of SV (as α increases) that is better demonstrated by the plots of per-α maximal TRRs and per-α minimal mean number of SVs. For this reason, we showed in Table 1 and Table 2 the values of TRRs and the mean numbers of SV only for the first considered values of α (from 5 to 35). The best TRR is achieved for $\alpha = 10$ and $\gamma = 1000$ and is equal to 97.3% which is as high as reported by [ADM$^+$09] where DWT coefficients instead of sparse coding coefficients are exploited. The corresponding values of sensitivity and specificity are 96.9% and 97.9%, respectively. The most striking result to emerge from Table 2 is that the same classification efficiency is achieved using only 17 support vectors (corresponding to 15% of a training dataset of

		α						
		5	10	15	20	25	30	35
	500	16.8	18.6	23.5	37.1	33.4	29.6	41.2
	1000	17.9	16.6	27.2	42.2	36.9	37.4	42.6
γ	1500	28.0	20.3	20.3	27.2	28.4	54.1	44.8
	2000	18.2	15.9	14.9	28.6	30.1	31.4	41.5
	1500	20.2	28.1	20.2	85.8	15.8	17.2	38.7

Table 2: Mean numbers of SVM support vectors for different pairs of α and γ for SVM classifiers using sparse coding coefficients, calculated by means of the double cross-validation (the size of a training dataset is 111).

size 111) vs. 43 reported for the DWT-SVM procedure. It seems possible that the low numbers of the SV are due to the low number of features used in classification (less than 10 according to Fig. 2 vs. 300–600 for DWT and 1500–7000 for APPDWT as reported in [ADM$^+$09]).

As discussed in [ADM$^+$09], the number of support vectors is a proxy-measure of generalization performance of the classifiers. Any significant improvement of the generalization performance is very important in mass spectrometry-based proteomics, since the results should be reproducible when the data is prepared using different protocols, measured in different laboratories and in different conditions. All this leads to additional non-reducible variability in data and imposes high demands on the generalization performance of the exploited classifiers. From this point of view, the achieved advantage in the number of support vectors seems to be relevant and significant.

3.3 Closer look at the prototype spectra and sparse coding coefficients

Let us consider the 4-th and 5-th prototype spectra, see Fig. 6, since as conducted by means of visual inspection, their coefficients are the most discriminative between cancer and control groups. This choice is partially confirmed by the following fact. Considering the values of loadings of the first principal component (a direction of the largest variance) which are 0.1 (for the first prototype spectrum), -0.1 (second), -0.0 (third), -0.4 (fourth), -0.9 (fifth), we see that the 4-th and 5-th prototype spectra have the largest loadings, i.e. the largest contributions into a direction of the highest variance. Fig. 6 shows the scaled cancer and control mean spectra as well. The cancer (control) mean spectrum is manually attributed to the 4-th (5-th) prototype spectrum.

Fig. 6 shows that the prototype spectra are very similar to the per-class means spectra although they are extracted in an unsupervised manner, i.e. not using the labels of spectra.

It is interesting to compare Fig. 6 with Fig. 4a of [ADM$^+$09] showing the biomarker patterns reconstructed by the 1784 most discriminative APPDWT coefficients. In the region of 1100–2400 Da the prototype patterns are very similar to the biomarker patterns which is not surprising since they are similar to the per-class mean spectra. At the same time,

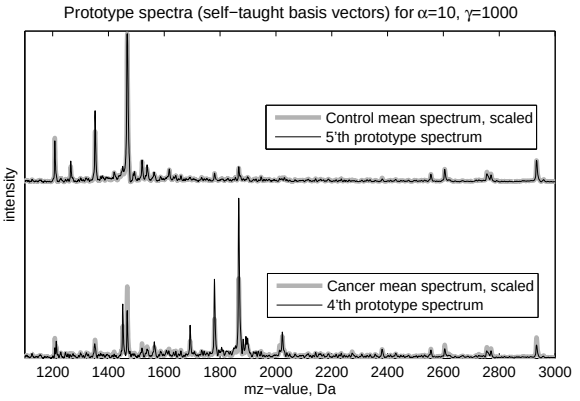


Figure 6: Forth and fifth prototype spectra (self-taught basis vectors) for $\alpha = 10$, $\gamma = 1000$ (shifted in intensity for better visualization) as well as the scaled cancer and control mean spectra.

note that the DWT biomarker patterns contain only a part of peaks presented in the mean spectra, which is especially noticeable in the region of 2400–3000 Da. This highlights the difference between local properties of wavelets and global (throughout the whole spectrum length) nature of the self-taught basis vectors.

An advantage of the self-taught sparse coding basis as compared to an APPDWT-induced dictionary is that it is learned in an unsupervised manner. Thus, the coefficients can be used not only for classification but also clustering of the spectra. For demonstration, we performed clustering of the spectra using High Dimensional Discriminant Analysis [BGS07]. The clusters number was set to 10 but the procedure automatically reduced it to 7; the used model is $[a_{ij}b_iQ_id_i]$; the scree-test threshold is 0.2, for explanations see [BGS07].

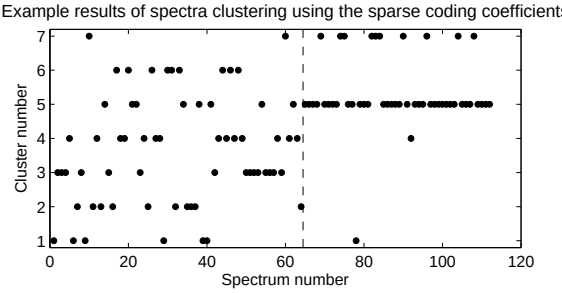


Figure 7: Example results of clustering of spectra using the sparse coding coefficients ($\alpha = 10$, $\gamma = 1000$): number of a cluster assigned to a spectrum against the spectrum number. A dash line is plotted for better visualization and separates cancer (spectra 1-64) from control (65-112) spectra.

Although Fig. 7 is shown mostly to demonstrate the potential of using sparse coding coefficients for spectra clustering, it is surprising to see that the control spectra are attributed

only to two clusters. At the same time, the cancer spectra are not so homogeneous and form five clusters that probably indicates the difference in protein profiles of the cancer samples due to several tumor stages used in the measurements or other factors. A special investigation of these results is required which is out of scope of this paper.

4 Conclusions

In this paper we proved the potential of the sparse coding classification scheme proposed by [RBL+07] using the improved sparse coding algorithm of [AKL+09] for applications in mass spectrometry. The combination of SC and SVM demonstrated the same accuracy as DWT-SVM procedure [ADM+09] but with a significantly higher generalization performance measured by the number of support vectors. We demonstrate that the SC coefficients can be used not only for classification but also for clustering of the spectra.

Acknowledgements. The author thanks Stefan Schiffler for his implementation of the Feature Sign Search algorithm.

References

[ADM+09] T. Alexandrov, J. Decker, B. Mertens, A. M. Deelder, R. A. E. M. Tollenaar, P. Maass, and H. Thiele. Biomarker discovery in MALDI-TOF serum protein profiles using discrete wavelet transformation. *Bioinformatics*, 25(5):643–649, 2009.

[AKL+09] T. Alexandrov, O. Keszoecze, D. A. Lorenz, S. Schiffler, and K. Steinhorst. An active set approach to the elastic-net and its applications in mass spectrometry. In *Proc. Int. Workshop on Sparsity in Signal Processing (SPARS)*, 2009. Available at `http://hal.archives-ouvertes.fr/docs/00/36/93/97/PDF/19.pdf`.

[ASKS09] T. Alexandrov, K. Steinhorst, O. Keszöcze, and S. Schiffler. SparseCodePicking: feature extraction in mass spectrometry using sparse coding algorithms. In *Proc. IFCS'09, submitted*, 2009. Available at `http://arxiv.org/abs/0907.3426`.

[BGS07] C. Bouveyron, S. Girard, and C. Schmid. High-dimensional data clustering. *Comp. Stat. Data Anal.*, 52:502–519, 2007.

[BST99] P. Bartlett and J. Shawe-Taylor. Generalization performance of support vector machines and other pattern classifiers. In B. Schölkopf, C. Burges, and A. Smola, editors, *Advances in kernel methods: SV learning*, pages 43–54. MIT Press, 1999.

[dNMO+06] M. de Noo, B. Mertens, A. Ozalp, M. Bladergroen, M. van der Werff, C. van de Velde, A. Deelder, and R. Tollenaar. Detection of colorectal cancer using MALDI-TOF serum protein profiling. *Eur. J. Cancer*, 42(8):1068–1076, 2006.

[LBRN06] H. Lee, A. Battle, R. Raina, and A. Y. Ng. Efficient sparse coding algorithms. In *Proc. NIPS'06*, pages 801–808, 2006.

[RBL+07] R. Raina, A. Battle, H. Lee, B. Packer, and A. Y. Ng. Self-taught learning: transfer learning from unlabeled data. In *Proc. 24th Int. Conf. on Machine learning (ICML)*, pages 759–766. ACM, 2007.

Semi-Supervised Learning for Improving Prediction of HIV Drug Resistance

Juliane Perner, André Altmann, Thomas Lengauer

Computational Biology and Applied Algorithmics
Max Planck Institute for Informatics
Campus E1.4
D-66123 Saarbrücken
{jperner,altmann,lengauer}@mpi-inf.mpg.de

Abstract: Resistance testing is an important tool in today's anti-HIV therapy management for improving the success of antiretroviral therapy. Routinely, the genetic sequence of viral target proteins is obtained. These sequences are then inspected for mutations that might confer resistance to antiretroviral drugs. However, interpretation of the genomic data is challenging. In recent years, approaches that employ supervised statistical learning methods were made available to assist the interpretation of the complex genetic information (e.g. *geno2pheno* and *VircoTYPE*). However, these methods rely on large amounts of labeled training data, which are expensive and labor-intensive to obtain. This work evaluates the application of semi-supervised learning (SSL) for improving the prediction of resistance from the viral genome.

1 Introduction

The Human Immunodeficiency Virus (HIV) is causing one of the most challenging infectious diseases. HIV is a retrovirus that mainly infects cells of the human immune system. Today there are about 25 antiretroviral drugs approved by the US Food and Drug Administration for treating HIV infections[1]. These drugs can be divided into different classes by their mechanism of action and the viral proteins they target. Reverse transcriptase inhibitors aim at prohibiting the synthesis of DNA from viral RNA by the viral protein reverse transcriptase (RT). This can currently be accomplished by nucleos(t)id analogs that lead to abortion of DNA synthesis after their incorporation. In constrast to these nucleoside reverse transcriptase inhibitors (NRTIs), non-nucleoside reverse transcriptase inhibitors (NNRTIs) bind to the viral RT and impair its flexibility. Integrase inhibitors prevent the integration of the viral DNA into the host genome by blocking the viral enzyme integrase. Finally, protease inhibitors (PIs) bind to the active site of the viral protease that cleaves precursor proteins into functionally units. The large number of drugs that are on the marketplace is required because the process of reverse transcription is error prone and therefore HIV eventually develops mutations in the targeted proteins that confer resistance against the applied drugs. These mutations enable the virus to replicate in the presence

[1] `http://www.fda.gov/oashi/aids/virals.html`

of a drug and are therefore selected evolutionarily. Unfortunately, these resistance mutations also confer drug resistance to drugs of the same class that were not applied yet, this phenomenon is termed *cross-resistance*. Resistance testing is an important tool in therapy management for choosing an appropriate drug regimen for the patient and consequently slow down disease progression to AIDS and death. There are two approaches to resistance testing. The first approach, *phenotyping*, affords a lab test that compares the viral replication of the virus of a patient with that of a wild type virus in the presence of the drug [Wa99]. The quotient of dosages of the drug that are required to cut the replication rate of the patient sample and the wild type, respectively, in half is called the *resistance factor*. The second approach, *genotyping*, amounts to sequencing the genes of the viral drug targets harbored by the virus variant predominating in the patient. These sequences have to be inspected for mutations that are related to drug resistance. Phenotyping is expensive and labor-intensive but delivers a single number per drug that is easy to interpret. Genotyping on the other hand is fast, cheap, and standardized, but the correct interpretation of the genetic sequence poses a major challenge. One way to address this problem is provided by knowledge-based approaches (expert systems) that apply classification rules. These rules are hand-crafted by experts based on literature, *in vitro* results, and clinical experience. Rule sets can be found, e.g. in Stanford's HIVdb [Rh99]. More systematic approaches employ supervised statistical learning methods to predict the resistance of a virus against drug based on the sequences of the genes coding for the target proteins, e.g. *geno2pheno* [Be03] and *VircoTYPE* [Ve07]. These supervised learning methods are trained on viral samples for which both, a genotypic test and a phenotypic test has been performed. However, for achieving a good performance a sufficient number of training samples is required (at least several hundred), which is in general expensive and labor-intensive to collect. Thus, especially, at the time shortly after the approval of a novel drug usually only a small number of genotype-phenotype pairs is available and consequently prediction methods lag behind in providing an assessment of these drugs. Since relevant parts of the HIV genome are routinely sequenced for diagnostic reasons, ample genotypic data without phenotypic measurements are available in clinical databases. This work focuses on the use of semi-supervised learning (SSL) for improving the prediction of drug resistance based on genotype-phenotype data together with available routinely collected sequence data. Recently, an SSL approach using unlabeled data from clinical routine for improved dimensionality reduction was applied to predict *in vivo* response to antiretroviral combination therapies [RAS09]. Section 2 provides a brief overview over the available data as well as supervised and semi-supervised methods that were applied. Section 3 presents the results, and section 4 gives a conclusion and an outlook.

2 Materials and Methods

2.1 Data

The genome sequences of the target proteins were available as amino acid sequences that had been aligned to the reference sequence HXB2. For the protease all 99 amino acids and

	NRTI						NNRTI	
drug name	ZDV	3TC	ddI	d4T	ABC	TDF	EFV	NVP
cutoff	0.9	1.37	0.37	0.25	0.54	0.25	0.7	0.67
susceptible (%)	49	57	50	53	44	42	61	50
$\|S_{labeled}\|$	1055	740	882	881	871	598	1037	880
$\|S_{drug}\|$	2717	5143	2329	3047	1225	1668	1264	1237
$\|S_{class}\|$	7887						2502	
	PI							
drug name	APV	ATV	IDV	LPV	NFV	SQV	DRV	TPV
cutoff	0.59	1.13	0.72	0.6	0.67	0.98	1.14	0.61
susceptible (%)	54	60	48	66	43	59	50	48
$\|S_{labeled}\|$	645	523	721	682	725	725	55	60
$\|S_{drug}\|$	290	320	756	1442	1075	687	0	0
$\|S_{class}\|$	4435							

Table 1: Description of the data. $|S_{labeled}|$ indicates the number of available genotype-phenotype pairs. The row *cutoff* lists the log_{10}(resistance factor) cutoff values used to dichotomize the continuous value into the categories *susceptible* (below the cutoff) and *resistant* (above the cutoff). The row susceptible (%) indicates the percentage of labeled data that was considered susceptible after dichotomization. The rows $|S_{drug}|$ and $|S_{class}|$ list the numbers of sequences that were obtained during exposure to the specific drug and drug class, respectively. Drugs: zidovudine (ZDV), lamivudine (3TC), didanosine (ddI), stavudine (d4T), abacavir (ABC), tenofovir disoproxil fumerate (TDF), efavirenz (EFV), nevirapine (NVP), (fos-)amprenavir (APV), atazanavir (ATV), indinavir (IDV), lopinavir (LPV), nelfinavir (NFV), saquinavir (SQV), darunavir (DRV), tipranavir (TPV).

for the RT only the first 220 amino acids were considered. The genotype-phenotype pairs were provided by the Arevir database [Ro06]. For every drug a different number of measured resistance factors (RFs) with corresponding genotype was available (see: Table 1). Unfortunately, most SSL approaches work for classification only, thus the continuous RFs were dichotomized to *susceptible* and *resistant* using a drug-specific cutoff. This cutoff was defined by the intersection of two Gaussian distributions, which the RFs display when plotted on logarithmic scale. The two Gaussian distributions represent the susceptible and resistant subpopulation as described in [Be03]. The cutoffs derived in this way for each drug are listed in Table 1. Sequences generated in diagnostic routine were taken from the EuResist database [Ro08] and constitute the unlabeled data used by the SSL methods. Sequences were categorized as to whether they were exposed to a specific drug (S_{drug}) or to a specific drug class (S_{class}) at the time the sample was obtained (see Table 1).

2.2 Statistical Methods

Semi-supervised learning methods operate on a labeled set $S_{labeled} = \{(x_1, y_1), (x_2, y_2)\ , ..., (x_n, y_n)\}$ and a set of unlabeled data $S_{unlabeled} = \{x_1^*, x_2^*, ..., x_m^*\}$, where x_i and y_i denote feature vector and corresponding label, respectively. The unlabeled data $S_{unlabeled}$

reveals information about the underlying data density. This knowledge can be exploited by SSL methods for generating improved prediction models compared to supervised methods. We can expect that SSL improves the prediction only, if labels show a tendency to be locally constant in input data space. This assumption is termed *smoothness assumption* and states that: if two points are located closely in data space, then their corresponding output is more likely to be similar (regression) or identical (classification). Consequently, the decision boundary derived by a SSL classification method should not cut through regions of high data density. Most of the semi-supervised methods perform transductive learning, i.e. the learner has to predict a set of labels $\{y_1^*, y_2^*, ..., y_m^*\}$ for the given unlabeled data $S_{unlabeled} = \{x_1^*, x_2^*, ..., x_m^*\}$. These unlabeled samples have to be available while training the method. According to the definition of transductive learning in [Zh07], transductive methods cannot handle unseen data. Thus, if a prediction for a new unlabeled sample x_{m+1}^* is needed, a new model using $x_{m+1}^* \cup S_{unlabeled}$ has to be trained for computing the label y_{m+1}^*. In contrast, inductive learners (e.g. classic supervised methods) yield a prediction function on the whole input space. Thus, inductive learners can also handle previously unseen data.

This section gives a brief overview over the SSL methods used in this work. The large number of different SSL approaches (for an overview see [CSZ06, Zh07]) was restricted to methods that are easily accessible (e.g. in the form of command line tools or available source code). As reference supervised methods support vector machines (SVMs) [CL01] were used for classification and regression, whereas regularized least-squares regression (RLSR) [SGV98] was used for regression only.

Transductive Support Vector Machine (tSVM) The standard soft margin SVM optimizes the following function:

$$\min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w}\| + C\sum_{i=1}^{N}\xi_i, \text{subject to } \xi_i \geq 0, y_i(\mathbf{w}x_i + b) \geq 1 - \xi_i, \forall_i \quad (1)$$

where $\mathbf{w}$ and b define the hyperplane, ξ_i are the slack variables that allow for misclassification and C is the cost parameter for misclassified examples. The tSVM aims at determining a separating hyperplane under consideration of the unlabeled samples, therefore equation (1) is extended in the following way:

$$\min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w}\| + C\sum_{i=1}^{n}\xi_i + C^*\sum_{j=1}^{m}\xi_j^*, \text{subject to}$$
$$\xi_i, \xi_j^* \geq 0, y_i(\mathbf{w}x_i + b) \geq 1 - \xi_i, y_j^*(\mathbf{w}x_j^* + b) \geq 1 - \xi_j^*, \forall_{i,j} \quad (2)$$

where the additional parameters ξ_j^* and C^* are the slack variables and the misclassification cost parameter for the unlabeled instances, respectively. Thus, the optimization problem in (2) differs from (1) in that the tSVM has to find a labeling $y_1^*, ..., y_m^*$ for the unlabeled data and a hyperplane $< \mathbf{w}, b >$ simultaneously. An approximative optimization procedure, which is required due to the complexity of the optimization problem, has been implemented in the software library SVM^{light} by Joachims [Jo99]. The approach begins with

a labeling of $x_1^*, ..., x_m^*$ based on the classification of an inductive SVM and a low weight C^* for the penalty for misclassified unlabeled data points. Then the labels of two randomly selected samples (one positive and one negative) are swapped. If the objective function is improved by that exchange of labels, then the switch is made permanent. This process is repeated until there are no more switches possible that yield an improved objective function. At this point the penalty for misclassified unlabeled data points C^* is increased and further labels are swapped to greedily improve the objective function. The iterative procedure stops when C^* exceeds a user defined value. Notice that applying the definition of transductive learning stated above, tSVMs are in fact inductive learners. However, the name tSVM originated from the intention to work only on the observed data [Zh07].

Low Density Separation The Low Density Separation (LDS) approach introduced in [CZ05] is a combination of a tSVM and a kernel based on graph distances that takes advantage of unlabeled data. The main idea of LDS is the construction of a density-sensitive kernel. This is achieved by representing the feature vectors x_i and x_j^* of labeled and unlabeled samples as nodes in a graph. Each node is connected to its k nearest neighbors by weighted edges, with the weight of an edge corresponding to the Euclidean distance of its endpoints. For all paths between two points the largest edge weight on the path is computed. The similarity of two points is then defined as a function of the minimum of these largest edge weights. The main idea behind the density-sensitive kernel for SSL is to enlarge the distance between points that are separated by regions of low data density. This kernel is used by a tSVM that applies gradient descent for finding a solution of a slightly modified version of equation (2) and is therefore termed ∇SVM. For a detailed description of the approach see [CZ05].

Co-Regularized Least-Squares Regression (coRLSR) In comparison with semi-supervised classification, semi-supervised regression is largely under-studied. However, in [Br06] an efficient semi-supervised regression method is introduced that is based on the idea of co-learning. Briefly, the approach assumes the existence of multiple views, i.e. distinct sets of features, which are equally well suited for predicting the outcome. CoRLSR trains one regularized least-squares regression (RLSR) for each view on the labeled data and the available unlabeled data are used to measure the disagreement of the models. By the optimization process the disagreement of models for different views is minimized. CoRLSR with two views has the following optimization function:

$$Q(\mathbf{c}) = \sum_{v=1}^{2} \left[\|y_v - L_v c_v\|^2 + \nu_v c_v^t L_v c_v \right] + \lambda_v \sum_{u,v=1}^{2} \|U_u c_u - U_v c_v\|^2 \qquad (3)$$

where $\mathbf{c} = (c_1, c_2) \in \mathbb{R}^{n_1} \times \mathbb{R}^{n_2}$ represents the trained model for each view, n_v is the number of training samples in each view, ν_v and λ_v control the influence of the regularization term $c_v^t L_v c_v$ and the penalty for disagreement between views, respectively. Furthermore, $L_v \in \mathbb{R}^{n_v \times n_v}$ is the kernel matrix for all labeled samples and the matrix $U_v \in \mathbb{R}^{m \times n_v}$ comprises the inner products of all combinations of unlabeled and labeled instances. The first term of the sum represents the optimization criterion for fitting a regularized least-squares model, the second part of the sum calculates the disagreement of two views on the

unlabeled samples. In the setting under study different views were not available. However, results in [BS04] demonstrated that for many problems the feature set can be randomly split into different views and, together with co-classification approaches, still outperform traditional single-view learning algorithms. Thus, in the experiments the amino acid positions of protease and RT were randomly distributed among two views.

2.3 Evaluation setup

The labeled data that were used to train the methods are denoted by L, where L is a subset of $S_{labeled}$. Method performance was then assessed on the remaining labeled data $S_{labeled} - L$. From this subset only the genome sequences were used in the training procedure of SSL approaches and those are referred to as $U_{lab} = \{x_i|(x_i, y_i) \in (S_{labeled} - L)\}$. The genetic sequences from routine diagnostic used by the SSL methods are referred to as S_{drug} and S_{class} for sequences exposed to the same drug and to the same drug class as the drug for which a prediction model is trained, respectively. The training data of the SSL methods comprised $L \cup U_{lab} \cup S_{drug}$ or $L \cup U_{lab} \cup S_{class}$ while the training data for the standard supervised methods were restricted to L. For each drug listed in Table 1 separate models were trained. Performance was computed by using 10-fold cross-validation, which means that for each cross-validation fold 90% of $S_{labeled}$ were attributed to L and the remaining 10% to U_{lab}. In addition to evaluating the usefulness of SSL methods (using different sets of unlabeled data) over standard supervised methods, the influence of the size of available labeled data on the prediction performance was studied. To this end, only a randomly chosen subset of L was actually used during training. The size of that subset was either 2.5%, 5%, 10%, 20%, 40%, 60%, 80%, or 100% of the size of L. The remaining samples from L were excluded from the respective analysis. All learning approaches except LDS applied a linear kernel. The amino acid sequences were encoded as described in [Be03]: one amino acid position was represented by 20 indicator variables, i.e. one indicator for each amino acid. Classification performance was assessed by calculating the the area under the receiver operating characteristics (ROC) curve (AUC). Regression performance was measured as mean squared error (MSE) between predicted log_{10}(RF) and measured log_{10}(RF). The model parameters of the methods (see section 2.2) were optimized during the 10-fold cross-validation. Sets of different parameters were tested for each fold and the set performing best was used for performance computation.

3 Results and Discussion

3.1 Classification

Figure 1 summarizes the classification results by depicting the performance of all methods for all drugs when 10% and 100% of L were used during training, respectively. These fractions of L were selected for reflecting the amount data typically available shortly after

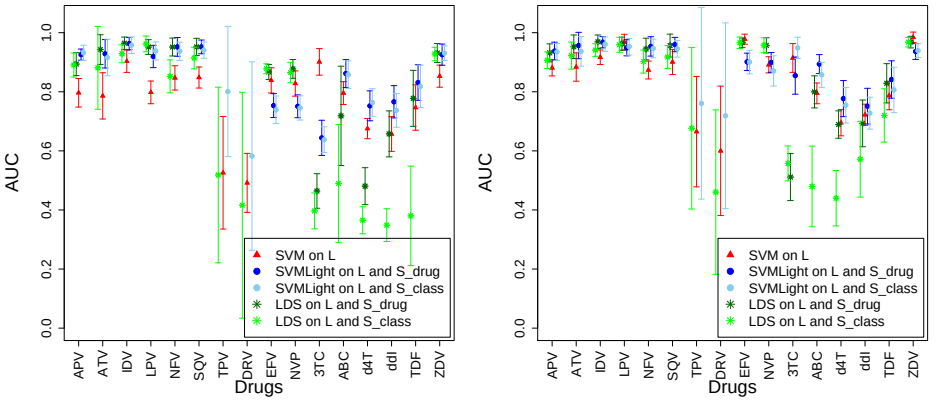


Figure 1: Mean area under the ROC curve (AUC) for 10% (left) and 100% (right) of the labeled data for the reference method and both SSL classification methods trained with the additional unlabeled sets S_{drug} and S_{class}, respectively. Whiskers indicate the standard deviation computed via 10-fold cross-validation.

approval of a novel drug and the amount of maximally available data. Figure 2 shows the AUC for varying volume of labeled data for three drugs representing the three drug classes. For protease inhibitors SSL brought a consistent benefit over supervised learning. With only 10% of L used during training all SSL methods performed at least as well as the supervised SVM for all PIs. Usage of the smaller unlabeled set S_{drug} brought a slight benefit over S_{class}. When 100% of L were used the gain in performance of SSL methods over the SVM was less pronounced. TPV and DRV are novel drugs in the class of PIs. The amount of labeled data is small and none of the available sequences were ever exposed to these drugs (Table 1). For both drugs SSL classification models did not show an improvement over the supervised SVM classification. However, this lack of observable improvement might be a consequence of the low number of instances available for assessing the performance. This assumption is supported by the large standard deviation of the AUC. For the two NNRTIs the results were less consistent. For EFV the SSL version in SVM^{light} did not show an improvement over classical supervised learning for any fraction of labeled data (Figure 2). For NFV the use of the SSL routine in SVM^{light} resulted in a clearly lower performance only when a small volume (10%) of labeled data was used. LDS performed for both drugs as well as or slightly better than the supervised SVM. For the group of NRTIs the results were even more diverse. While for ZDV and a small volume of labeled data the SSL methods displayed an improvement over the supervised SVM, for 3TC both SSL approaches drastically corrupted the performance. This difference might be explained by the different resistance profiles of the drugs. For 3TC one amino acid exchange is sufficient to confer complete resistance, while for ZDV several mutations are necessary. NRTIs are usually given in pairs to the patients, thus viruses that were exposed to 3TC were also exposed to other NRTIs with more complicated resistance patterns (e.g. ZDV). As a consequence, the data density does not reflect the labeling of 3TC resistance, which is a violation of the smoothness assumption. This finding is supported by the fact

that LDS with its density-sensitive kernel performs worse than SVM^{light} for 3TC. For the remaining four NRTIs the classification performance was worse compared to the remaining drugs. This is related to the small ranges of resistance factors that are observed for these drugs. Consequently, the Gaussian densities for the susceptible and resistant subpopulations are heavily overlapping, and therefor the computations of an appropriate cutoff is difficult. However, SVM^{light}performed better than the supervised SVM.

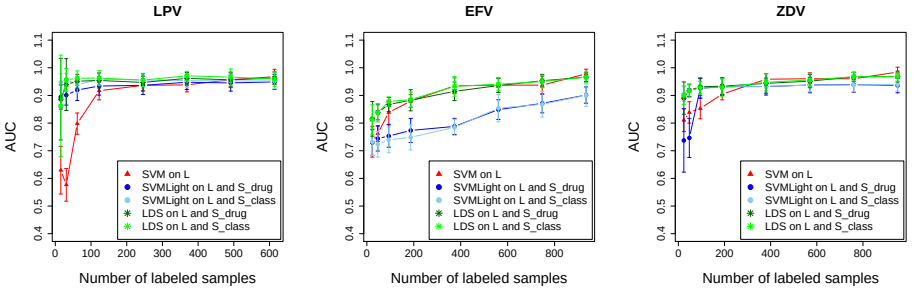


Figure 2: Development of the area under the ROC curve (AUC) for different volumes of labeled data for the reference method and both SSL classification methods trained with the additional unlabeled sets S_{drug} and S_{class}, respectively, for three drugs. Lopinavir (left), efavirenz (middle), and zidovudine (right). Whiskers indicate the standard deviation computed via 10-fold cross-validation.

3.2 Regression

Figure 3 depicts the performance of SVM, RLSR, and coRLSR for all drugs when 10% and 100% of L were used during training, respectively. Figure 4 shows the detailed development of the mean squared error for increasing volume of labeled data for LPV, EFV, and ZDV. CoRLSR, the only semi-supervised regression method tested in this study, did not improve the performance over RLSR or support vector regression. Moreover, the set of unlabeled data used during training (S_{drug} or S_{class}) did not play any substantial role in the performance of coRLSR. CoRLSR performed worse than RLSR for 3TC. As a consequence of dividing the amino acid positions among the two views for coRLSR, only one view had access to the single amino acid position that causes 3TC resistance. This fact violates the assumption that both views are sufficient for correct predictions and therefore lead to a significantly decreased performance compared to RLSR.

4 Conclusion and Outlook

Semi-supervised learning has the capability to improve the prediction of drug resistance from important regions in the HIV genome. The classification methods displayed a clear benefit over classical supervised learning for most drugs when only few labeled training

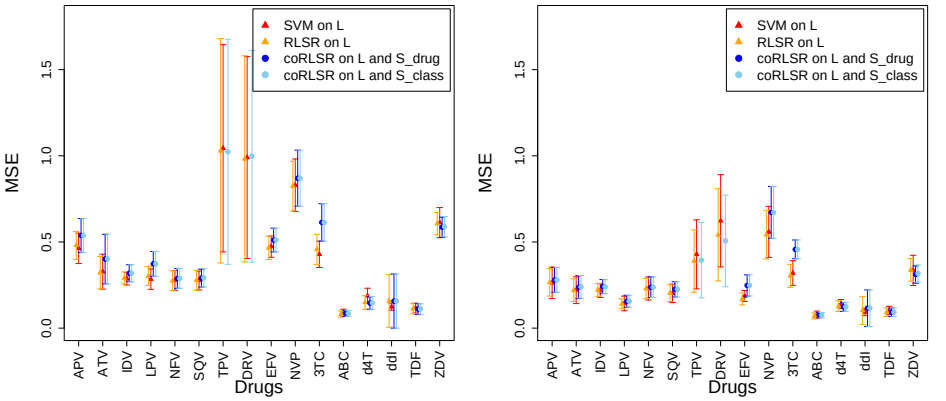


Figure 3: Mean squared error for 10% (left) and 100% (right) of the labeled data for the reference methods and coRLSR trained with the unlabeled sets S_{drug} and S_{class}, respectively. Whiskers indicate the standard deviation computed via 10-fold cross-validation.

samples were available. PIs, a drug class with strong cross-resistance between drugs, benefited the most from the use of SSL. The results support that SSL methods are suitable for improving prediction of drug resistance for novel drugs in established drug classes, such as daraunavir and tipranavir. Generally, it is not clear whether SSL is helpful for drugs belonging to novel drug classes (e.g. integrase inhibitors), because only few sequences harboring resistance mutations are available and SSL can also corrupt the classification results as seen for 3TC. The only semi-supervised regression model coRLSR could not improve the performance over the supervised methods.

Acknowledgments

The authors thank the Arevir project for providing training data in form of genotype-phenotype pairs, the EuResist project (IST-4- 027173-STP) and its coordinators Francesca Incardona and Maurizio Zazzi for providing a large number of sequences from routine diagnostics, and Ulf Brefeld for sharing the coRLSR code.

References

[Be03] Beerenwinkel, N. *et al.*: Geno2pheno: estimating phenotypic drug resistance from HIV-1 genotypes. In: Nucleic Acids Research, Vol. 31, No. 13, 2003. Oxford University Press, 2003; pp. 3850-3855.

[Br06] Brefeld, U. *et al.*: Efficient Co-Regularized Least Squares Regression. In: Proceedings of the 23rd International Conference on Machine Learning, Pittsburgh, 2006. pp.137 - 144.

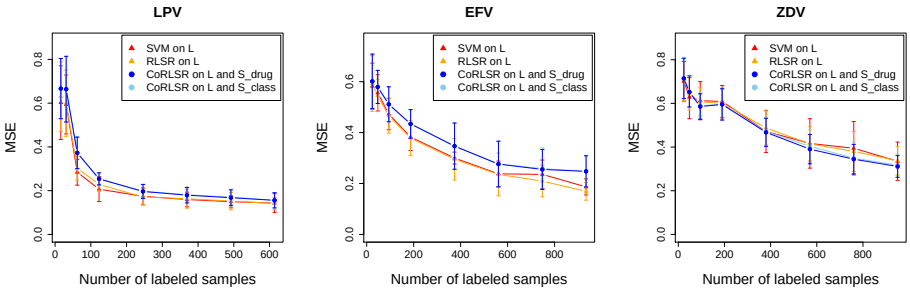


Figure 4: Development of the mean squared error for different volumes of labeled data for the reference methods and coRLSR trained with the unlabeled sets S_{drug} and S_{class}, respectively, for three drugs. Lopinavir (left), efavirenz (middle), and zidovudine (right). Whiskers indicate the standard deviation computed via 10-fold cross-validation.

[BS04] Brefeld, U.; Scheffer, T.: Co-EM Support Vector Learning. In: Proceedings of the 21st International Conference on Machine Learning, Banff, 2004. pp.121-128.

[CL01] Chang C.-C.; Lin C.-J.: LIBSVM: a library for support vector machines. 2001. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`

[CSZ06] Chapelle, O.; Schölkopf, B.; Zien, A.: Semi-Supervised Learning. The MIT Press, Cambridge, Massachusetts, 2006.

[CZ05] Chapelle, O.; Zien, A.: Semi-Supervised Classification by Low Density Separation. In: Proceedings of the 10th International Workshop on Artificial Intelligence and Statistics, Barbados, 2005. pp.57-64.

[Jo99] Joachims, T.: Transductive Inference for Text Classification using Support Vector Machines. In: Proceedings of the 16th International Conference on Machine Learning, Bled, 1999. pp.200-209.

[RAS09] Rosen-Zvi, M.; Aharoni, E.; Selbig, J.: HIV-1 Drug Resistance Prediction and Therapy Optimization: A Case Study for the Application of Classification and Clustering Methods. M. Biehl *et al.* (Eds.): Similarity-Based Clustering, LNAI 5400, 2009. pp.185-201.

[Rh99] Rhee, SY *et al.*: Human immunodeficiency virus reverse transcriptase and protease sequence database. In: Nucleic Acids Research, 2003, Vol. 31, No. 1. Oxford Press, 2003. pp. 298-303.

[Ro06] Roomp, K. *et al.*: Arevir: a secure platform for designing personalized antiretroviral therapies against HIV. In: Lecture Notes in Computer Science, Vol. 4075, Berlin/Heidelberg. Springer, 2006. pp.185-194.

[Ro08] Rosen-Zvi, M. *et al.*: Selecting anti-HIV therapies based on a variety of genomic and clinical factors. In: Bioinformatics, Vol. 24, Issue 13, 2008. Oxford Journals, 2008. pp. i399-i406.

[SGV98] Saunders, C.; Gammerman, A.; Vovk, V.: Ridge regression learning algorithm in dual variables. In: Proceedings of the 15th International Conference on Machine Learning, San Francisco, 1998. Morgan Kaufmann, 1998. pp. 515-521.

[Ve07] Vermeiren, H. *et al.*: Prediction of HIV-1 drug susceptibility phenotype from the viral genotype using linear regression modeling. In: Journal of Virological Methods, Vol. 145, Issue 1, 2007. Elsevier Science, 2007. pp. 47-55.

[Wa99] Walter, H. *et al.*: Rapid, phenotypic HIV-1 drug sensitivity assay for protease and reverse transcriptase inhibitors. In: Journal of Clinical Virology, Vol. 13, Issue 1, 1999. Elsevier Science, 1999. pp. 7180.

[Zh07] Zhu, X.: semi-supervised learning Literature Survey. Webpage: `http://pages.cs.wisc.edu/~jerryzhu/research/ssl/semireview.html`, accessed Mai 2008.

CUDA-based multi-core implementation of MDS-based bioinformatics algorithms

Thilo Fester
Martin-Luther-University Halle-Wittenberg
thilo.fester@student.uni-halle.de

Falk Schreiber
IPK Gatersleben & Martin-Luther-University Halle-Wittenberg
schreibe@ipk-gatersleben.de

Marc Strickert
IPK Gatersleben
stricker@ipk-gatersleben.de

Abstract: Solving problems in bioinformatics often needs extensive computational power. Current trends in processor architecture, especially massive multi-core processors for graphic cards, combine a large number of cores into a single chip to improve the overall performance. The Compute Unified Device Architecture (CUDA) provides programming interfaces to make full use of the computing power of graphics processing units. We present a way to use CUDA for substantial performance improvement of methods based on multi-dimensional scaling (MDS). The suitability of the CUDA architecture as a high-performance computing platform is studied by adapting a MDS algorithm on specific hardware properties. We show how typical bioinformatics problems related to dimension reduction and network layout benefit from the multi-core implementation of the MDS algorithm. CUDA-based methods are introduced and compared to standard solutions, demonstrating 50-fold acceleration and above.

1 Introduction

Bioinformatics is faced with accelerating increase of data set sizes originating from powerful high-throughput measuring devices. The implementation of computational intensive tasks in parallel technology is one of the key solutions to time-efficient data processing. Today, often compute jobs are performed on cluster computers or on large multi-core servers to take advance of parallelization. We will discuss an evolving path to provide work-efficient, parallel and desktop-suitable solutions based on acceleration by graphics processing units using the compute unified device architecture (CUDA) for computation on commonly available graphics processing units (GPU). High-throughput multi-dimensional scaling (HiT-MDS) is a versatile tool for biological data analyses that is systematically transferred to the GPU for taking advantages of the massively parallel hardware architecture for scientific computing.

1.1 Multidimensional Scaling

Multidimensional scaling (MDS) is a data processing method suitable for addressing several analytical purposes: (i) for dimension reduction of vector data, providing a nonlinear alternative to the projection to principal components; (ii) for the reconstruction of a data dissimilarity matrix of pairwise relationships in the Euclidean output space; (iii) for conversion of a given metric space, such as data compared by Manhattan distance, into Euclidean space, (iv) for dealing with missing data relationships using zero force assumption. These features make MDS a valuable tool for the analysis of large data tables and for dealing with (partial) information about data relationships [IMO09, SSUS07, TO05]. We focus on two examples of MDS application: one is related to dimension reduction in gene expression time series data, the other one is related to network layout from adjacency information.

1.2 GPGPU Programming with NVIDIA CUDA

In the last ten years general purpose computing on graphics processors became more and more important. Higher memory bandwidth, increasing (parallel) floating point performance compared to CPUs and rising memory capacities as well as low costs get attractive to scientists because of impressive speed up factors of up to several hundred times in different CUDA based analyses [BK09, GHGC09, JK09, LKPM09]. Following the trend to take advance of a little 'supercomputer at home', approaches with massive parallelism on the GPU have been implemented in scientifically important tools such as MATLAB or FORTRAN libraries [FJ07, GDD08].

Because of the development from simple graphics devices into highly parallel, multithreaded many-core processors, today GPUs are very appropriate to solve problems transformable into data parallel instructions operation. That is, the more independent subsequent instructions are the lower is the communication overhead which usually causes performance loss. By massive parallel operations memory access latency can even be avoided by in-place recalculations instead of accessing big data caches. For that purpose, parallel instructions are embedded into a logical grid of thread blocks, which is mapped to scalar processors by the instruction unit of a multiprocessor as illustrated in Figure 1. This architecture is called SIMT (single-instruction, multiple-thread) which is similar to the well-known SIMD (single-instruction, multiple-data) concept [Cor08].

For controlling the GPU computation CUDA was developed as a hybrid CPU-GPU interaction model. The above mentioned single-instruction functions are called from a CPU thread (referred to as host in the following). Such functions are called kernels whose instructions and amount of executed threads can be specified by the coder. As shown by Ryoo et al. not only aiming at best local acceleration, but also the distribution of threads within the grids and blocks can significantly influence the performance [RRS+07].

Another important feature of NVIDIA's CUDA enabled devices is the heterogeneous memory (see Figure 1). The large global memory reaching gigabytes of capacity contains small

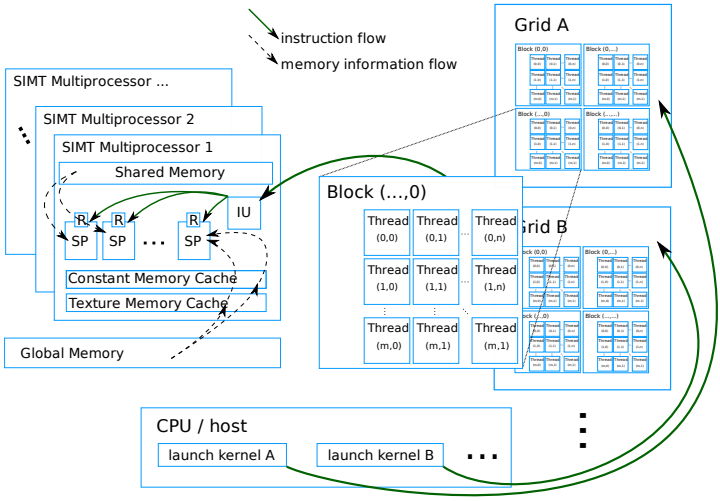


Figure 1: CUDA memory, processor and programming model. IU - Instruction Unit, R - Register, SP - Scalar processor

subsets of two cached memory types, texture and constant memory, which are available in every thread and accessible as fast as registers after being cached. The on-chip shared memory is available within all threads of a block. In case of no bank conflicts, it is as fast as register. Bank conflicts occur if two threads try to read the same memory contemporaneously, which is then serialised [Cor08]. Nearly all implementations use these fast memories to achieve communication between threads. This leads to massive performance gain compared to the usage of global memory or the even much worse communication delay via host control [CMS08, Cor08, Sel08]. To get the best out of global memory Seland pointed out to access contiguous (coalesced) memory and reported speed up factors of up to ten by using this technique [Sel08].

Furthermore NVIDIA declares CUDA as an extension to the C programming language and targets to simplify parallel computation. Hence working with CUDA is quite intuitive, and there is a low learning curve even without much knowledge about graphics hardware or OpenGL. This makes CUDA attractive for tool development for bioinformatics tasks.

2 Methods

This section is organised as follows: First we describe the multi-dimensional scaling (MDS) method. Then the implementation and optimisation of the algorithms in CUDA are described in detail. We close this section with two important applications of MDS: gene expression analysis and automatic layout of biological networks.

2.1 Multi Dimensional Scaling

Very intuitive visualisation of relationships between different data records can be obtained by reconstructing these relationships as pairwise distances in the usual Euclidean 2D plane or 3D space. Usually data projections to the principal components are used for that purpose, referred to as PCA projection. However, PCA is restricted to linear mappings of high-dimensional data, thereby focusing on directions of maximum Euclidean variance. A more natural goal is to obtain a low-dimensional display of a Euclidean space that reflects most faithfully the similarities among the source data.

In principle, this goal can be reached by using multi-dimensional scaling (MDS) techniques. In classical approaches, distances between the reconstructed low-dimensional points should be maximum similar to distances between the original data records. This strict optimisation task can be very hard, though, because of ambiguous compromise solutions for complex source relationships being rendered into a low-dimensional target space. Most MDS methods define quite stringent cost functions, such as least squares approaches targeting identity of the distances between the reconstructed point locations and the distances of corresponding input data.

Alternatively, Pearson correlation $\mathsf{r} \in [-1;1]$ can be computed between the distance matrices $\mathbf{D} = (\mathrm{d}_{ij})_{i,j=1\ldots n}$ and $\hat{\mathbf{D}} = (\hat{\mathrm{d}}_{ij})_{i,j=1\ldots n}$ of input data and of reconstructed points, respectively, by

$$\begin{aligned} \mathsf{r}(\mathbf{D},\hat{\mathbf{D}}) &= \frac{\sum_{i<j}^{n}\left(\mathrm{d}_{ij}-\mu_{\mathbf{D}}\right)\cdot\left(\hat{\mathrm{d}}_{ij}-\mu_{\hat{\mathbf{D}}}\right)}{\sqrt{\sum_{i<j}^{n}\left(\mathrm{d}_{ij}-\mu_{\mathbf{D}}\right)^2}\cdot\sqrt{\sum_{i<j}^{n}\left(\hat{\mathrm{d}}_{ij}-\mu_{\hat{\mathbf{D}}}\right)^2}} =: \frac{\mathscr{B}}{\sqrt{\mathscr{C}\cdot\mathscr{D}}} \\ \text{with}\quad \mu_{\dot{\mathbf{D}}} &= \frac{2}{n\cdot(n-1)}\cdot\sum_{i<j}^{n}\dot{\mathrm{d}}_{ij}\,, \quad \dot{\mathbf{D}}\in\{\mathbf{D},\hat{\mathbf{D}}\}\,,\dot{\mathrm{d}}_{ij}\in\{\mathrm{d}_{ij},\hat{\mathrm{d}}_{ij}\}\,. \end{aligned} \tag{1}$$

This correlation approach allows infinitely many more solutions than strict identity optimisation, while ensuring maximum correlation between source and target distances. Relaxation of the optimisation procedure is explained by the invariance of Pearson correlation against rescaling of vectors by a factor and against baseline shifts by an additive offset. The following method, called high-throughput multidimensional scaling (HiT-MDS), describes how correlation is used to help alleviate the optimisation task of finding proper low-dimensional point locations.

Referring to source vectors $\mathbf{x}^i \in \mathbf{X}$, target vectors $\hat{\mathbf{x}}^i \in \hat{\mathbf{X}}$ and their respective dimensions q and $\hat{q}$, the correlation $\mathsf{r}(\mathbf{D},\hat{\mathbf{D}})$ between entries of the source distance matrix $\mathbf{D}$ and the reconstructed distances $\hat{\mathbf{D}}$ is maximised by minimising the following embedding cost function:

$$\mathrm{s} = -\mathsf{r}\circ\hat{\mathbf{D}}\circ\hat{\mathbf{X}} \quad\Rightarrow\quad \frac{\partial \mathrm{s}}{\partial \hat{x}_k^i} = -\sum_{j=1\ldots n}^{j\neq i}\frac{\partial \mathsf{r}}{\partial \hat{\mathrm{d}}_{ij}}\cdot\frac{\partial \hat{\mathrm{d}}_{ij}}{\partial \hat{x}_k^i} \to 0,\ i=1\ldots n \tag{2}$$

Locations of all points $\hat{\mathbf{x}}^i$ in the target space induce pairwise distances and, consequently,

correlations between source and target distances. These locations are obtained by gradient descent on the stress function s using the chain rule. The derivatives in Equation 2 are [SSUS07]

$$\begin{aligned}\frac{\partial \mathsf{r}}{\partial \hat{\mathrm{d}}_{ij}} &= \frac{(\mathrm{d}_{ij} - \mu_{\mathbf{D}}) - \frac{\mathscr{B}}{\mathscr{D}} \cdot (\hat{\mathrm{d}}_{ij} - \mu_{\hat{\mathbf{D}}})}{\sqrt{\mathscr{C} \cdot \mathscr{D}}} \\ \frac{\partial \hat{\mathrm{d}}_{ij}}{\partial \hat{x}^i_k} &= (\hat{x}^i_k - \hat{x}^j_k) \,/\, \hat{\mathrm{d}}_{ij} \quad \text{for Euclidean} \quad \hat{\mathrm{d}}_{ij} = \sqrt{\sum\nolimits_{l=1}^{\hat{q}} (\hat{x}^i_l - \hat{x}^j_l)^2}\,.\end{aligned}$$

While for intuitive plotting results target distances $\hat{\mathrm{d}}_{ij}$ are usually Euclidean, input distances can be mere dissimilarities, such as mirrored Pearson correlation $\mathrm{d}_{ij} = (1 - \mathsf{r}(\mathbf{x}^i, \mathbf{x}^j))$ or powers of which. These correlations between data vectors must not be confused with the target value r in the correlation-based cost function optimisation in Equation 2 of HiT-MDS.

Two major revisions are made to the previous version of HiT-MDS described in [SSUS07].

First, the update replaces the specific value of the cost function derivative in Equation 2 by the sign $\mathrm{sgn}(\partial \mathrm{s}/\partial \hat{x}^i_k)$. This forces updates, irrespective of the order of magnitude of the derivative for maintaining a constant convergence process. The effective rate of convergence is controlled by a single factor only, the learning rate γ_t, decreasing in time. Thus, an atomic update quantity of the k-th component of the i-th reconstruction point at time point t is computed by

$$\Delta_t \hat{x}^i_k = -\gamma_t \cdot \mathrm{sgn}\left(\frac{\partial \mathrm{s}}{\partial \hat{x}^i_k}\right)\,, \quad \gamma_t \to 0 \quad \text{for} \quad t \to t_{\max}\,. \tag{3}$$

Convergence is forced by driving the learning rate monotonously to zero, in the limit of maximum cycles $t_{\max}+1$. In practice, the learning rate starts at $\gamma_0 = 0.1$ and gets linearly decreased to zero. This update scheme is very robust against the choice of the learning rate and turns out to yield excellent results.

Secondly, batch optimisation is realised. This means that updates from all pairs of data records are integrated before being applied synchronously to the reconstructed points. This strategy can be formally expressed as operations on distance matrices and, hence, efficiently parallelised. Illustrative MATLAB/Octave and R implementations with vectorised code as well as CUDA codes are available online [Hit].

A general formulation of the point reconstruction procedure is given in Algorithm 1. Much of the work is actually done in line 8 of the program. Apparently, the depicted algorithm is specialised in the task of fast reconstruction of a given dissimilarity matrix $\mathbf{D}$, thereby depending only on the target dimension, adaptation rate, and the number of cycles.

One of the main challenges of transferring the general algorithm to CUDA is an efficient use of memory and threads, which is detailed in the next sections. Another important issue to be discussed is the handling of adjacency matrices for being processed by the HiT-MDS algorithm.

Algorithm 1 General HiT-MDS algorithm

Initialise $\hat{x}_k^i$ randomly from the unit interval
for $t \leftarrow 1 \dots t_{\max}$ {iterations} **do**
  Calculate distance matrix $\hat{\mathbf{D}}$ of all $\hat{\mathbf{x}}$, including $\mathscr{B}, \mathscr{C}, \mathscr{D}$, and $\mu_{\hat{\mathbf{D}}}$
  Calculate update rate $\gamma_t \leftarrow \gamma_0 \cdot (1 - t/(t_{\max} + 1))$
  for $k \leftarrow 1 \dots \hat{q}$ {each target dimension} **do**
    reset k-th dimension update vector $\boldsymbol{y} \leftarrow \mathbf{0}$
    for $i \leftarrow 1 \dots n$ {each target point} **do**
      $y_i \leftarrow \Delta_t \hat{x}_k^i$ (using Equation. 3)
    end for
    for $i \leftarrow 1 \dots n$ {apply integrated update to each target point} **do**
      $\hat{x}_k^i \leftarrow \hat{x}_k^i + y_i$
    end for
  end for
end for

2.2 Implementation on CUDA

As depicted in equations 1, 2 and 3 the essential work of HiT-MDS is done by calculating the Pearson correlation coefficient $\mathsf{r}(\mathbf{D}, \hat{\mathbf{D}})$. Therefore, three intensive summations including mean value computation as well as the Euclidean distances have to be computed. These two tasks are very well suited for being transformed into parallel problems, as described following. We will point out the way of theoretical parallelization complexities and CUDA specific implementation details. Additionally, we included a degree based and a second all-pairs-shortest-path based algorithm to enhance the graph distance interpretation possibilities of HiT-MDS for creating network layouts.

All pairwise distances are stored in a half n^2-matrix and accessed by a function $getPos(x, y) = row\_coord[y] + x$ for a graph $G = (V, E)$, $(x, y) \in E$. row_coord contains pre-calculated coordinates of starting points for each row of the given half matrix. On the GPU this is realised as an array in fast constant memory to provide best access times.

The Prefix Reduction or partial-sums problem is a well understood algorithmic approach to maintain partial sums of a given array $A[1..n]$. It is specified for elements $A[i]$ to come from an arbitrary group H containing at least 2^δ elements. For the cell-probe model with b-bit cells a problem complexity of $\Omega = \left(\frac{\delta}{b} \cdot \lg n\right)$ was proven [PD04].

For parallel implementations, it was shown that the naïve algorithm's time complexity is $\mathcal{O}(\log n)$ performing $\mathcal{O}(n \log_2 n)$ addition operations. Furthermore, work efficient implementations (Algorithm 2) perform only $\mathcal{O}(n)$ addition operations [HSO07].

To get additional speed improvements, it is necessary to take advantage of the multiprocessors shared memory. All threads running in the same block have communication access to the same shared memory. In this case, communication means to copy all elements known

Algorithm 2 Work-efficient partial-sum algorithm

1: **for** $d \leftarrow 0 \ldots \log_2 n - 1$ **do**
2: **for** $k \leftarrow 0 \ldots n-1$ by 2^{d+1} in parallel **do**
3: $A[k + 2^{d+1} - 1] \leftarrow A[k + 2^d - 1] + A[k + 2^{d+1} - 1]$
4: **end for**
5: **end for**

by a thread into a smaller shared memory array. As mentioned above, shared memory is substantially faster than global memory.

To avoid bank conflicts, we use the blockIDs and threadIDs to calculate memory addresses of elements in shared memory that a thread adds up, schematically given as

$$sum = shared\,[threadID] + shared\,[threadID + blockDim/2] \tag{4}$$

Thus, with a limitation to the maximum number of threads per block of 512, caused by CUDA constraints, one can add 1024 elements per block. The result is written back to global memory and is source of the next loop. Therefore, in every step of outer for-loop of Algorithm 2, there are $2 \cdot k$ global memory accesses. To take advance of coalescing memory, we address the elements read from global memory by a similar idea as in formula 4.

Euclidean Distances computation time complexity is in $\mathcal{O}\left(\frac{n^2}{p}\right)$. We reached acceleration factors of more than 20 up to 30 by using a simple kernel according to Algorithm 3. This approach uses the CUDA built-in register variables $id.x$ and $id.y$ to find out the virtual location of an active thread.

Algorithm 3 Parallel pairwise Euclidean distances with thread IDs in $\hat{q}$-dimensional space

1: **for** $i \leftarrow 0 \ldots n^2$ in parallel **do**
2: $\hat{\mathrm{d}}_{id.x,id.y} \leftarrow \sqrt{\sum_{l=1}^{\hat{q}} (\hat{x}_l^{id.x} - \hat{x}_l^{id.y})^2}$
3: **end for**

Floyd-Warshall Algorithm According to use MDS as graph layout tool, a simple approach to get more information out of sparse graphs is to compute extra distances from existing graph edges by finding all node pairs shortest path. The Floyd-Warshall algorithm was designed with this in mind and is, similar to the Euclidean distances algorithm, very simple to transform into a parallel version. It is a single $\mathcal{O}(n)$ operation looping over $\mathcal{O}(n)$ threads as shown in Algorithm 4 and pointed out by Harish and Narayanan [HN07] who reported significant speed improvements. Again, we use the texture memory access method to profit from caching effects.

Degree Based Distance Manipulation An alternative and fast method to visualise network structures is to pre-compute distances out of adjacencies. The main idea is to declare

Algorithm 4 Parallel Floyd-Warshall(G=(V,E))

```
create adjacency matrix A from G
for k ← 1...n do
    for all elements in the adjacency matrix A, where 1 ≤ i, j ≤ n in parallel do
        A[i, j] ← min(A[i, j], A[i, k] + A[k, i])
    end for
end for
```

nodes with many neighbours as large nodes. Basically, this method is in a time complexity of $\mathcal{O}(no)$ with o is the average number of neighbours per node. Since most networks in biology are sparse, the algorithm works very fast. In substance, in such a network $G = (V, E)$ the distances are defined as $l(e) = \deg(u) + \deg(v)$ for $e = (u, v) \in E$.

HiT-MDS is applied two times on the pre-computed distance values using half of the standard cycle number each. In the first run, we set unknown distances belonging to different components to the graph's doubled diameter. This first step separates all components. In the second step, and hence, during the second half of total algorithm cycles, points are moved to their best correlation based positions. A result of this approach is given in Figure 2.

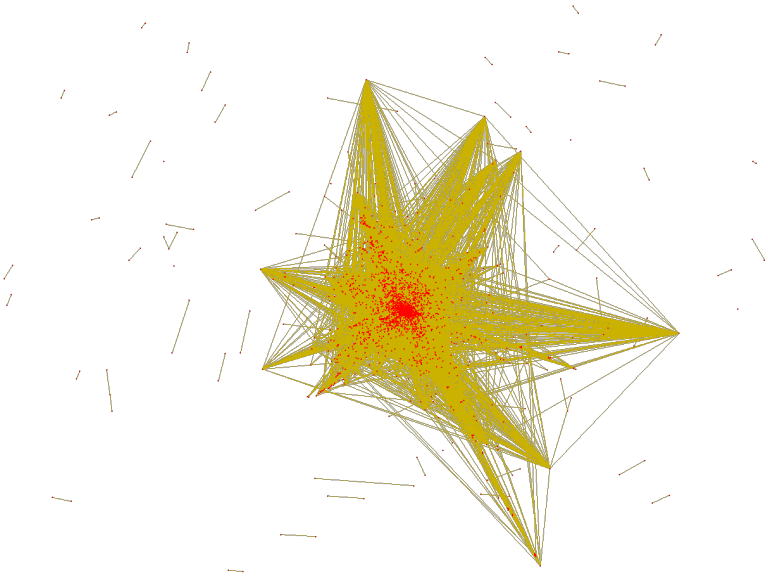


Figure 2: Yeast protein interaction network with 4554 nodes, evaluated with degree based distance interpretation. The node positions are visualised on the basis of CUDA's OpenGL-interoperability.

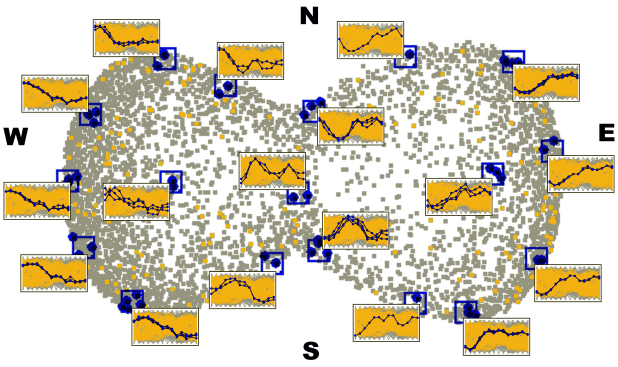


Figure 3: HiT-MDS scatter plot of embedded temporal gene expression data. Correlation similarity $(1-\mathsf{r}(\mathbf{x}^i,\mathbf{x}^j))^p$ is considered at $p=8$ for magnification of high-correlation subsets, which explains the characteristic sand glass shape.

2.3 Application Examples

2.3.1 Global Patterns of Gene Expression

Visualisation is sought for 4824 high-quality genes covering 14 time points of developing Barley
grains [SSUS07]. Their scatter plot is obtained by running HiT-MDS for 50 data cycles, yielding the high-quality display shown in Figure 3. In contrast to previous results the processing time dropped from 861 seconds by a sequential C program to merely 6 seconds using CUDA (not including disk read).

The characteristic sand glass shape results from using eighth power of the correlation measure, more precisely, $(1-\mathsf{r}(\mathbf{x}^i,\mathbf{x}^j))^8$, applied to highly correlated 14-dimensional time series profiles of up-regulation vs. down-regulation processes. The power of eight magnifies subtle dissimilarities in highly correlated genes, this way enhancing their visual differentiation. By posterior labelling with known gene annotations, the exemplary group of hormone and signaling related genes are highlighted in orange colours, other functional categories are marked in gray. Additionally, data boxes, brushed in blue, and their corresponding plots of temporal patterns have been manually picked in order to demonstrate the high spatial connectivity of similar regulatory profiles and their embedded two-dimensional counterparts. A smooth transition can be found from the western side (W) with patterns of down-regulation, via south (S) corresponding to patterns of intermediate up-regulation and up-regulation located in the east (E) to north (N) with intermediate down-regulation, back to west. Since the underlying array was designed for capturing gene expression connected to developmental processes, the majority of genes is in fact expected to be either up- or down-regulated, as visually confirmed by the two major structures. Rare and unique regulation patterns are found in the interior of the sand glass shape.

The prominent temporal expression patterns are easily revealed by browsing the scatter

plot in the way described above. The plot shows that the correlation space is very homogeneous, dominated by patterns of up- and down-regulation, according to the experimental design. Overall, the HiT-MDS embedding procedure applied to transcriptome data of barley tissue development yields a faithful arrangement of genes with their typical temporal expressions. Together with functional annotation data this is a very instrumental tool for screening sets of co-regulation and for an initial derivation of tentative pathways.

2.3.2 Network Layout

Many processes and interactions in biology are represented as networks. Furthermore, there are two common ways to interpret experimental data resulting in networks: i) as a biological network and ii) in the context of an underlying network. Due to the increasing amount of experimental data and the steadily growing size of networks, automatic network layout is important to better understand the relationships and interactions between biological objects such as genes, transcripts, proteins and metabolites. One widely used method for network layout is the force-directed layout method [FR91].

Let $G = (V, E)$ be a network consisting of a set of nodes $V = \{v_1, \ldots, v_n\}$ representing the biological objects (e. g. proteins) and a set of edges $E = \{(v_i, v_j) | v_i, v_j \in V\}$ representing the interactions between the biological objects (e. g. interactions between proteins). A layout of the network is represented by coordinates for the nodes and curves for the edges. A force-directed layout method uses a physical analogy to draw networks. It simulates a system of physical forces defined on the network and produces a drawing which represents a locally minimal energy configuration of that physical system. Force-directed layout methods consist of two parts: i) a system of forces defined by the nodes and edges, and ii) a method to find positions for the nodes (representing the layout of the network) such that for each node the total force is (close to) zero.

A typical method interprets nodes as mutually repulsive 'particles' and edges as 'springs' connecting these particles. This results in attractive forces f_a between adjacent nodes and repulsive forces f_r between non-adjacent nodes. For the current layout for each node $v \in V$ the force $F(v) = \sum_{(u,v)\in E} f_a(u, v) + \sum_{(u,v)\in V\times V} f_r(u, v)$ is computed, which is the sum of all attractive forces f_a and all repulsive forces f_r affecting node v. For example, for the x component the forces f_a and f_r are defined as $f_a(u, v) = c_1 \cdot (d(u, v) - l) \cdot \frac{x(v)-x(u)}{d(u,v)}$ and $f_r(u, v) = \frac{c_2}{d(u,v)^2} \cdot \frac{x(v)-x(u)}{d(u,v)}$, respectively, where l is the optimal distance between any pair of adjacent nodes, $d(u, v)$ is the current distance between the nodes u and v, $x(v)$ is the x-coordinate of node v, and c_1, c_2 are positive constants. Iterative numerical analysis is used to find a locally minimal energy configuration by moving each node in the direction of $F(v)$ to produce a new layout. Finally, the nodes are connected by straight lines.

There are several often used varieties of force-directed methods [Ead84, KK89, SM95]. The computation of the layout is computationally demanding, and fast force-directed methods have been proposed such as an incremental multidimensional scaling heuristic [Bas99] or Walshaw's algorithm (a multi level version of the original algorithm [FR91]) in [HJP02]. The previously shown network example in Figure 2 is based on the HiT-MDS node layout for visualizing a protein interaction network in yeast containing 4554 nodes.

instant size	**MATLAB** [s]	**CUDA** [s]	**speedup**
64	0.114±0.010	0.012±0.004	9.5 x
128	0.126±0.006	0.014±0.007	9.0 x
256	0.616±0.012	0.021±0.003	29.3 x
512	2.777±0.205	0.048±0.004	57.9 x
1024	9.975±0.485	0.178±0.004	56.0 x
2048	43.473±2.832	0.721±0.003	60.3 x
4096	183.700±11.915	3.361±0.024	54.7 x
8192	750.129±48.643	13.997±0.015	53.6 x

Table 1: Performance comparison of optimized MATLAB and CUDA code on test instances of different sizes. The target dimension is three. Measurements refer to time in seconds, excluding data import time.

3 Results

The core HiT-MDS algorithm has been implemented on three different platforms. Two vectorized code samples are available for R and MATLAB/GNU Octave as well as the CUDA version. Code profiling tools of MATLAB and CUDA were used to optimize the performance. Code for R and GNU Octave was manually optimized and is by a factor of 2 to 4 slower than the MATLAB version, because only MATLAB is able to make use of fast single precision arithmetics, thereby using multi-threaded linear algebra routines and loop optimization. Therefore, the fastest code of MATLAB is compared with the CUDA implementation.

Random distance matrices of different sizes were generated for performance tests on the reference server machine, a 16 core server equipped with 3 GHz AMD Opteron CPUs and a NVidia TESLA S870 GPU rack. MATLAB 7.7.0 with multi-thread mode and CUDA 2.1 were used for performance comparisons. Average run times of 10 independent starts with 50 cycles per run were measured and compiled in Table 1. The instant size column refers to matrices representing between 64x64 to 8192x8192 distances. For the fixed number of cycles, the embedding speed is independent of the matrix entries, no matter if full or sparse matrices are processed. This also indicates a general validity of the recorded speed, no matter if for scatter plot generation or for network layout.

Significantly faster execution times of CUDA are found. Yet, small instances yield less speedup than instances of sizes around 2048x2048 for which robust factors over 50 fold acceleration can be stated. Moreover, very small standard deviations are obtained for CUDA, indicating undisturbed use of the GPU hardware for high-performance scientific calculations.

4 Discussion

HiT-MDS is a versatile algorithm with good parallelization potential for reconstruction of dissimilarity relationships in a Euclidean space. It can be used as faithful dimension reduction method, for converting data with a specific data metric into a Euclidean representation, and, by a straight-forward extension, for network reconstruction of adjacency matrices. The method is thus perfectly suited for dealing with data screening, complexity reduction, and relationship characterization, tasks that regularly exist in biological sciences.

At first glance the presented performance comparison between MATLAB and CUDA might seem to be unfair. Yet, only few lines of MATLAB code need to be interpreted per algorithmic cycle. Virtually all matrix operations are handled by internal MATLAB functions of optimized algebra subroutines that can be hardly beaten by hand-written C++ code. Just another theoretical factor of 2 would be gained for MATLAB if symmetry of the matrices could be efficiently exploited. Yet another clear time benefit of CUDA remains: thanks to the GPU server architecture heavy computations can run almost independent of the host, if memory transfer between CPU and GPU remains at a low level.

The main ingredients to successful utilization of CUDA turned out to be (i) the consistent use of the reduction principle for using fast shared memory on the multiprocessors instead of slow global memory on the graphics board, (ii) the use of texture memory, and (iii) a good arrangement of threads into logical blocks, and (vi) the use of thread-interleaving memory access (coalescence). For the computing task at hand, single precision floating point numbers of 32 bit worked as reliably as double precision. The technical limitation of the size of the distance matrix is currently at about 14000x14000 elements on a graphics board with 1.5 GB memory. Yet, larger memory capacity, double precision calculations, and more multiprocessors per GPU are already available at low prices.

Future tasks are related to deal with larger network structures, which requires an implementation of a sparse matrix data structure. Another challenge will be the identification of network nodes in very large graph structures.

References

[Bas99] W. Basalaj. Incremental Multidimensional Scaling Method for Database Visualization. In R. F. Erbacher, P. C. Chen, and C. M. Wittenbrink, editors, *Visual Data Exploration and Analysis VI (Proc. SPIE)*, volume 3643 of *Proceedings of SPIE*, pages 149–158, 1999.

[BK09] J. Breitbart and G. Khanna. An exploration of CUDA and CBEA for a gravitational wave data-analysis application. Einstein@Home, 2009.

[CMS08] S. Che, J. Meng, and J. W. Sheaffer. A performance study of general purpose applications on graphics processors using CUDA. *Journal of Parallel and Distributed Computing*, 68(10):1370–1380, 2008.

[Cor08] NVIDIA Corporation. *NVIDIA CUDA Programming Guide Version 2.1*, 12/8/2008.

[Ead84] P. Eades. A Heuristic for Graph Drawing. *Congressus Numerantium*, 42:149–160, 1984.

[FJ07] M. Fatica and W. Jeong. *Accelerating MATLAB with CUDA*. HPEC, 2007.

[FR91] T. Fruchterman and E. Reingold. Graph Drawing by Force-directed Placement. *Software - Practice and Experience*, 21(11):1129–1164, 1991.

[GDD08] N. A. Gumerov, R. Duraiswami, and W. Dorland. *Efficient Personal Supercomputing in Fortran 9x on CPU-GPU Systems*. CSCAMM, 2008.

[GHGC09] A. Godiyal, J. Hoberock, M. Garland, and J. C.Hart. Rapid Multipole Graph Drawing on the GPU. In *Graph Drawing*, volume 5417 of *LNCS*, pages 90–101. Springer, 2009.

[Hit] HiT-MDS at IPK Gatersleben - Data Inspection Group. http://dig.ipk-gatersleben.de/.

[HJP02] K. Han, B.-H. Ju, and J. H. Park. InterViewer: Dynamic Visualization of Protein-Protein Interactions. In M. T. Goodrich and S. G. Kobourov, editors, *Graph Drawing (Proc. GD '02)*, volume 2528 of *LNCS*, pages 364–365. Springer, 2002.

[HN07] P. Harish and P.J. Narayanan. Accelerating Large Graph Algorithms on the GPU Using CUDA. In *High Performance Computing*, volume 4873 of *LNCS*, pages 197–208. Springer, 2007.

[HSO07] M. Harris, S. Sengupta, and J. D. Owens. Parallel Prefix Sum (Scan) with CUDA. In *GPU Gems 3*, pages 851–875. NVIDIA Corporation, 2007.

[IMO09] S. Ingram, T. Munzner, and M. Olano. Glimmer: Multilevel MDS on the GPU. *IEEE Transactions on Visualization and Computer Graphics*, 15(2):249–261, 2009.

[JK09] M. Januszewski and M. Kostur. Accelerating numerical solution of Stochastic Differential Equations with CUDA. *ArXiv e-prints*, 2009.

[KK89] T. Kamada and S. Kawai. An Algorithm for Drawing General Undirected Graphs. *Information Processing Letters*, 31(1):7–15, 1989.

[LKPM09] H. Li, A. Kolpas, L. Petzold, and J. Moehlis. Parallel Simulation for a Fish Schooling Model on a General-Purpose Graphics Processing Unit. In *Concurrency and Computation: Practice and Experience*, 21(6), pages 725–737, 2009.

[PD04] M. Pătraşcu and Demaine. Lower Bounds for Dynamic Connectivity. In *Encyclopedia of Algorithms*, pages 473–477. Springer, 2004.

[RRS+07] S. Ryoo, C. I. Rodrigues, S. S. Stone, S. S. Baghsorkhi, S. Ueng, and W. W. Hwu. Program optimization study on a 128-core GPU. Presented at the First Workshop on General Purpose Processing on Graphics Processing Units, October 2007.

[Sel08] J. Seland. CUDA Programming, 2008. http://heim.ifi.uio.no/ knutm/geilo2008/seland.pdf.

[SM95] K. Sugiyama and K. Misue. A Simple and Unified Method for Drawing Graphs: Magnetic-Spring Algorithm. In R. Tamassia and I. G. Tollis, editors, *Graph Drawing (Proc. GD'94)*, volume 894 of *LNCS*, pages 364–375. Springer, 1995.

[SSUS07] M. Strickert, N. Sreenivasulu, B. Usadel, and U. Seiffert. Correlation-maximizing surrogate gene space for visual mining of gene expression patterns in developing barley endosperm tissue. *BMC Bioinformatics*, 8:165, 2007.

[TO05] Y.-H. Taguchi and Y. Oono. Relational patterns of gene expression via non-metric multidimensional scaling analysis. *Bioinformatics*, 21(6):730–740, 2005.

Identification of cancer and cell-cycle genes with protein interactions and literature mining

Loic Royer, Conrad Plake and Michael Schroeder
{loic.royer,conrad.plake,michael.schroeder}@biotec.tu-dresden.de
Biotec, TU Dresden, Dresden, Germany

Abstract: Gene prioritization based on background knowledge mined from literature has become an important method for the analysis of results from high-throughput experimental assays such as gene expression microarrays, RNAi screens and genome-wide association studies. We apply our gene mention identifier, which achieved the best result of over 80% in the BioCreative II text-mining challenge [HPR+08], and show how text-mined associations can be complemented using guilt-by-association on high confidence protein interaction networks.

First, we predict hand-curated gene-disease relationships in the OMIM database, Entrez Gene summaries and GeneRIFs with 37% success rate. Second, we confirm 24% of novel cell-cycle genes identified in a recent RNAi screen [KPH+07] by using text-mining and high confidence protein interactions. Moreover, we show how 71% of GOA cell-cycle annotations can be automatically recovered. Third, we devise a method to rank genes based on novelty, increasing interest, impact, and popularity.

1 Introduction

With high-throughput methods such as gene expression analyses, high-throughput RNA interference, and genome-wide association studies, gene prioritization becomes an important problem. Gene prioritization orders lists of genes according to their likelihood to be associated to a process, phenotype, or disease. In particular, genetic linkage analyses identify chromosomal regions, which are linked to a disease and which can contain hundred candidate genes linked to a disease. The problem becomes one of establishing indirect links from the candidate genes to the disease. These links can be of very different nature such as protein interactions [GLF+06, LKS+07], similarity of annotations from controlled vocabularies (phenotype in MeSH [LKS+07]), GeneOntology [AAE+05, PIBA02, TCS03], anatomy [TKP+05], sequence similarity [TCS03, GLF+06, AAE+05, LBO04, PIBA02], phylogeny [LBO04], or co-expression of genes [TKP+05, TCS03, vDCK+03, vDCK+05]. As a result, these approaches manage to significantly reduce the number of candidate genes [TAT+06] or even directly identify the disease gene such as [LKS+07], who predict for 298 out of 669 linkage intervals the correct disease gene.

While the above studies prioritize some hundred genes regarding their link to a disease, other efforts aim to establish large scale links between all genes of a genome and disease. [BK06] mine meta-information of all data sets in the Gene Expression Omnibus by ex-

tracting UMLS concepts from descriptions. This way they can identify novel genes linked to aging. Similarly, [GCV+07] link human protein interaction with expression and disease data. They conclude that disease genes are less likely to be essential interaction hubs and are functionally on the periphery.

In general, protein interaction data can be beneficial to infer indirect relationships by applying the principle of guilt-by-association. Both [GLF+06] and [LKS+07] make use of interactions in their analysis of linkage intervals and [GUT+08] find that half of their correct gene-disease associations are indirectly inferred via protein interactions.

Summarizing the above work, there are three interesting aspects:

- First, only few approaches (e.g. [TKP+05, LKS+07]), apply text-mining to associate genes and diseases and none of them apply large-scale identification of genes and diseases in the whole of the medical literature.
- Second, only few (e.g. [GLF+06, LKS+07, GUT+08]) use protein interactions and the principle of guilt-by-association.
- Third, experimental validation of prediction of novel disease genes is scarce, since such links are inherently difficult to verify due to the complexity of diseases.

In this paper, we address these three points. We show how state-of-the-art entity recognition of cancer genes, cell-cycle terminology, diseases, and their co-occurrences combined with the principle of guilt-by-association can predict hot cancer genes and novel cell-cycle genes. Hot cancer genes i) are novel, ii) have been published in high impact journals, iii) their popularity has not yet peaked, iv) and they attract a large group of researchers. This meta information, which is the result of the comprehensive mining of literature, lends itself to identify genes worthy of exploitation since there is a direct or indirect link and they are truly novel candidates. As argued above, the validation of gene-disease predictions is difficult, since there are no straight forward experiments. We address this problem by validating our approach on novel cell-cycle genes, which have been identified in a genome-wide RNAi screen [KPH+07]. The screen provides a gold standard of over 850 novel cell cycle genes, which have not been discussed in literature before.

The cornerstone of our approach is our entity recognition algorithm, which achieved the best results (81% success rate) in the recent BioCreative text-mining task of human gene name identification [HPR+08]. Since then we have further improved it to 86% success rate. We applied the algorithm, which is online accessible via the BioCreative meta server [LKRP+08], to over 17,000,000 abstracts from PubMed. Additionally, we considered for each abstract any annotated disease terms from the Medical Subject Heading, MeSH, and all stemmed cell-cycle terminology from the Gene Ontology literally appearing in the abstracts. Overall, our method identifies 2.74 million abstracts mentioning a gene, 1.71 million abstracts mentioning a gene and a disease, and 210.000 a gene and cell-cycle term. This resource is now available as GoGene [1] [PRW+09].

With this large data source, we define a simple co-occurrence model and set out to solve the following problems: First, how well can our model predict hand-curated gene-disease

[1] http://gopubmed.org/gogene

relationships in the OMIM database, and in Entrez Gene GeneRIFs. Second, how many of the over 850 novel cell-cycle genes identified in the RNAi screen [KPH+07] can be predicted with our method and which role does text-mining play and which inference through protein interactions? Third, we devise a method to rank genes based on novelty, increasing interest, impact, and popularity. With this ranking, we discuss 50 hot cancer genes and 20 hot cell-cycle genes in detail.

2 Methods

Identification of human genes in PubMed. We identify human gene mentions in PubMed by parsing each abstract with a dictionary of gene names, synonyms, and spelling variants. First we find as many hits as possible. In a second step, a context sensitive filter is applied to remove false positive matches by looking at tokens in the neighborhood of each name/hit and by resolving abbreviations to long forms. Finally, polysemous names, i.e. names referring to more than one gene, are disambiguated by comparing the text at hand against each candidate profile. A profile contains all known information of a gene, e.g. GO annotations, diseases, background texts etc. taken from the high quality databases Entrez Gene and SwissProt. The profile that best fits the text is taken as sense for the ambiguous gene name. For a detailed explanation of our gene identification method see [HPR+08].

Gene ranking. For researchers trying to obtain insights from large screening data, not all genes are equally important. Some genes have often been discussed in the literature. For some genes, the research interest of the community has reached saturation and has since then declined. These declining genes have well-known and stable functions. It is less probable that new insights into their function can be discovered . Yet, they do provide a rich source of information that can shed light on the experiment. In contrast, other genes lie at the forefront of research and have just recently received names and are only found in recent publications. The probability to discover new insights for these genes is higher due to their novelty. Moreover, a novel gene with many recent mentions in high impact journals constitutes an even better candidate. By compiling the publication dates of all human gene mentions in MEDLINE we can decide for each human gene whether research interest has peaked and is dwindling, if the gene belongs to some hot topic of research, or if the gene is discussed in a large body of high-impact literature.

Bibliometric features. We chose four features to measure how 'interesting' a gene is. First the category peaked/not-peaked: $peaked_g$, second the number of publications weighted by impact factor: $volume_g$, third how recent is the interest in the gene independently of the total number of papers: $novelty_g$, and finally the total number of distinct authors that contributed to the publications for that gene: $community_g$. We defined a gene has having peaked if the highest count of papers is at least 3 years old and if since then there was a consistent decrease in the number of papers. We compute novelty using a

simple exponential decrease of the relevance of old mentions divided by the total sum of all impact factor points: $novelty_g = \sum_{y=1950}^{2007} \alpha^{y-2007} c_{g,y}$

In this formula $c_{g,y}$ represents the cumulative impact factor for gene g for the year y. We chose to use a yearly decrease of 50% toward the past ($\alpha = 0.5$). The $community_g$ measure has a minor impact on the ranking as it is strongly correlated to the $volume_g$, but it does contribute additional information about the size of the research community for a gene. To combine these measures we use a Pareto ranking approach[MA94]. The advantage of this scheme is that it ranks genes according to all four features in a balanced manner. For example, shown in Table 4 are the top five genes ranked for breast cancer. Among these is SIRT7, a novel gene with many high-impact publications in recent years, but also BRCA1, a well known and important gene for breast cancer. Among the 36509 human genes, 16078 are found mentioned in MEDLINE and among these 31% have peaked (as of January 2008).

Association of Genes to MeSH and GO terms. To annotate genes with terms from MeSH and GO, we basically count the number of co-occurrences in the literature. For each gene-term pair we compute an association-score as follows: $\text{score}_{g,t} = log_2 \frac{N \times n_{g,t}}{n_g \times n_t}$

where N is the number of articles mentioning any gene and any term from the branch (e.g. a disease), $n_{g,t}$ is the number of articles mentioning the gene and the term, n_g is the number of articles mentioning the gene and any term from the branch, and n_t is the number of articles mentioning the term and any gene. The higher the association score the more likely this pair will be mentioned together in the literature. An association score of zero means that gene and term occur independently of one another. A negative score signals an underrepresentation of a pair in the literature.

Guilt-by-association. Guilt-by-association is the principle by which qualities can be transfered between associated items. In our case, we transfered the information *is cell-cycle related* to all direct interaction partners of a gene if it co-occurs significantly with cell-cycle terms in the literature. We experimented with several decision functions but observed that simply transferring to all direct neighbors of a gene performed best. In particular, adding more distant neighbors increases recall but leads to a significant decrease in precision (data not shown).

3 Results

Prediction of Gene-Disease Associations. We compare text mined gene-disease associations to Entrez Gene Summaries and GeneRIFs texts[2] and to the OMIM database (Tables 3 and 3). The achieved precision rates in Table 3 are underestimations because of the many incomplete GeneRIFs and Summaries (GRS) [BCF+07]. While the precision is low because of many incomplete GRS, the recall suffers from false positives in the benchmark

[2] Downloaded December 2007

Min. Score	Sig. level (%)	Precision (%)	Recall (%)	F1 (%)
12	0.04	**59.2**	1.7	3.3
10	0.22	44.0	6.1	10.6
8	0.97	21.8	11.8	**15.4**
6	3.5	9.3	19.6	12.6
4	10.8	4.7	33.0	8.2
2	28.6	3.0	53.7	5.6
0.6	50.0	2.5	**73.5**	4.8

Table 1: Results for text mined gene-disease associations. Comparison of gene-disease associations for different score thresholds. Predicted diseases are compared to automatically annotated Entrez Gene Summaries and GeneRIFs texts.

Min. Score	Sig. level (%)	Precision (%)	Recall (%)	F1 (%)
13	0.01	**88.5**	6.4	12.0
12	0.04	76.3	10.2	18.1
10	0.22	50.0	29.0	**36.7**
8	0.97	22.4	44.5	29.8
6	3.5	8.1	59.1	14.3
4	10.8	3.1	73.8	6.0
2	28.6	1.4	88.3	2.7
0.6	50.0	0.08	**93.8**	1.6

Table 2: Results for text mined gene-disease associations in OMIM. Comparison of gene-disease associations for different score thresholds. Predicted associations are compared to the OMIM gene-disease catalog.

data set. Annotation of GRS was done automatically by simple, context insensitive matching to have as many training examples as possible of all kinds of diseases at the expense of including false positives due to ambiguous disease terms. Since GRS are manually added to a gene's record, they might also contain information, which is not present in the abstracts of publications. Thus, only by looking at abstracts and not at full texts we are likely to miss gene-disease associations. Our predictions of human genes associated with genetic disorders are evaluated by comparing to the OMIM gene-disease catalog Table 3. We successfully mapped 358 disease concepts in MeSH to their respective counterparts in OMIM and used them as a benchmark data set. As expected, the achieved results in terms of recall are much better than for our GRS benchmark, since all annotations in OMIM can be regarded as more reliable as GRS entries.

Prediction of Cell-Cycle Genes. A recent genome-wide high-throughput RNAi screen identified 1351 genes important for cell division in HeLa cells [KPH$^+$07]. Among these 1351 genes, only 243 were previously associated with cell-cycle progression, and 252 previously uncharacterized genes were assigned this function. The remaining 882 genes were known to be implicated in other functions than cell-cycle. Another study characterized the genome-wide program of gene expression during the cell division cycle in HeLa cells us-

ing cDNA micro-arrays [WSS+02]. They identified genes periodically expressed during cell-cycle progression. These genes are not necessarily important to cell-cycle itself, but are nevertheless downstream of the cell-cycle machinery. As shown in Fig. 1D only 89 genes are both important for cell-cycle and periodically expressed, and 53 among these were previously known to be important for cell-cycle.

We compare these experimental knowledge to gene–cell-cycle associations mined from MEDLINE abstracts Table 3. To improve recall, we used protein sequence homology as well as protein interactions to transfer associations using the principle of guilt-by-association. First we evaluated how many genes associated to cell-cycle in GOA and KEGG can be recovered using text-mining. We achieve a recall of 40.6% and a precision of 43.3% using co-occurrence of genes with cell-cycle GO terms. Using the principle of guilt-by-association, the recall can be improved to 45.3% when using protein sequence homology, and to 67.9% when using protein interactions from the HPRD database [MSK+06]. Thus, high-quality protein interactions are a valuable resource for improving recall. Moreover, predicting all genes known to be related to cell-cycle either in GOA, KEGG, [KPH+07] or [WSS+02] leads to a decrease in recall of 19.3% except when using protein interactions. In this case, the recall more than doubles to 44.3% with the best F1-measure of 34.5%.

Next, we tried to predict the 882 new cell-cycle related genes identified by [KPH+07] using text-mining, protein sequence homology and protein interactions. Using solely gene–cell-cycle co-occurrences, our approach achieved a maximum p-value of 0.51 indicating the difficulty of predicting previously unknown cell-cycle genes using text-mining alone. Adding protein sequence homology leads to a slightly better p-value of 0.24 – still insignificant. However, using protein interactions improves the statistical significance of the results with a p-value of 0.016 (HPRD). In this case we are able to confirm 24.3% of the new cell-cycle genes identified by [KPH+07].

4 Discussion

The method we proposed can find novel links between genes and diseases not yet contained in databases such as Entrez Gene and OMIM. Especially genes that link to neoplasms are of high importance because of the high mortality of cancer patients. Among those genes, we can highlight interesting ones using a ranking that integrates different measures for interest such as *novelty*, *volume*, *community*, and *peaked*. We picked 10 different cancers categories (8 by site and 2 by type) and ranked the associated genes for each category (Tab. 4). Interestingly, none of our top ranked genes for pancreatic cancers is listed in OMIM. A manual inspection of those genes showed that each is indeed linked to pancreatic cancers. A possible explanation is that OMIM only includes genes shown to follow Mendelian inheritance patterns. For example, the top-ranked gene SIRT4 was first reported in 1999 in a publication about the characterization of five human yeast SIR2 homologs. Next publications followed in 2002, 2003, and 2005 revealing the regulation of SIRTs by histone deacetylase inhibitors. Then in 2006, two papers in the high-impact journal Cell were published reporting that SIRTs turn out to be critical regulators of metabolism

Prediction	Rec.(%)	Prec.(%)	F1(%)	p-value(<)
gene-cell cycle GO terms co-occurrence:				
GOA	55.7	32.2	40.4	10^{-186}
GOA+KEGG	40.6	43.3	41.4	10^{-204}
GOA+KEGG+Kittler	21.3	47.3	29.0	10^{-123}
GOA+KEGG+Kittler+Whitfield	19.3	52.2	27.8	10^{-125}
gene-cell cycle GO terms co-occurrence + sequence homology:				
GOA	61	28.0	38.3	10^{-191}
GOA+KEGG	45.3	38.4	41.2	10^{-209}
GOA+KEGG+Kittler	24.4	43.0	30.8	10^{-125}
GOA+KEGG+Kittler+Whitfield	19.3	52.2	27.8	10^{-125}
gene-cell cycle GO terms co-occurrence + PPI:				
GOA	**71.7**	13.5	21.9	10^{-136}
GOA+KEGG	**67.9**	17.1	27	10^{-154}
GOA+KEGG+Kittler	**45.3**	23.8	30.4	10^{-87}
GOA+KEGG+Kittler+Whitfield	**44.3**	28.3	**34.2**	10^{-100}
Predicting new cell cycle genes found in Kittler et al. [KPH$^+$07]:				
Cell cycle term co-occurence	3.4	7.0	4.2	0.51
+ protein sequence homology	6.7	7.7	6.4	0.24
+ protein interactions (HPRD)	**24.3**	8.0	12	**0.016**

Table 3: Predicting cell cycle related genes using GO term co-occurrence, protein sequence homology and protein interactions. Predicting known cell cycle related genes from GOA can be done at a maximal recall of 71%. Protein sequence homology only improves recall at the cost of a loss in precision. Using protein interactions improves recall, and when predicting all known cell cycle related genes (GOA+KEGG+Kittler+Whitfield) it achieves a higher F1-measure than pure co-occurrence. Predicting the new cell-cycle genes of [KPH$^+$07] does not work using pure text-mining ($P = 0.51$), is only marginally improved using protein sequence homology ($P = 0.24$), but becomes significant when using protein interactions from HPRD ($P = 0.016$).

Bone	Brain	Breast	Eye	Leukemia	Liver	Lymphoma	Pancreas	Prostate	Skin
FXYD6	CRB3	SIRT7	E2F5	CLLU1	LIN28B *	NPC1L1	SIRT4	OR51E1 *	KRT1
ADAM8	GHRHR	BRIP1 *	E2F1 *	ARL11 *	HNF1A *	ULBP2 *	G6PC2	OR51E2 *	MSH2 *
C9orf46	SCGN	BRCA1 *	CDK4 *	FCRL3	TCP10L *	TBX21	SOX2	TMPRSS2 *	MSH6 *
FGF23 *	HDAC-3	SERPINB5 *	KIF14 *	CCDC28A	UGT2B7	FHL2	FFAR1	PCA3 *	MLH1 *
TRIB2	SOX4	AP1S2	RBBP8 *	GATA1 *	ZNF689	CCR7	CDX2	PI16	KRT15

Table 4: Top five genes for 10 cancer. The top five genes according to the Pareto ranking for 10 different neoplasms. Most of the listed genes have seen an increase in research interest in recent years and have a high volume of high impact publications. Genes with a star are mentioned in OMIM to be related to the corresponding disease, genes without a star are not. Note that for brain and pancreas cancer none the top 5 genes identified are listed in OMIM.

and that SIRT4 acts in the mitochondria of pancreatic cells. The loss of SIRT4 in insulinoma cells up-regulates amino-acid-stimulated insulin secretion, which links SIRT4 to pancreatic cancers [HMH$^+$06]. Another member of the histone deacetylase gene family is SIRT7, which we found associated to breast cancers. Together with SIRT4, this gene was

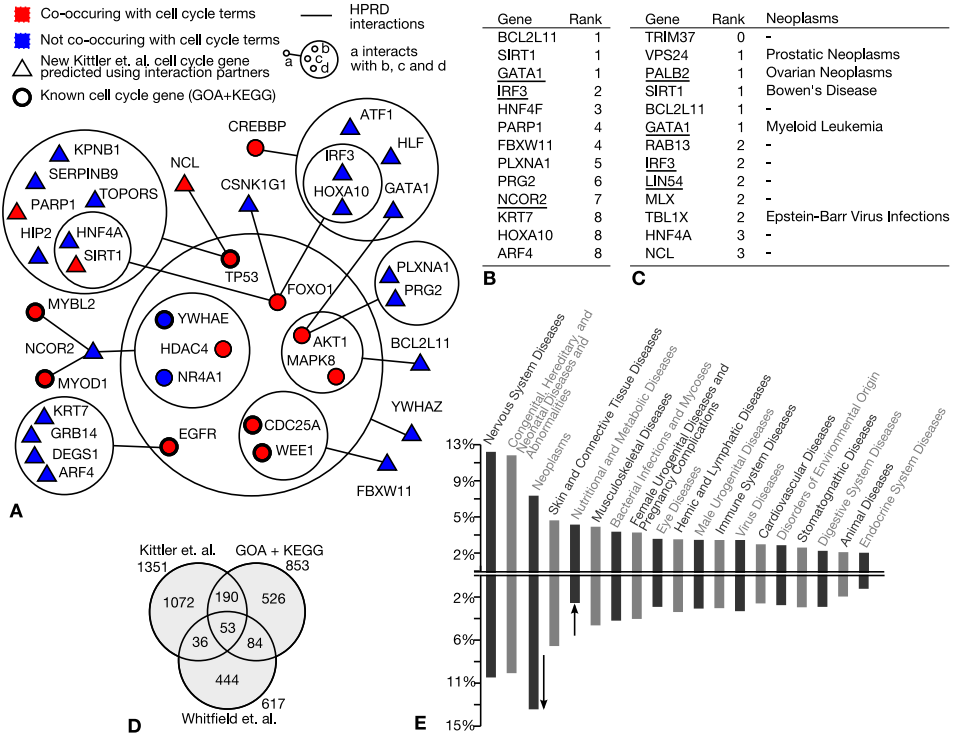
first reported in 1999. A recent publication in the British Journal of Cancer reports that levels of SIRT7 expression were significantly increased in breast cancers.

As expected, we found more links to cancers among genes known to be involved in cell-cycle progression, since defects in the cell-cycle are causative for cancers development (Fig. 1E). A recent RNAi screen identified more than 850 new genes with impact on cell-cycle progression [KPH$^+$07]. Out of those, 24% can be further confirmed by literature mining combined with high confidence protein interaction networks. A ranking of these genes highlights the interesting candidates for further research and confirmation studies (Fig. 1B and C). Figure 1A shows a sub-graph of the HPRD network with genes predicted by our method. For example, let's examine the gene IRF3 – an interferon regulatory factor. It forms a complex with CREBBP and thus interacts in the network with CREBBP [YLM$^+$02]. Moreover, CREBBP co-occurs with cell-cycle terms such as 'DNA replication checkpoint', 'centriole replication', 're-entry into mitotic cell-cycle'. These co-occurrences together with the interaction between IRF3 and CREBBP is our evidence for a link between IRF3 and cell-cycle. The importance of IRF3 for the cell-cycle can be further confirmed in that the target genes of IRF-3 are themselves involved in cell-cycle as shown in a recent Nature publication [AVC$^+$07].

PALB2 is a breast cancer susceptibility gene that interacts with BRCA2 to enable its recombinational repair and checkpoint functions [XSN$^+$06]. When mutated, it more than doubles the risk of breast and ovarian cancers [WK07]. PALB2 is among the genes identified by [KPH$^+$07] and predicted using both gene-cell-cycle co-occurrences and protein interactions (Fig. 1C). In HPRD, PALB2 is reported to interact with BRCA2, a gene which we find co-occurring significantly with cell-cycle terms in MEDLINE. Yet, PALB2 itself is not associated to cell-cycle in GOA or KEGG nor does it significantly co-occur with cell-cycle terms in the literature. This example shows how protein interactions can help to recover such candidates. As seen in Fig. 1C, our method correctly associates PALB2 as top ranked for ovarian cancers. PALB2 is definitely a 'hot' gene first researched in 2006 and discussed in 2007 in four Nature publications.

5 Conclusion

We showed the feasibility of a simple statistical co-occurrence model to find links between genes and diseases as well as between genes and cell-cycle processes by automatically searching the literature and using high confidence protein interactions. The achieved results for finding those associations are comparable to recent approaches to relationship extraction from texts, such as protein-protein interactions [KLV07]. The main contributions of our work are: i) the application of a state-of-the-art gene name identifier to all articles indexed in MEDLINE, ii) the ranking of all genes discussed in the literature by different measures of interest, iii) the potential to find novel links between genes, cancers, and cell-cycle processes not yet annotated in public databases, and iv) the support of high-throughput experiments by filtering results using knowledge from literature and known interaction networks to select the most promising gene candidates.



Gene	Rank
BCL2L11	1
SIRT1	1
GATA1	1
IRF3	2
HNF4F	3
PARP1	4
FBXW11	4
PLXNA1	5
PRG2	6
NCOR2	7
KRT7	8
HOXA10	8
ARF4	8

B

Gene	Rank	Neoplasms
TRIM37	0	-
VPS24	1	Prostatic Neoplasms
PALB2	1	Ovarian Neoplasms
SIRT1	1	Bowen's Disease
BCL2L11	1	-
GATA1	1	Myeloid Leukemia
RAB13	2	-
IRF3	2	-
LIN54	2	-
MLX	2	-
TBL1X	2	Epstein-Barr Virus Infections
HNF4A	3	-
NCL	3	-

C

Figure 1: Predicting Cell Cycle genes. (A) Example HPRD Sub-network of protein interactions for new cell cycle genes [KPH$^+$07] predicted using our method, and visualized using power graphs [RRAS08]. Genes like IRF3 and NCOR2 can be predicted using both gene–cell cycle term co-occurrence and high quality protein interactions from HPRD. (B) Top 12 hottest genes among genes shown in the example sub-network. Genes BCL2L11, SIRT1, GATA1 and IRF3 are at top. (C) Top 12 hottest genes among all new cell cycle genes from [KPH$^+$07] predicted by our method together with significantly co-occurring neoplasms. (D) Overlap between [KPH$^+$07] cell cycle genes, [WSS$^+$02] cell cycle periodic genes, and known cell cycle genes annotated in GOA or KEGG. Among the 53 genes in all three sets we find genes involved in the structural aspects of cell cycle such as histones, centromere proteins, tubulins and kinesins, that are both important and periodically expressed. Only 36 genes are both found by [KPH$^+$07] and [WSS$^+$02] but are not annotated in GOA or KEGG such as MELK and CBX3. (E) Disease associations mined from literature for all human genes (top) compared to disease associations for cell cycle genes. As expected, cell cycle genes are enriched in neoplasms and depleted in nutritional and metabolic diseases.

References

[AAE+05] Euan A Adie, Richard R Adams, Kathryn L Evans, David J Porteous, and Ben S Pickard. Speeding disease gene discovery by sequence based candidate prioritization. *BMC Bioinformatics*, 6:55, 2005. disease.

[AVC+07] J. Andersen, S. Vanscoy, T-F. Cheng, D. Gomez, and N. C. Reich. IRF-3-dependent and augmented target genes during viral infection. *Genes Immun*, Dec 2007.

[BCF+07] WA Baumgartner, KB Cohen, LM Fox, G Acquaah-Mensah, and L Hunter. Manual curation is not sufficient for annotation of genomic databases. *Bioinformatics*, 23(13):i41–i48, Jul 2007.

[BK06] Atul J Butte and Isaac S Kohane. Creation and implications of a phenome-genome network. *Nat Biotechnol*, 24(1):55–62, Jan 2006. disease.

[GCV+07] Kwang-Il Goh, Michael E Cusick, David Valle, Barton Childs, Marc Vidal, and Albert-László Barabási. The human disease network. *Proc Natl Acad Sci U S A*, 104(21):8685–8690, May 2007. disease.

[GLF+06] Richard A George, Jason Y Liu, Lina L Feng, Robert J Bryson-Richardson, Diane Fatkin, and Merridee A Wouters. Analysis of protein sequence and interaction data for candidate disease gene prediction. *Nucleic Acids Res*, 34(19):e130, 2006. disease.

[GUT+08] Graciela Gonzalez, Juan C Uribe, Luis Tari, Colleen Brophy, and Chitta Baral. Mining gene-disease relationships from biomedical literature: weighting protein-protein interactions and connectivity measures. *Pac Symp Biocomput*, pages 28–39, 2008.

[HMH+06] MC Haigis, R Mostoslavsky, KM Haigis, K Fahie, DC Christodoulou, AJ Murphy, DM Valenzuela, GD Yancopoulos, M Karow, G Blander, C Wolberger, TA Prolla, R Weindruch, FW Alt, and L Guarente. SIRT4 inhibits glutamate dehydrogenase and opposes the effects of calorie restriction in pancreatic beta cells. *Cell*, 126(5):941–54, Sep 2006.

[HPR+08] J. Hakenberg, C. Plake, L. Royer, H. Strobelt, U. Leser, and M. Schroeder. Gene mention normalization and interaction extraction with context models and sentence motifs. *Genome Biology*, 2008. to appear.

[KLV07] Martin Krallinger, Florian Leitner, and Alfonso Valencia. Assessment of the Second BioCreative PPI task: Automatic Extraction of Protein-Protein Interactions. In *Proceeding of the Second BioCreative Challenge Evaluation Workshop*, pages 41–54, 2007.

[KPH+07] Ralf Kittler, Laurence Pelletier, Anne-Kristine Heninger, Mikolaj Slabicki, Mirko Theis, Lukasz Miroslaw, Ina Poser, Steffen Lawo, Hannes Grabner, Karol Kozak, Jan Wagner, Vineeth Surendranath, Constance Richter, Wayne Bowen, Aimee L Jackson, Bianca Habermann, Anthony A Hyman, and Frank Buchholz. Genome-scale RNAi profiling of cell division in human tissue culture cells. *Nat Cell Biol*, 9(12):1401–1412, Dec 2007. cell cycle.

[LBO04] Núria López-Bigas and Christos A Ouzounis. Genome-wide identification of genes likely to be involved in human genetic disease. *Nucleic Acids Res*, 32(10):3108–3114, 2004. disease.

[LKRP+08] F. Leitner, M. Krallinger, C. Rodriguez-Penagos, J. Hakenberg, C. Plake, C. Kuo, C. Hsu, R. Tsai, H. Hung, W. Lau, C. Johnson, R. Sæ tre, K. Yoshida, Y. Chen, S. Kim, S. Shin, B. Zhang, W. Baumgartner, L. Hunter, B. Haddow, M. Matthews, X. Wang, P. Ruch, F. Ehrler, A. Ozgur, G. Erkan, D. Radev, M. Krauthammer, T. Luong, R. Hoffmann, C. Sander, and A. Valencia. Introducing Meta-Services for Biomedical Information Extraction. *Genome Biology*, 2008. accepted.

[LKS+07] Kasper Lage, E. Olof Karlberg, Zenia M Storling, Páll I Olason, Anders G Pedersen, Olga Rigina, Anders M Hinsby, Zeynep Tümer, Flemming Pociot, Niels Tommerup, Yves Moreau, and Soren Brunak. A human phenome-interactome network of protein complexes implicated in genetic disorders. *Nat Biotechnol*, 25(3):309–316, Mar 2007. disease.

[MA94] Osborne M.J. and Rubenstein A. *A Course in Game Theory.* MIT Press, 1994.

[MSK+06] Gopa R Mishra, M. Suresh, K. Kumaran, N. Kannabiran, Shubha Suresh, P. Bala, K. Shivakumar, N. Anuradha, Raghunath Reddy, T. Madhan Raghavan, Shalini Menon, G. Hanumanthu, Malvika Gupta, Sapna Upendran, Shweta Gupta, M. Mahesh, Bincy Jacob, Pinky Mathew, Pritam Chatterjee, K. S. Arun, Salil Sharma, K. N. Chandrika, Nandan Deshpande, Kshitish Palvankar, R. Raghavnath, R. Krishnakanth, Hiren Karathia, B. Rekha, Rashmi Nayak, G. Vishnupriya, H. G Mohan Kumar, M. Nagini, G. S Sameer Kumar, Rojan Jose, P. Deepthi, S. Sujatha Mohan, T. K B Gandhi, H. C. Harsha, Krishna S Deshpande, Malabika Sarker, T. S Keshava Prasad, and Akhilesh Pandey. Human protein reference database–2006 update. *Nucleic Acids Res*, 34(Database issue):D411–D414, Jan 2006.

[PIBA02] Carolina Perez-Iratxeta, Peer Bork, and Miguel A Andrade. Association of genes to genetically inherited diseases using data mining. *Nat Genet*, 31(3):316–319, Jul 2002. disease.

[PRW+09] Conrad Plake, Loic Royer, Rainer Winnenburg, Jörg Hakenberg, and Michael Schroeder. GoGene: gene annotation in the fast lane. *Nucleic Acids Res*, 37(Web Server issue):W300–W304, Jul 2009.

[RRAS08] Loïc Royer, Matthias Reimann, Bill Andreopoulos, and Michael Schroeder. Unraveling protein networks with power graph analysis. *PLoS Comput Biol*, 4(7):e1000108, 2008.

[TAT+06] Nicki Tiffin, Euan Adie, Frances Turner, Han G Brunner, Marc A van Driel, Martin Oti, Nuria Lopez-Bigas, Christos Ouzounis, Carolina Perez-Iratxeta, Miguel A Andrade-Navarro, Adebowale Adeyemo, Mary Elizabeth Patti, Colin A M Semple, and Winston Hide. Computational disease gene identification: a concert of methods prioritizes type 2 diabetes and obesity candidate genes. *Nucleic Acids Res*, 34(10):3067–3081, 2006.

[TCS03] Frances S Turner, Daniel R Clutterbuck, and Colin A M Semple. POCUS: mining genomic sequence annotation to predict disease genes. *Genome Biol*, 4(11):R75, 2003. disease.

[TKP+05] Nicki Tiffin, Janet F Kelso, Alan R Powell, Hong Pan, Vladimir B Bajic, and Winston A Hide. Integration of text- and data-mining using ontologies successfully selects disease gene candidates. *Nucleic Acids Res*, 33(5):1544–1552, 2005. text mining disease.

[vDCK+03] Marc A van Driel, Koen Cuelenaere, Patrick P C W Kemmeren, Jack A M Leunissen, and Han G Brunner. A new web-based data mining tool for the identification of candidate genes for human genetic disorders. *Eur J Hum Genet*, 11(1):57–63, Jan 2003. disease.

[vDCK+05] M. A. van Driel, K. Cuelenaere, P. P C W Kemmeren, J. A M Leunissen, H. G. Brunner, and Gert Vriend. GeneSeeker: extraction and integration of human disease-related information from web-based genetic databases. *Nucleic Acids Res*, 33(Web Server issue):W758–W761, Jul 2005. disease.

[WK07] Tom Walsh and Mary-Claire King. Ten genes for inherited breast cancer. *Cancer Cell*, 11(2):103–105, Feb 2007.

[WSS+02] Michael L Whitfield, Gavin Sherlock, Alok J Saldanha, John I Murray, Catherine A Ball, Karen E Alexander, John C Matese, Charles M Perou, Myra M Hurt, Patrick O

Brown, and David Botstein. Identification of genes periodically expressed in the human cell cycle and their expression in tumors. *Mol Biol Cell*, 13(6):1977–2000, Jun 2002.

[XSN+06] Bing Xia, Qing Sheng, Koji Nakanishi, Akihiro Ohashi, Jianmin Wu, Nicole Christ, Xinggang Liu, Maria Jasin, Fergus J Couch, and David M Livingston. Control of BRCA2 cellular and clinical functions by a nuclear partner, PALB2. *Mol Cell*, 22(6):719–729, Jun 2006.

[YLM+02] Hongmei Yang, Charles H Lin, Gang Ma, Melissa Orr, Michael O Baffi, and Marc G Wathelet. Transcriptional activity of interferon regulatory factor (IRF)-3 depends on multiple protein-protein interactions. *Eur J Biochem*, 269(24):6142–6151, Dec 2002.

Converting DNA to Music: COMPOSALIGN

Todd Ingalls[a] Georg Martius[b] Marc Hellmuth[c] Manja Marz[c*]
Sonja J. Prohaska[c]
[a] Arts, Media and Engineering, Arizona State University, Tempe, AZ 85289-8709, USA
[b] MPI Göttingen, Bunsenstrasse 10, 37073 Göttingen, Germany
[c] Bioinformatics, University of Leipzig, Härtelstr. 16-18, 04107 Leipzig, Germany
[*] Corresponding author

Abstract: Alignments are part of the most important data type in the field of comparative genomics. They can be abstracted to a character matrix derived from aligned sequences. A variety of biological questions forces the researcher to inspect these alignments. Our tool, called COMPOSALIGN, was developed to sonify large scale genomic data. The resulting musical composition is based on COMMON MUSIC and allows the mapping of genes to motifs and species to instruments. It enables the researcher to listen to the musical representation of the genome-wide alignment and contrasts a bioinformatician's sight-oriented work at the computer.

1 Introduction

Evolution and Selection shape the phenotype and genotype of an organism in an unique way. Homologous sequences are derived from a common ancestor by a sequence of selective changes and diverge over time. Multiple selective constraints on a genomic sequence constrain evolution and result in interesting structures, e.g. modularization. Evolutionarily shaped structures become discernible when sequences derived from a common ancestor are aligned. The result as well as the method is called "alignment". The data structure is a matrix, which is not only highly informative and story-telling for a biological expert but also patterned in a sometimes aesthetic way. Some patterns are visible when one of the numerous visualization tools is applied [RPC+00, KKZ+09, GJ05, LBB+07].

Nevertheless, the modular and structured nature of much music has struck many as providing opportunities to understand genomic data by translating it to sound [Ohn93, Ohn87, OO86]. However, only a few trials have been made to use music to convey the patterns to the interested party [HCL+99, HMR00, TM07, LWHC00]. All of them focus on single DNA or protein sequences. Early attempts transposed DNA sequences directly to music [OO86]. The assignment of two notes to each of the four characters (4 nucleotides) allowed for some flexibility to arrange notes to musical themes. Sonification of protein sequences offered a larger set of initial characters (20 amino acids) but was even more constrained and suffered from the creation of a monotonous string of notes without musical depth. Consideration of further properties [HM84, GS95, GS01, DC99] of characters or groups of characters and mathematical derivation based upon this additional information resulted in more exciting music but blurred the underlying information. A tool called

gene2music [TM07] can be used for automated conversion of protein-coding sequences to music. It maps the 20 amino acids on 13 chords, grouping chemically similar characters together while the chord duration is dependent on the frequency of the underlying codon. One system, PROMUSE [HCL$^+$99] deals with sonification of amino acid features as well as structural information and the similarity between related proteins along the sequences. This similarity between proteins and genomic sequences results from common ancestry and light variation and is of central importance to studies in evolution and genomics.

Presentation of highly complex, multidimensional data requires far more channels to transport information than can be handled in the visual channel alone. Visualization and animation are fairly well developed, however, research on the transport of information via sonification is only recently gaining some interest [HR05]. Surprisingly, the complexity of the information transported by the audio channel is usually low, even though musical compositions for entertainment or artistic purposes show highly complex structures. In a multi-media setting, Lodha et al. [LWHC00] showed that sonification can be efficient in disambiguating data in cases where visual presentation alone would be unclear. However, a direct comparison of the efficiency in auditory or visual information uptake is hard to perform. We can expect, however, that the perception of data via sonification and visualization is conceptually very different. Whether this can be beneficial for data presentation is an area we wish to continue to exam.

In this contribution we describe COMPOSALIGN, the first prototype for alignment sonification that translates genome-wide aligned data into a musical composition. Such an acoustic representation requires an unique mapping of alignment information onto musical features. While some mapping is easy to frame, we strive for a intuitive mapping that is easy to perceive and also lives up to the demand to be artistic, pleasant and interesting.

2 Methods

2.1 Mapping

The main focus of our approach is to sonify the presence and absence of characters in the alignment such that their assignment to the corresponding sequence/species is clear. For simplicity, we assume that sequences are from different species, which allows us to refer to "different sequences" as "different species". However, the sources of the sequences are not essential for our theoretical framework but can be added in later steps. Therefore we have chosen the following mapping, formalized as follows:

A *musical motif* or *pattern* is an ordered set of notes and pauses played in one measure with a specific rhythm. Given a set $\mathbb{S}$ of species, a set $\mathbb{I}$ of instruments and a set $\mathbb{P}$ of (different) patterns, we assign to each species an instrument and a pattern played by the assigned instrument. Therefore, we define an injective function $f : \mathbb{S} \to \mathbb{A}$ with $\mathbb{A} = \{(x, y)$ with $x \in \mathbb{I}$ and $y \in \mathbb{P}\} = \mathbb{I} \times \mathbb{P}$, i.e. we assign to *every* species $S \in \mathbb{S}$ a value $f(S)$. Thus it holds $|\mathbb{S}| \leq |\mathbb{A}|$, since f is injective. Many mappings f fulfill the requirement that each species $S \in \mathbb{S}$ is determined and distinguishable from another species by its values

$f(S)$. The remaining degrees of freedom can be used to include auxiliary information such as the phylogenetic relationship of the species. Therefore, we assign instruments to species such that the relationships among the instruments reflect the relationship among species. However, this assignment is done by hand since the relatedness for instruments is a matter of perception. The usage of two independent features (x, y) with $x \in \mathbb{I}$ and $y \in \mathbb{P}$ to encode the species allows us to handle alignments with up to $|\mathbb{I} \times \mathbb{P}|$ species (here $10 \times 10 = 100$) and to represent two-dimensional phylogenetic information as returned by `SplitsTree` [HB06]. In addition to these 100 possibilities we provide 2 further motifs played by drums and cymbals, respectively. These rhythmical motifs are, in particular, useful to sonify outgroup species.

Given a sequence s we consider n units $u_1, \ldots, u_n$ which are, in particular, subsequences of s such that $\bigcup_{i=1}^{n} u_i \subseteq s$. Biologically, these units are referred to as characters in general, "genes" in this contribution. Moreover the units $u_1, \ldots, u_n$ are ordered, such that u_i occurs before u_j whenever $i < j$.

Each unit u_i can be absent, i.e. "0", or directed, i.e. "$+$" or "$-$" if present.

We are now able to define the following matrix A, also called *alignment*.

$$A_{i,j} = \begin{cases} + & \text{, if } u_i \text{ appears in species } S_j \text{ in } + \text{ orientation} \\ - & \text{, if } u_i \text{ appears in species } S_j \text{ in } - \text{ orientation} \\ 0 & \text{, else} \end{cases}$$

This means that all entries $A_{i,j} \neq 0$ for a fixed i are homologous. As explained we have assigned to every species a particular instrument playing a particular pattern. In general, an instrument and the corresponding pattern $f(S_j)$ assigned to species S_j plays during time interval i whenever unit u_i occurs in species S_j, i.e $A_{i,j} \neq 0$. Otherwise the instrument will rest. Whether $f(S_j)$ sounds or not is only dependent on $A_{i,j}$. However, three options can be set to highlight particular information:

Orientation. This option indicates whether a pattern is played forwards or backwards, depending on the orientation of the occurring unit. To be more precise let $f(S_j) = (I, P)$ and let unit u_i occur in species S_j. Then pattern P is played forwards or backwards, whenever $A_{i,j} =$ "$+''$ or $A_{i,j} =$ "$-''$, respectively. As a default $A_{i,j} \neq 0$ is set to $A_{i,j} =$ "$+''$.

Conservation. Conservation information is of central importance for a biological researcher. In some situations, units present in *all* species are the most interesting units which are analyzed in further detail. This option emphasizes units, present/conserved in all species. We have chosen to implement this as a change in harmony. Altering the harmony of a motif is done by a *diatonic transposition*. It shifts every pitch of a pattern by a fixed number of scale steps relative to the pattern's musical scale. To *every* pattern we apply a transposition that is selected with a probability depending on the patterns current scale whenever unit u_i is present in *all* species, Figure 2.1. The probability values, are in part based upon general principles of common practice tonal harmony [KP00] for making well-formed harmonic progressions. Thus a transposition maps a pattern P_j to pattern P_j', which defines the new P_j. This process is well-known as *first-order Markov chain*. For all

Table 1: Transposition probabilities between Markov states: I maj6 – Tonic major sixth, ii m7 – Supertonic minor seventh, iii m7 – Mediant minor seventh, IV maj7 – Subdominant major seventh, V7 – Major Dominant seventh, vi m6 – Submediant minor sixth, vii o7 – Leading-tone diminished seventh.

from/to	I maj6	ii m7	iii m7	IV maj7	V7	vi min6	vii °
I maj6		0.2	0.2	0.2	0.1	0.2	0.1
ii m7				0.2		0.8	
iii m7				0.3		0.7	
IV maj7	0.3	0.4			0.2		0.1
V7	0.8					0.2	
vi min6		0.7		0.3			
vii °	0.8				0.2		

untuned idio- and membranophones P'_j equals P_j (i.e. the motifs cannot be transposed), in our case this holds for drums and cymbals. Notice that patterns P_j and P'_j are perceived as equal up to the change in scale.

Compression. Phylogenetic analyzes focus on differential information. In such a situation, conserved units are considered as uninformative. This option can be used to compress the detailed information in conserved units, while indicating the occurrence of a unit in all species. Under default options, the musical motif is played as it is. If we switch on the compression option and unit u_i is present in *all* species then for all species $S \in \mathbb{S}$ the chosen instruments are simultaneously playing the first note of each of the respective patterns $f(S)$ relative to their orientation, resulting in a so-called *tutti* chord.

2.2 Invertibility of the Mapping

Information representation, visualization as well as sonification, attempts to convey abstract information in intuitive ways. First, we require the information to be formally retrievable from the representation. In mathematical terms, the introduced mapping needs to be bijective, and thus provide an unique way to retrieve the information from the representation. Second, the information must be perceivable to the human ear. Therefore, we want to take advantage of the human sense of hearing.

If all options are set to "off", it is easy to see that we can determine the species S_i by their values $f(S_i)$ since $f : \mathbb{S} \to A' \subseteq \mathbb{A}$ with $A' = \{f(S) \text{ with } S \in \mathbb{S}\}$ is a bijective function.

Orientation – Induced Constraints. If we want to distinguish if a particular unit appears in forward or backward direction in species $S \in \mathbb{S}$ it must be possible to distinguish whether its motif is played forwards or backwards. Thus no symmetric patterns are allowed. Moreover, it is not allowed to have patterns $P, P' \in \mathbb{P}$ such that playing P backwards sounds just like P' in forward direction and vice versa.

Conservation – Induced Constraints. This option requires restrictions on instrument and pattern usage if we want to distinguish different species S by listening to their respective values $f(S)$. We will denote $f_1(S)$ and $f_2(S)$, as the instrument and the pattern of S, respectively. We can distinguish two cases. First, for all pairs of species S and S' holds that

the instruments are unequal ($f_1(S) \neq f_1(S')$). Then the choice of pattern is unrestricted, since each species is determined by its instrument. Second, if some species S and S' have the same instruments we have to distinguish them by their particular pattern. Thus it is not allowed that any composition of transpositions of $f_2(S)$ and $f_2(S')$, resp., leads to one and the same pattern even in scale. If the orientation option is switched on in addition, we have to make sure that no transposition leads to a symmetric pattern. By definition of the term transposition this case cannot occur if no pattern is originally symmetric.

Compression – Induced Constraints. Recall that this option is used to emphasize the occurrence of a unit in all species and to hide detailed information by means of compression. This could be realized in many ways. One of the simplest is the insertion of a single beep. Due to musical reasons, we decided to play the already mentioned tutti chord instead. We are aware that compression causes informational loss in most cases, e.g. orientation. However, we argue that the qualitative information "presence in all species given" is sufficient in most cases. Concerning the remaining cases, we suggest to omit the compression option.

2.3 Implementation

Our program COMPOSALIGN consists of a back end for the composition of the music using COMMON MUSIC [Tau] which runs in Gauche Scheme [Kaw]. COMMON MUSIC is a valuable toolbox for algorithmic composition and also for outputting MIDI data. It allows for a high level description of the compositional elements and convenient definition of the transformation process due to the expressive power of SCHEME. Additionally, there is a web front-end written in Haskell [tC] acting as a CGI program[1], which allows easy usage without the need to install additional software. The data flow is depicted in Figure 1.

The user can upload an input file. After the initial analysis of the file and automatic selection of settings the user has the opportunity to change various parameters. Among these are the selection of the reference sequence and the assignment of musical instrument and motifs to the individual sequences. The default settings are the ones discussed in this paper, however, depending on the biological question, a different assignment might be optimal.

The alignment data is transformed to music based on the settings. For this purpose, an appropriate SCHEME file is generated which is in turn processed by COMMON MUSIC to create a MIDI file. The SCHEME file contains the collection of motifs, the rules for the composition, and the mapping of the species to any of the twelve motifs and available instruments. The user can listen to or download the generated piece of music.

Input. We use a custom comma separated ASCII file type as input which is organized as follows. The input is a $n \times (3 \cdot m)$ matrix consisting of n rows for n units and m blocks of columns each of which holds the genomic start position, end position, and orientation of the unit for every m species. In each row all single columns are separated by a comma. If the unit is not present in a sequence, `NA` is used as the value for all 3 entries (start position, end position, and orientation). Comment lines start with a "#" symbol. The first block of

[1] `http://www2.bioinf.uni-leipzig.de/ComposAlign/`

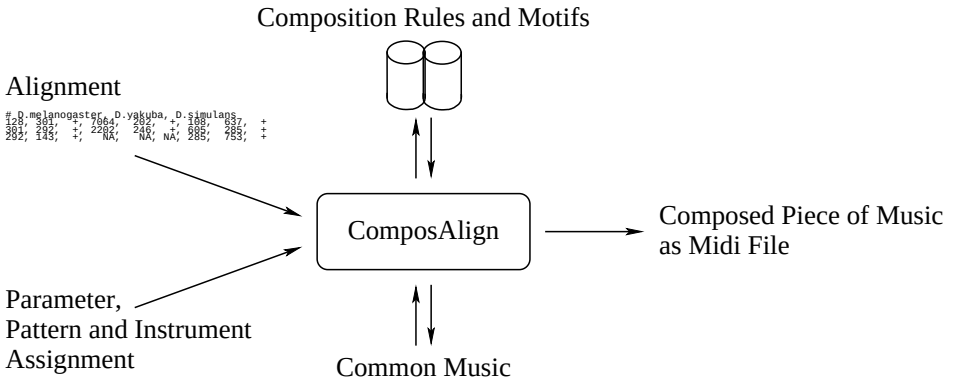


Figure 1: Data flow diagram of COMPOSALIGN. An alignment (input data), parameter settings and the mapping of species to an instrument and pattern are given to COMPOSALIGN via the front-end `www2.bioinf.uni-leipzig.de/ComposAlign`. Using a list of prepared motifs and mapping rules a piece of music is composed.

columns is always treated as the reference species. In principle, it is possible to use any tabular data with absence/presence information for sonification with COMPOSALIGN. An example input file and the corresponding output files can be found in the supplemental material at `http://www2.bioinf.uni-leipzig.de/ComposAlign/`.

3 Application and Results

3.1 Application in Gene Annotation Alignments

For a real data application we have chosen the 12 fly species, each assigned to an unique instrument and pattern. One possible mapping is given by figure 2 and table 3. In all our applications the assignment of species to instruments and patterns fulfills the conditions of an unique mapping for all parameter settings except of the restriction that orientation information is lost in the case of compression, see Section 2.2.

We attempted to sonify data of this kind in a flexible way. These motifs were designed so that they could be placed in various registers. They were also created with varied contours and rhythms to aid in them being individually perceivable in a musical texture.

We used the gene annotations and gene correspondences of chromosome 3R from *D. melanogaster* and the other 11 sequenced Drosophilid genomes as input [Con07]. The input is a matrix $345 \times (3 \cdot 12)$, i.e. 345 genes (units) and 12 species. The genes are given by their genomic sequence interval and their orientation. We sorted the genes by the start position in the reference species (here *D. melanogaster*). Furthermore, we used a relative orientation information, with the orientation of *D. melanogaster* genes set to "+", and the orientation for other genes given by "+" or "-" when the orientation is 'the same' or 'reverse' compared to *D. melanogaster*, respectively.

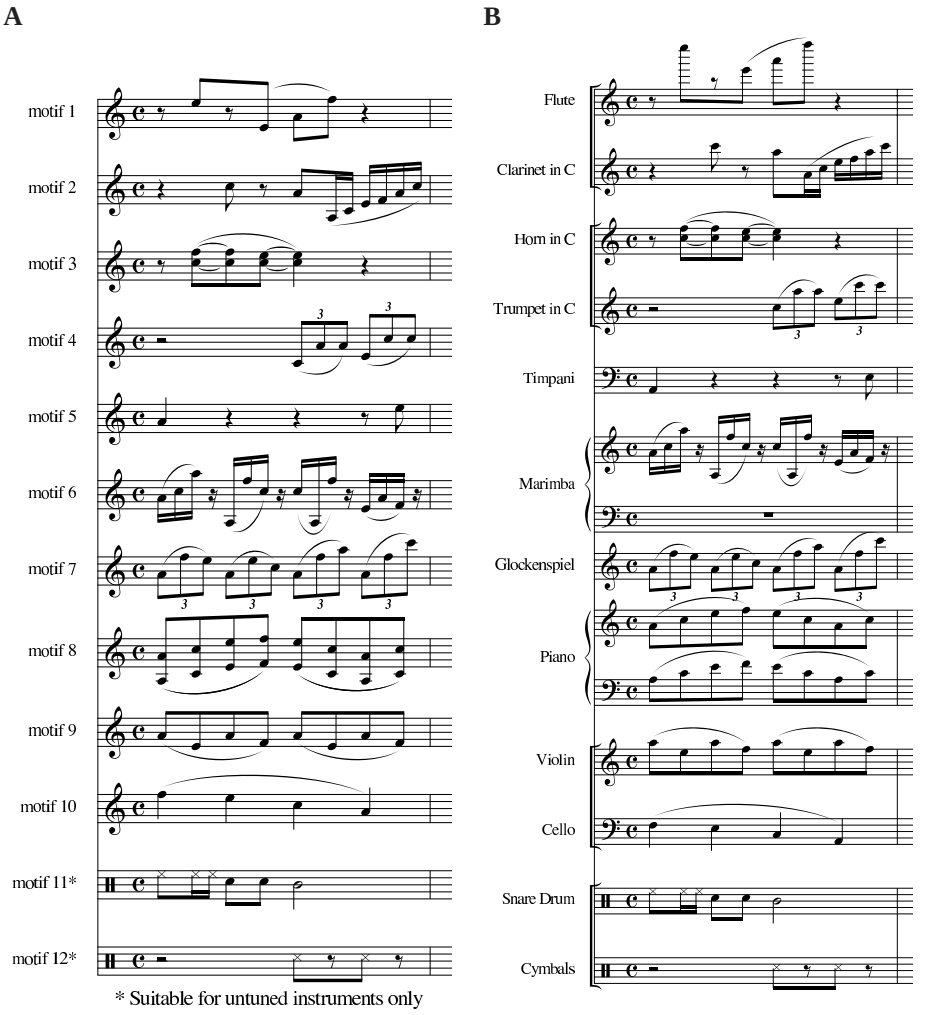


Figure 2: Panel A shows the 12 motifs in forward orientation. Panel B shows the assignment of instruments to the transposed motifs from panel A. The transpositions are based on appropriate instrument ranges. E.g., motif 1 is transposed up two octaves to sound in a more typical flute range. When motif 2 is set to clarinet it is transposed up an octave in order for it to be perceptible when other instruments sound. The motifs 11 and 12 are for untuned instruments only and will be assigned to snare drums and cymbal, respectively, in all our applications.

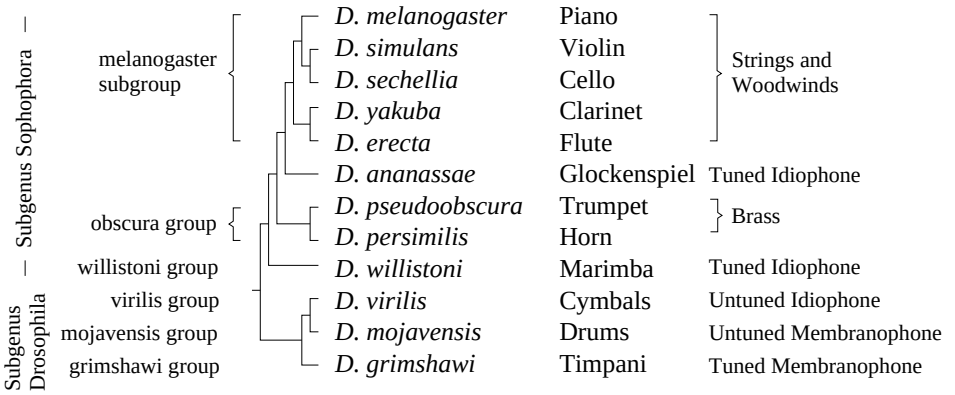


Figure 3: Mapping of fly species to instruments. The tree on the left-hand side represents the topology of the phylogenetic tree [Con07]. Branch lengths are arbitrary.

Moreover, we wanted to have the instrumentation reflect the relative closeness of each species. This closeness is part of a biologist's expert knowledge and reflected in the tree in Figure 3. Of the 12 Drosophila species, five are very closely related – *D. melanogaster*, *D. simulans*, *D. sechellia*, *D. yakuba* and *D. erecta*. One of them, *D. melanogaster*, is the model organism and reference species, which we placed in a continuous motif played by the piano, since this provided the basis for the rest of the music. Furthermore, we looked to place the other four in strings and woodwinds so as to provide some similarity but also enough timbral and register difference so they could be distinguished (Figure 3).

As currently implemented each measure takes 2 seconds resulting in a piece of music, 11.5 minutes long, for all 345 genes.

3.2 Evaluation

In Section 2.2 we have formally shown that the selection of an unique instrument and pattern for each species will allow an unique mapping under certain restrictions. However, it remains to be evaluated how the sonification is perceived by the user. The following analysis of COMPOSALIGN is based on impressions of 50 un-trained, non-musician test persons. The example described in 3.1 is only one of several tested cases with various setting.

Number of Organisms/Instruments. Depending on the education in the arts of the test persons, up to 12 instruments could be recognized. Most people felt confident to distinguish six instruments. If distinction of more (instrument) tracks is desired the majority of people need to be trained to more clearly differentiate the instruments or patterns. We might also want to consider to utilize other types of instrumental or synthesized sounds which would be more easily identified by untrained users.

In the case of 2 or 3 species, the composition was described as "musically pleasing" and

users found it easy to hear which genes were present in which species. However, the ability to resolve the presence/absence pattern decreased rapidly with the number of different instruments and/or motifs playing per measure. Nevertheless, presence/absence of genes that involve groups of species, was still found easy to hear.

Most people who concentrated on a specific instrument and tried to observe the presence/absence at a specific time point, found the correct solution independently of the number of instruments played concurrently.

Conserved Sites – Changes in Harmony. The introduction of changes in harmony based on the local context improved the artistic value of the output and the listeners attention span. All participants had the impression of a much more interesting piece of music, if the conservation option was used. Apart from this aesthetic effect, it also helped emphasized conservation and to draw the listeners attention to conserved regions.

Conserved Sites – Compressed Units. While this sets the presence of m (while m is the total number of species in the alignment) and less than m species clearly apart from each other, it also causes a time compression and allows the user to focus on the data where the absence/presence patters are more informative from a biological perspective. For emphasis of conservation, users preferred the compression option over the conservation option.

All test persons were enthusiastic after including changes in harmony and compressed chords about the musical variability. The outcome was described much more "happier", "interesting, irregular", "less crowded", "rhythmically interesting" and "dramatic". The interrogation also provides an intriguing result in that certain choices that were made largely for aesthetic reasons also appear to make the sonification more legible to users.

Orientation of a Gene – Forward and Backward Motifs. The asymmetry of the individual motifs, some of which are clearly ascending, is an essential attribute to sonify a character's direction information. The character of the motifs allows the user still to identify the mirrored motifs as belonging to the same motif. The results sound pleasant, however most test persons found it difficult to follow which motifs were reversed when several instruments played at the same time. It is unclear if the ear needs some training only or if it might be necessary to explore other strategies which may help in communicating this information.

Mapping – Assignment of instruments and patterns. Using different settings we expected to find combinations that might sound unpleasant. Given an uncommon combination of instruments (e.g. drums, marimba and trumpet) most people found the outcome to be surprisingly rich in character and interesting. When various outputs for the same data file were heard with different instruments and patterns in place, the participants felt that this emphasized the underlying structure in the data.

4 Conclusion and Future Work

To date, COMPOSALIGN is the first prototype of an alignment sonification tool. Existing sonification methods for single biological sequences map each individual characters (e.g.

nucleotides or amino acids) on single notes or chords. We decided to map one character to a measure. This had mainly two effects. First, it added the necessary degrees of freedoms to encode more information and still allowed us to take compositional aspects into account and make it sound pleasant. Second, it stretched the information onto a larger time interval, allowed organized presentation of the information with a measure and therefore insured that the information was easy to perceive. COMPOSALIGN draws its power from the motif design and mapping rules that are modular and flexible. Also, biological sequence alignments are particularly suited for sonification since individual elements of information become blurred in a composition when researcher's become more interested in the overall picture (e.g. groups of species with a conspicuous absence/presence pattern in the sequences). It might turn out that music is a suitable medium to convey information on different levels of resolution at the same time. This leads us immediately to the question: Can sonification compete with or outperform the currently dominating visualization? If not, is sonification able to transport a certain kind of information better than visualization? The omnipresence of visualization might suggest a better performance in all respects. However, to perform a fair test, a competitive sonification tool first needs to be developed. Our prototype is just a small step in this direction.

Based on the experience gained during our project, we intend to construct a mapping for alignments that allows us to add different kinds of additional/contextual information (e.g. lengths of characters, distance between characters, higher order annotation, phastcons score). An interactive interface shall allow the user to edit the parameters on runtime and display the scores and alignment in flying windows. This shall allow the interested user to play (with) his/her alignment.

"Play is the highest form of research." (quote by Albert Einstein)

Acknowledgments.

This work was supported in part by the *Graduierten-Kolleg Wissensrepräsentation* and by a grant (01GQ0432) from the BMBF in the NNCS program. We thank the anonymous reviewers for their valuable and constructive comments.

References

[Con07] Drosophila 12 Genomes Consortium. Evolution of genes and genomes on the Drosophila phylogeny. *Nature*, 450(7167):203–218, Nov 2007.

[DC99] John Dunn and Mary Ann Clark. Life Music: The Sonification of Proteins. *Leonardo*, 32(1):25–32, 1999.

[GJ05] S Griffiths-Jones. RALEE–RNA ALignment editor in Emacs. *Bioinformatics*, 21(2):257–259, Jan 2005.

[GS95] P Gena and C Strom. Musical synthesis of DNA sequences. In *in XI Colloquio di Informatica Musicale*, pages 203–204, Bologna, I, 1995.

[GS01] P Gena and C Strom. A physiological approach to DNA music. In *in Proceedings of CADE 2001*, pages 81–86, Glasgow, UK, 2001. Glasgow School of Art Press.

[HB06] D H Huson and D Bryant. Application of phylogenetic networks in evolutionary studies. *Mol Biol Evol*, 23(2):254–267, Feb 2006.

[HCL+99] M D Hansen, E Charp, S Lodha, D Meads, and A Pang. PROMUSE: a system for multi-media data presentation of protein structural alignments. *Pac Symp Biocomput*, pages 368–379, 1999.

[HM84] K Hayashi and N Munakata. Basically musical. *Nature*, 310(5973):96–96, Jul 1984.

[HMR00] T Hermann, P Meinicke, and H Ritter. Principal Curve Sonification. In *in Proceedings of the Int. Conf. on Auditory Display*, pages 81–86, 2000.

[HR05] Thomas Hermann and Helge Ritter. Crystallization sonification of high-dimensional datasets. *ACM Trans. Applied Perception*, 2(4):550–558, 10 2005.

[Kaw] Shiro Kawai. Gauche Scheme - `http://practical-scheme.net/gauche/index.html`.

[KKZ+09] R M Kuhn, D Karolchik, A S Zweig, T Wang, K E Smith, K R Rosenbloom, B Rhead, B J Raney, A Pohl, M Pheasant, L Meyer, F Hsu, A S Hinrichs, R A Harte, B Giardine, P Fujita, M Diekhans, T Dreszer, H Clawson, G P Barber, D Haussler, and W J Kent. The UCSC Genome Browser Database: update 2009. *Nucleic Acids Res*, 37(Database issue):755–761, Jan 2009.

[KP00] Stefan M. Kostka and Dorothy Payne. *Tonal Harmony, with an introduction to twentieth-century music*. McGraw-Hill, Boston, 4th edition, 2000.

[LBB+07] M A Larkin, G Blackshields, N P Brown, R Chenna, P A McGettigan, H McWilliam, F Valentin, I M Wallace, A Wilm, R Lopez, J D Thompson, T J Gibson, and D G Higgins. Clustal W and Clustal X version 2.0. *Bioinformatics*, 23(21):2947–2948, Nov 2007.

[LWHC00] Suresh K Lodha, Doug Whitmore, Marc Hansen, and Eric Charp. Analysis and user evaluation of a musical-visual system: Does music make any difference. In *in Proceedings of the Int. Conf. on Auditory Displays*, pages 167–172, 2000.

[Ohn87] S Ohno. Repetition as the essence of life on this earth: music and genes. *Haematol Blood Transfus*, 31:511–518, 1987.

[Ohn93] S Ohno. A song in praise of peptide palindromes. *Leukemia*, 7 Suppl 2:157–159, Aug 1993.

[OO86] S Ohno and M Ohno. The all pervasive principle of repetitious recurrence governs not only coding sequence construction but also human endeavor in musical composition. *Immunogenetics*, 24(2):71–78, 1986.

[RPC+00] K Rutherford, J Parkhill, J Crook, T Horsnell, P Rice, M A Rajandream, and B Barrell. Artemis: sequence visualization and annotation. *Bioinformatics*, 16(10):944–945, Oct 2000.

[Tau] Heinrich Taube. Common Music Website - `http://commonmusic.sourceforge.net/doc/cm.html`.

[tC] HASKELL Community. Common Music Website - `http://www.haskell.org/`.

[TM07] R Takahashi and J H Miller. Conversion of amino-acid sequence in proteins to classical music: search for auditory patterns. *Genome Biol*, 8(5):405–405, 2007.

Integration and Visualisation of Multimodal Biological Data

Hendrik Rohn[1], Christian Klukas[1], Falk Schreiber[1,2]

[1] Leibniz Institute of Plant Genetics and Crop Plant Research Gatersleben, Germany
[2] Martin Luther University Halle-Wittenberg, Germany
{rohn,klukas,schreibe}@ipk-gatersleben.de

Abstract: Understanding complex biological systems requires data from manifold biological levels. Often this data is analysed in some meaningful context, for example, by integrating it into biological networks. However, spatial data given as 2D images or 3D volumes is commonly not taken into consideration and analysed separately. Here we present a new approach to integrate and analyse complex multimodal biological data in space and time. We present a data structure to manage this kind of data and discuss application examples for different data integration scenarios.

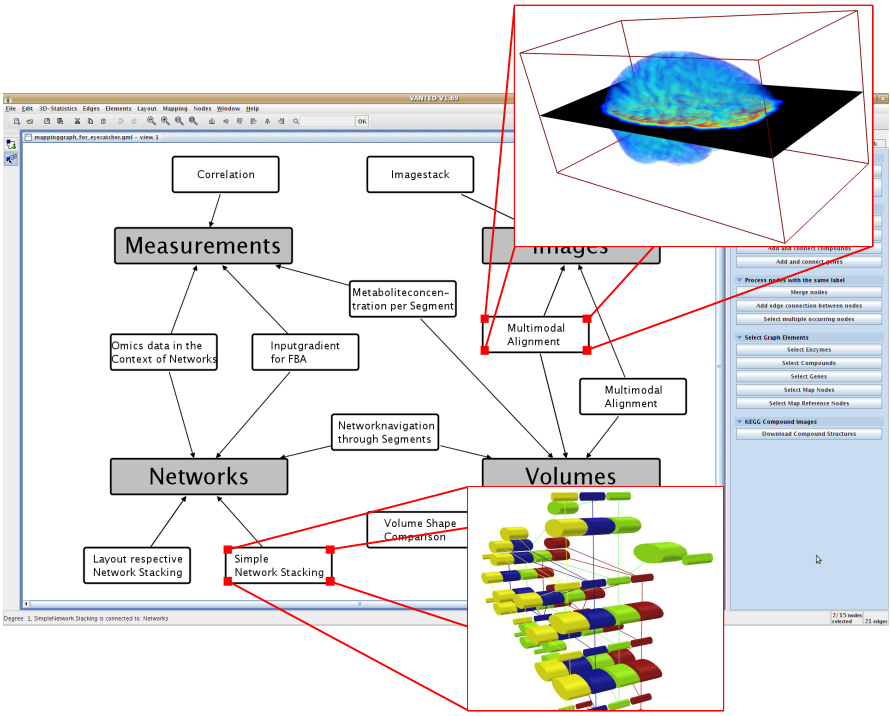


Figure 1: Preview of a prototypic system which integrates, analyses and visualises multimodal biological data based on a mapping graph.

1 Background

Modern life science researchers are able to acquire massive data by using high-throughput techniques. This leads to the accumulation of data from gene and protein activity, protein interaction and metabolite concentration, usually called -omics data. Additionally manifold *in silico* analysis such as flux balance analysis, kinetic modelling, network-centralities and -motifs can gather new information about the intrinsic properties of biological systems. To put this data into biological context network models describing the interactions and relations between biological objects are developed, such as gene regulatory or metabolic networks. Also spatial data, such as structural and functional NMR volume data, histological cross-sections, *in situ* hybridization and surface models, are measured and obtained in increasing quantity and quality and should be considered as valuable parts of models of biological systems.

To answer biological questions often different types of data have to be integrated and considered in spatial and temporal context. Using data mapping one can bring the multimodal data into context to each other, allowing more intuitive analysis, navigation and interpretation of the data. Currently there exist some tools for integration of -omics data into the context of networks [HMWD04, JKS06, KBT$^+$06, Kol02, SMO$^+$03, vIKP$^+$08]. Also some 2D and/or 3D data integration tools exist [Bar06, HLD$^+$07, MPLB07, SWH05]. However, integration of all datatypes in one application with complex mapping possibilities is not considered. In this paper we present a novel approach combining biological -omics data, 2D data, 3D data and network models under consideration of space and time.

The structure of this paper is as follows: First, we propose a data structure to represent and integrate such diverse data types. Second, we discuss different ways of mapping and visualising the multimodal data. Last, we show some example use cases for real-world data mapping applications. Fig. 1 gives an impression of such a system, which is able to intuitively integrate multimodal biological data.

2 Modelling Biological Data

Data is gathered from different parts of a biological system with different resolution. What is the structure of the data? How do we account for the spatial and temporal dimension?

The data structure for multimodal biological data can be seen in Fig. 2. It consists of two main parts: the measured data (highlighted in blue-grey) and annotation data. There are four types of measured data: “Simple measurements” standing for single values, such as the concentration of a metabolite without any further spatial information (-omics data is usually modelled by simple measurements). “Images” represents two-dimensional data such as histological cross-sections or *in situ* hybridisations. “Volumes” denote three-dimensional data such as structural and functional NMR imaging data. “Networks” stand for structural information of biological pathways expressed as a graph. Simple measurements, images and volumes have a “replicateID” to be able to distinguish experiments carried out several times helping to obtain statistical significant results. In addition to

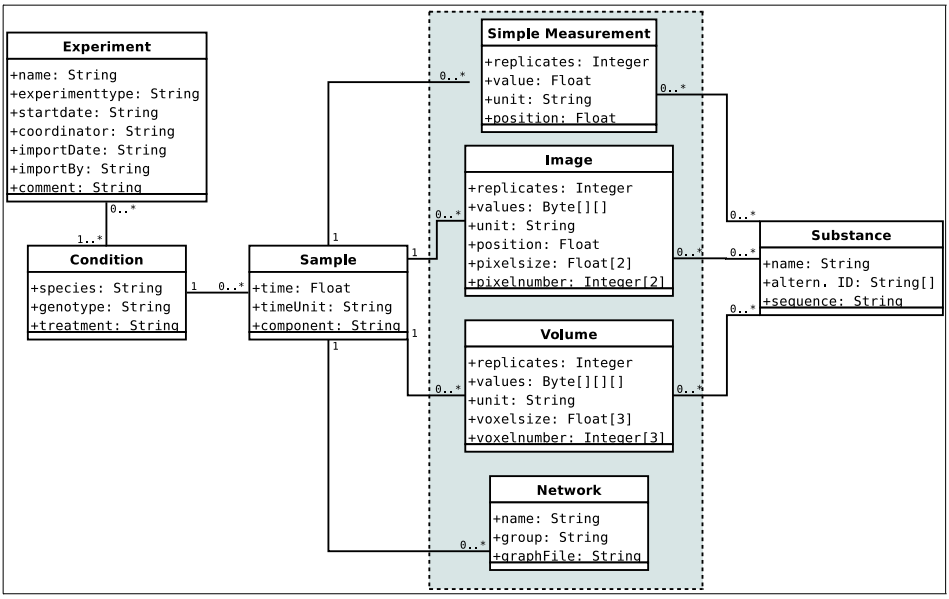


Figure 2: The model for data from experiments. Experiments are carried out under special conditions and consist of a number of samples. These include four different types of measurements: simple measurements, images, volumes and networks. Each measurement except networks may belong to a substance representing the measured biological object.

simple measurements images and volumes include information for their size in respective coordinate systems (pixel- and voxelsize and -numbers). Simple measurements and images also store position information, allowing to describe a vector of simple measurements (e.g. gradients) or images (e.g. position of image in the real biological object) in spatial context. Networks have a name and belong to a certain network group.

An biological experiment has some metadata such as name, coordinator of the experiment, date of import and who imported it. Additional information, for example, about the experiment setup, can be stored unstructured in the "comment" attribute. Each experiment has a number of conditions under which it was carried out: The name of the species is stored in the first attribute. The "genotype" attribute indicates a normal genotype or altered one (e.g. different transgenic lines). "Treatment" may be oxygen-depletion or other environmental properties. Under these conditions some samples are collected at a specific time-point, representing the temporal dimension. Measurements are also collected from a certain "component", for example, chloroplast (cell level) or brain (organ level). Each sample consists of a number of measurements, described above. All measurements but networks describe the quantity of a certain substance measured in the experiment. The substance will serve as an identifier in the data mapping, which will be described in detail in the next section. For simple measurement data the identifier is, for example, a metabolite or a protein, whereas the identifier for two-dimensional data may be the transcript measured in an *in situ* hybridization. For three-dimensional data the substance can be the metabolite the NMR image is based on, e.g., water or protons. Networks are not related to

substances because they only describe structural relations.

The proposed data model is simpler than that one used in the MIAME standard [BHQ+01] (microarray data), PEDRo database [TPG+03] (proteomics data) or ArMet framework [RJL+07] (metabolomics data). The reason is, that we do not want to model the complete experiment workflow. This would include experiment description, design and setup, normalisation methods, annotation methods, the raw and processed data, data standards and more. Instead the focus of our model is on already processed, filtered and normalised experimental data and metadata. Therefore we consider only data required for visualisation and analysis.

3 Integration of Multimodal Biological Data

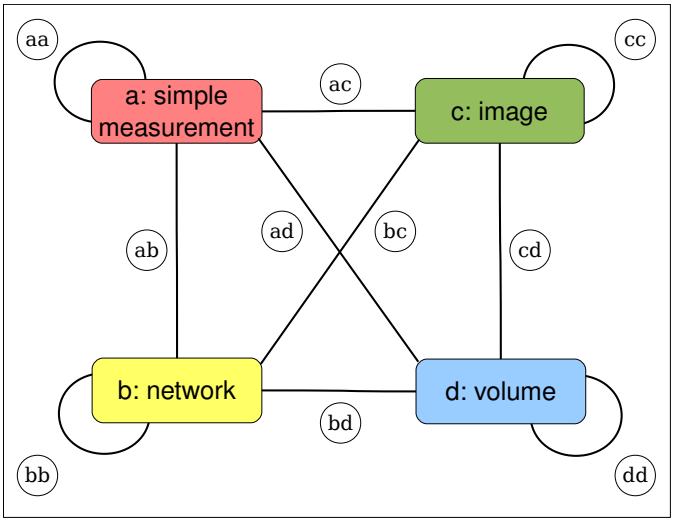


Figure 3: Mapping graph for integrating multimodal data. A node contains all biological data of one type (simple measurements, images, volumes and networks as shown in Fig. 2). An (hyper)edge represents a mapping between one, two or more types of biological data. There are several mappings possible, but for comprehensibility only one- and two-type mappings are shown.

The integration of multimodal biological data is achieved by a mapping graph, whose structure is shown in Fig. 3. The nodes represent different types of biological data, whereby the edges represent a possible mapping between these types. In the following we describe the kinds of data mapping. There are mappings between data of the same type (e.g. “aa”), mappings between data of two different types (e.g. “ac”), mappings between three types (e.g. “acd”), and mappings between all types of data (“abcd”). Note that the mapping usually allows several ways to be processed and visualised. For example, mapping one network on another could be represented as a stacking (see Fig. 4) or as one network showing the difference between both. Here we will give a typical example for some of the

mappings, but many more are possible.

aa: Mapping of simple measurements on simple measurements, for example, visualising the correlation of metabolite concentrations by scatter plots.

bb: Mapping of networks on other networks, for example, network stacking. A detailed use case can be found in Section 4.1.

cc: Mapping of images on images, for example, image stacking of cross-sections. This can be useful if several cross-sections of one object have been obtained and the images have to be placed according to their position in the real object.

dd: Mapping of volumes on volumes, which can be useful for comparing tissue shapes. Here researchers may acquire information how the shape of tissues differ for genetically altered systems.

ab: Mapping of simple measurements on networks, for example, concentration-dependent node colouring. A detailed use case can be found in Section 4.3

ac: Mapping of simple measurements on images, for example, combining high-resolution metabolite concentration data and low-resolution image data showing the concentration distribution in two dimensions.

cd: Mapping of images on volumes, for example multimodal alignment. High-resolution or special coloured cross-sections taken from a biological object are aligned into the three-dimensional representation of the object. A detailed use case can be found in Section 4.2

bd: Mapping of networks on volumes, for example, for navigation. Here segmented tissues of the volume can be used to navigate through the different networks obtained in experiments.

More complex mappings are also possible (e.g. “abc”), but depend on the requirements of life scientists and are therefore often purpose-built. By using this mapping graph the multimodal biological data, consisting of different data types, can be seamlessly combined and integrated into one system.

The data can be imported into the mapping graph using file open dialogs or drag and drop functionality. Such files can be exported from various tools and databases, e.g. KEGG, MetaCrop, AMIRA [SWH05]. Several data formats will be accepted, e.g. GML, GraphML, SBML and KGML for networks, CSV textfiles and Excel spreadsheets for simple measurements, VRML and Analyze 7.5 for volumes and PNG, JPEG and TIFF for images.

4 Use Cases

To show the functionality of the integration via a mapping graph we will highlight four exemplary use cases in detail.

4.1 Network Stacking

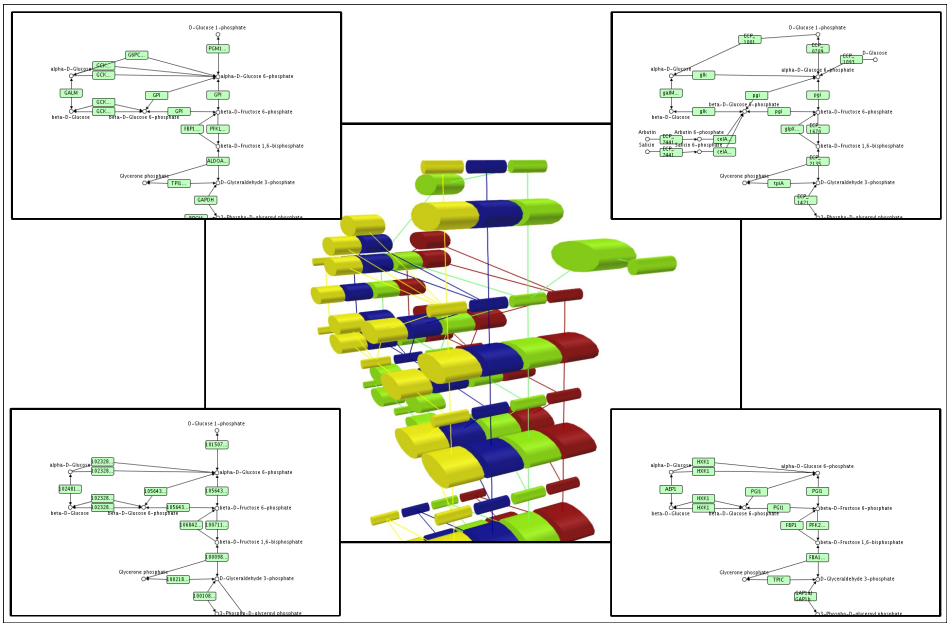


Figure 4: Use case network stacking: Four networks of glycolysis from different species are stacked in the three-dimensional space to support exploration of structural differences such as missing metabolites and interactions.

The first use case is network stacking (see Fig. 1 and 4), which is an instance of mapping case “bb” and represented in the mapping graph by an edge between networks (see Fig. 5). Here several networks are aligned allowing visual comparison of network properties: a network is mapped at one plane lying in a three-dimensional space. The next network and its plane are aligned in such a way that the corresponding nodes in the networks are stacked on top of each other respecting the layouts (see [BDS04] for further details). Additional networks can be aligned in the same way creating a $2\frac{1}{2}$D-stacking of networks. This representation allows to explore structural differences and similarities such as missing metabolites, unique interactions and conserved motifs for different species or genotypes.

4.2 Multimodal Alignment

Multimodal Alignment is a technique to align two-dimensional images into three-dimensional volumes. Often the images are high-resolution cross-sections through the biological object, allowing high detailed analysis and yield information obtained with specific methods such as *in situ* hybridisations. Volumes on the other hand represent lower-resolution three-dimensional information of an object. The idea is to combine both information,

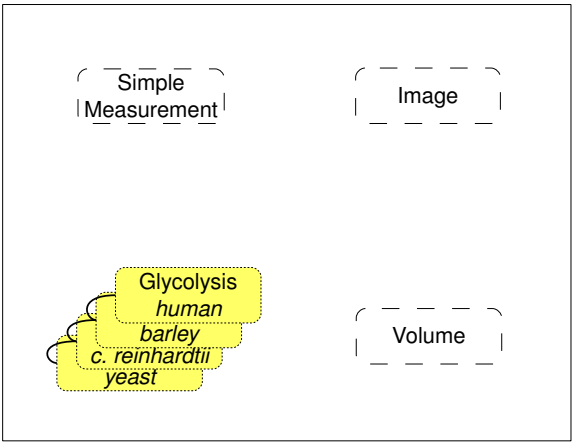


Figure 5: Instance of the mapping graph for use case network stacking. The mapping graph consists of four glycolysis networks of different species.

which means the image has to be moved to the correct position in the volume. To align the images one can use cross-correlation or other methods (e.g. some algorithms are implemented in the Insight Segmentation and Registration Toolkit [YAL02]). This means the second use case is an instance of mapping case “cd”. An example of the result of such a mapping can be seen in Fig. 1 on the first page.

4.3 Omics Data in the Context of Networks

For the analysis of biological data it is useful to apply an integrated view on the measured data and its related background information, such as metabolic pathways or regulative processes. For this purpose one can map the biological data (e.g. protein activity, metabolite concentration, etc.) to structural information such as glycolysis pathway, which represents a mapping of type “ab”. An automatic mapping of experiment data onto relevant network elements occurs if the measured data and the network nodes have common identifiers. For the visualisation of mapped data the display of multiple mapped datasets for a single network element is supported. Using line charts, bar charts and similar techniques the scientist is able to visualise more complicated datasets, such as data from different time points, experimental conditions and replicates. For further information about this mapping see [JKS06].

4.4 Oxygen Gradient and Flux Balance Analysis

The last exemplary use case consists of a mapping “ab” and can be seen in Fig. 6 and 7. At the top of Fig. 6 there is an oxygen gradient, which consists of a number of simple mea-

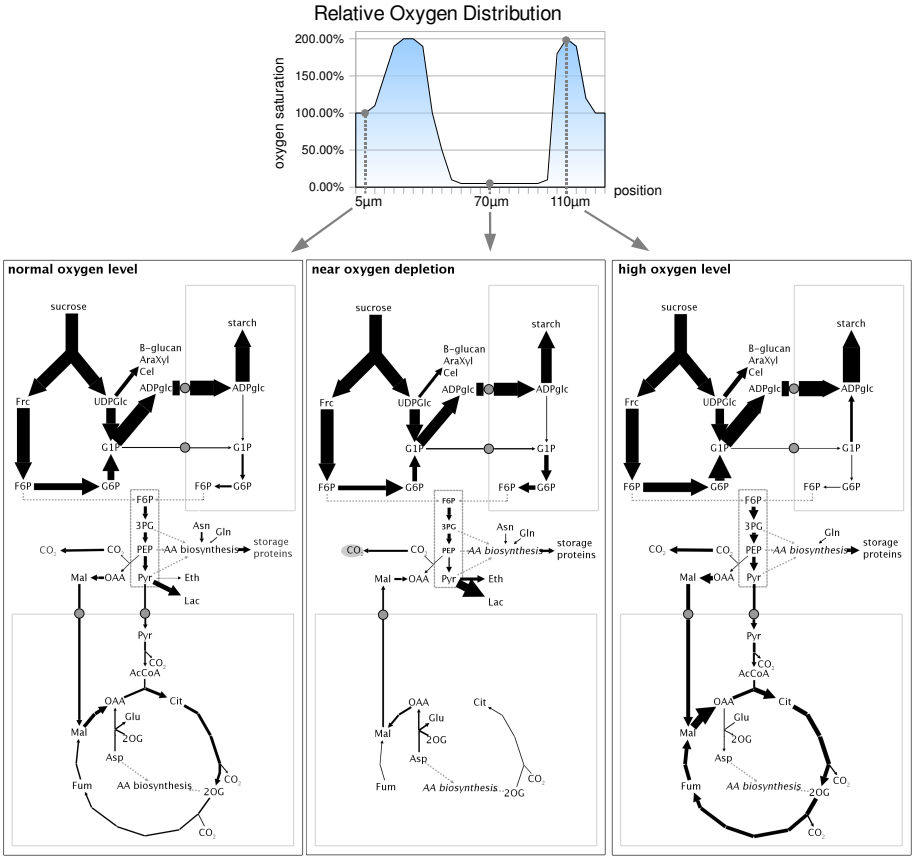


Figure 6: Use case flux balance analysis: One-dimensional oxygen gradient used as an input for flux balance analysis [KPE03]. The simulation results for different oxygen levels are mapped to the glycolysis network. The visualisation of the data shows, that the higher the oxygen concentration the higher the starch accumulating flux (middle, left, right).

surements. Such gradients could be obtained as time series or by a probe moving through a tissue and measuring the relative oxygen level for different positions (see [RWW+04]). The values of this gradient are used as an input for flux balance analysis [KPE03], which models fluxes in networks on basis of structural and stoichiometric information. Some starting concentrations are necessary, which are taken from the oxygen gradient as input for different scenarios: The middle network visualises the fluxes near oxygen depletion, the left one normal oxygen level and the right one higher oxygen level than in the air. In this way respective flux visualisations can be shown for different scenarios, based on simple measurements.

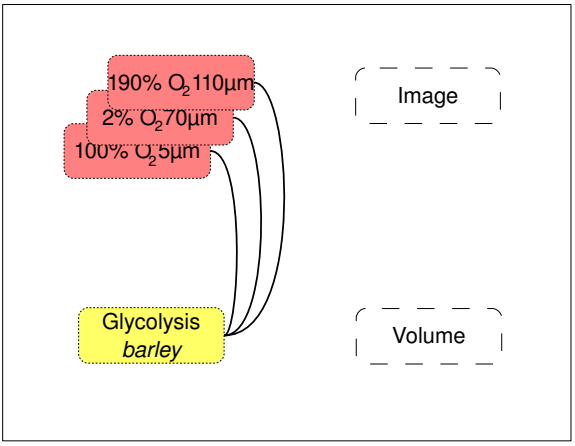


Figure 7: Instance of the mapping graph for use case flux balance analysis. The mapping graph consists of an oxygen gradient (simple measurement), which is mapped to a glycolysis network of barley and used for flux balance analysis.

5 Conclusion and Outlook

Using high-throughput methods biological researchers gather lots of data of different types from multiple -omics areas, network models and spatial data. For intuitive exploration of this data we propose a data structure representing the biological data and supporting all necessary mapping and data exploration methods. The biological data was integrated using a mapping graph, which allows intuitive combination of data. Its nodes represent the data types and its edges represent mappings between data types. Some mapping types were analysed and finally four exemplary use cases of data integration were described in detail.

The data structure and some of the mappings and use cases are implemented on the basis of the VANTED system [JKS06] in Java3D to provide scientists with the possibility to handle not only -omics data and network models, but also to account for two- and three-dimensional data in one system. We plan to complete the development and implementation of further mapping and interaction methods together with life scientists before releasing it as an Open Source Add-On for VANTED.

Acknowledgements

We would like to thank Rainer Pielot for help with the multimodal alignment and Eva Grafahrend-Belau for help with the flux balance analysis. This work was supported by grant BMBF *0315044A*.

References

[Bar06] K. U. Barthel. 3D-Data Representation with ImageJ. In *ImageJ Conference*, 2006.

[BDS04] U. Brandes, T. Dwyer, and F. Schreiber. Visual Understanding of Metabolic Pathways Across Organisms Using Layout in Two and a Half Dimensions. *Journal of Integrative Bioinformatics*, 1(1):119–132, 2004.

[BHQ+01] A. Brazma, P. Hingamp, J. Quackenbush, G. Sherlock, P. Spellman, C. Stoeckert, J. Aach, W. Ansorge, C. A. Ball, H. C. Causton, T. Gaasterland, P. Glenisson, F. C. Holstege, I. F. Kim, V. Markowitz, J. C. Matese, H. Parkinson, A. Robinson, U. Sarkans, S. Schulze-Kremer, J. Stewart, R. Taylor, J. Vilo, and M. Vingron. Minimum information about a microarray experiment (MIAME) - toward standards for microarray data. *Nature Genetics*, 29(4):365–371, 2001.

[HLD+07] T. Hjornevik, T. B. Leergaard, D. Darine, O. Moldestad, A. M. Dale, F. Willoch, and J. G. Bjaalie. Three-Dimensional Atlas System for Mouse and Rat Brain Imaging Data. *Frontiers in Neuroinformatics*, 1:1–12, 2007.

[HMWD04] Z. Hu, J. Mellor, J. Wu, and C. DeLisi. VisANT: an online visualization and analysis tool for biological interaction data. *BMC Bioinformatics*, 5:17.1–8, 2004.

[JKS06] B. H. Junker, C. Klukas, and F. Schreiber. VANTED: A system for advanced data analysis and visualization in the context of biological networks. *BMC Bioinformatics*, 7:109.1–13, 2006.

[KBT+06] J. Köhler, J. Baumbach, J. Taubert, M. Specht, A. Skusa, A. Ruegg, C. Rawlings, P. Verrier, and S. Philippi. Graph-based analysis and visualization of experimental results with ONDEX. *Bioinformatics*, 22(11):1383–1390, 2006.

[Kol02] F. A. Kolpakov. BioUML - Framework for visual modeling and simulation of biological systems. In *International Conference on Bioinformatics of Genome Regulation and Structure*, pages 130–133, 2002.

[KPE03] K. J. Kauffman, P. Prakash, and J. S. Edwards. Advances in flux balance analysis. *Current Opinion in Biotechnology*, 14(5):491–496, 2003.

[MPLB07] E. B. Moore, A. V. Poliakov, P. Lincoln, and J. F. Brinkley. Mindseer: A Portable and Extensible Tool for Visualization of Structural and Functional Neuroimaging Data. *BMC Bioinformatics*, 8:389.1–12, 2007.

[RJL+07] D. V. Rubtsov, H. Jenkins, C. Ludwig, J. Easton, M. R. Viant, U. Günther, J. L. Griffin, and N. Hardy. Proposed reporting requirements for the description of NMR-based metabolomics experiments. *Metabolomics*, 3(3):223–229, 2007.

[RWW+04] H. Rolletschek, W. Weschke, H. Weber, U. Wobus, and L. Borisjuk. Energy state and its control on seed development: starch accumulation is associated with high ATP and steep oxygen gradients within barley grains. *Journal of Experimental Botany*, 55(401):1351–1359, 2004.

[SMO+03] P. Shannon, A. Markiel, O. Ozier, N.S. Baliga, J.T. Wang, D. Ramage, N. Amin, B. Schwikowski, and T. Ideker. Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Research*, 13(11):2498–2504, 2003.

[SWH05] D. Stalling, M. Westerhoff, and H.-C. Hege. *Amira: A highly interactive system for visual data analysis*, chapter 38, pages 749–767. Academic Press, Inc. Orlando, FL, USA, 2005.

[TPG+03] C. F. Taylor, N. W. Paton, K. L. Garwood, P. D. Kirby, D. A. Stead, Z. Yin, E. W. Deutsch, L. Selway, J. Walker, I. Riba-Garcia, S. Mohammed, M. J. Deery, J. A. Howard, T. Dunkley, R. Aebersold, D. B. Kell, K. S. Lilley, P. Roepstorff, J. R. Yates, A. Brass, A. J. Brown, P. Cash, S. J. Gaskell, S. J. Hubbard, and S. G. Oliver. A systematic approach to modeling, capturing, and disseminating proteomics experimental data. *Nature Biotechnology*, 21(3):247–254, 2003.

[vIKP+08] M. van Iersel, T. Kelder, A. Pico, K. Hanspers, S. Coort, B. Conklin, and C. Evelo. Presenting and exploring biological pathways with PathVisio. *BMC Bioinformatics*, 9:399.1–9, 2008.

[YAL02] T. S. Yoo, M. J. Ackerman, and W. E. Lorensen. Engineering and algorithm design for an image processing API: a technical report on ITK - the insight toolkit. In *Proceedings of Medicine Meets Virtual Reality*, pages 586–592, 2002.

A new method for the design of degenerate primers and its use to identify homologues of apomixis — associated genes in Brachia

Eduardo Gorrón[1,2], Fausto Rodríguez[1], Diana Bernal[1], Silvia Restrepo[2] and Joe Tohme[1,*]

[1]Centro Internacional de Agricultura Tropical (CIAT), A.A. 6713, Cali, Colombia
[2]Universidad de los Andes, Carrera 1 No 18A-10, Bogotá, Colombia

[*]Corresponding author, j.tohme@cgiar.org

Abstract: Apomixis is a reproductive phenomenon that occurs in flowering plants. It allows a plant to produce asexual seeds, with its same genetic constitution. The existence of a genetic basis for apomixis is crushing, but the molecular mechanisms are unclear. The search for the "master apomixis gene" had led to the isolation of various candidate transcripts, but neither of them could be confirmed in different plant species. Here we tried to isolate homologues to all those transcripts in one unique plant, Brachiaria. In order to achieve this, a new method for degenerate primer design was employed, since classical methods have proven to be unsuccessful. We used multiple local alignments, instead of global, with the Multiple Expectation – Maximization for Motif Elicitation (MEME) algorithm, to find conserved blocks and motifs. These alignments were followed by ePCR simulation and standard primer pair design programs. The method demonstrated to be useful to amplify fragments homologous to genes poorly molecular and biologically characterized, with which multiple global alignments showed no conserved regions. The obtained amplicons showed differential expression according to tissue in some cases. This technique can be used to design degenerate primers in cases where one sequence exhibits poor global similarity and has low biological characterization, and it is useful to amplify orthologous genes in an organism weakly described at the molecular level.

Keywords: Primer design, apospory, degenerate primers, MEME

1 INTRODUCTION

The development of degenerate primer pairs often involves the amplification of orthologous DNA fragments which have little conservation, even in the same taxonomic group. Therefore, typical tecnhiques used to design them are unable to find the conserved blocks necessary to generate the oligonucleotides. One reason for this is the employment of multiple global alignment algorithms as the starting point to find the blocks [GMDK05, KCSW94]. The use of this kind of algorithms demands a relatively high degree of conservation along the entire sequence and between all the aligned sequences. If the analyzed

DNA region lacks it, we are forced to look for specific primers for each genera if we try to amplify it in different members of a taxa, or to find alternate ways to do that.

We faced this problem in our laboratory in the analysis of sequences related to a phenomenon called apomixis. Apomixis consists in the ability of some flowering plants to produce viable seeds by asexual ways. It has two basic characteristics: the avoiding of meiosis or the degeneration of meiotic-derived cells, a situation known as apomeiosis, and the generation of a megagametophyte genetically equal to the surrounding tissues [AJ92]. Many observations in different apomictic plants [AJ92, BBK00, KG03, OARH98, POL$^+$97, Mat89] has led to the conclusion that apomixis has a very strong genetic component, and this phenomenon is regulated by one or a few genes [AHOA05, AMR$^+$05, LBC$^+$02, PMB$^+$04, PEO$^+$98]. Some previous works have found some candidate transcripts [CCA$^+$99, GRR$^+$00, LAP$^+$97, PEM$^+$01, RCD$^+$03, VCNB$^+$96], but neither could confirm clearly any of them. For these reasons, trying to analyze expression patterns of candidate transcripts in one plant species is very important, as a first step to obtain a general and clearer molecular model. To do this, we tried to design degenerate primers from many reported sequences related to apomixis in different plant species, in order to obtain all their orthologous counterparts in the genus Brachiaria. However, many of the reported transcripts have not obvious molecular functions and, in fact, are sequences with unknown function in many cases. There are not reference sequences with high similarities with them. Multiple global alignments did not show clear conserved regions. Hence, primer design, even degenerate primer design, was not possible with the standard bioinformatic techniques [GMDK05, KCSW94].

In this work, we propose the use of a new method to design primers, which could be used to create oligonucleotides for sequences with poor global similarity to their suspected homologues or when they don't show well conserved blocks in multiple global alignments. The technique is based in the use of multiple local alignments instead of global, process carried with the Multiple Expectation – Maximization for Motif Elicitation (MEME) algorithm. This algorithm was initially suited to look for conserved motifs in protein and DNA sequences, in order to identify possibly functional homologies between them. In our study we used it to search for conserved regions long enough to design an acceptable primer pair. The result of MEME is powered with a confirmation with electronic PCR (e-PCR) over the list of sequences used and a verification of annealing temperatures, secondary structure formation and primer dimers with standard programs. With it, amplification of homologues of one gene in a related species, which is poorly characterized at the molecular level, is possible, as we confirmed this with laboratory assays.

2 MATERIALS AND METHODS

Obtention of sequences related to apomixis. Sequences associated to the apomictic trait were retrieved from the GenBank database, after a literature review about expression analysis and candidate genes in different apomictic plant species. The original GenBank accesion number was used to reference the obtained primer pairs and the respective results.

Construction of a database for BLAST search and BLAST analysis. In order to look for homologues of these transcripts in relative species of Brachiaria, we used the TIGR databases TIGR Gene Indices (`http://compbio.dfci.harvard.edu/tgi/`) and TIGR Plant Transcript Assemblies (`http://plantta.tigr.org`, both consulted on September 2007). The assemblies from all species included in the families Alliaceae and Poaceae were downloaded, to construct a database of EST assemblies specific for relatives of Brachiaria. The sequences obtained from GenBank as associated to apomixis were used as queries in searches against this database using the BLAST algorithm. The maximum accepted e-value in this test was 10-6; sequences without homology under this value were discarded.

MEME alignment. Each sequence and the group of sequences similar to it according to BLAST were aligned, initially using T-COFFEE (`http://tcoffee.vital-it.ch/cgi-bin/Tcoffee/tcoffee_cgi/index.cgi? stage1=1& daction=TCOFFEE::Regular& referer0=embnet`) to see global similarity, and then with MEME (`http://meme.sdsc.edu/meme4_1/cgi-bin/meme.cgi`) [BWML06]. The conditions of this last alignment included a minimum length of 18 nt and a maximum of 25, and minsites = 2/3 of the sum of sequences, conditions that was considered optimal for primer design. In order to maintain a relatively equal representation of different species, when some plant species predominated in BLAST results (for example rice), some of its associated sequences were eliminated. Sequences with low similarity in MEME alignment were also deleted. With the remaining sequences, a second round of MEME was made.

Construction of motif pairs and e-PCR simulation. The 20 first motifs obtained in each MEME run were extracted and ordered by their score given in the program. All possible pairs between them were virtually assembled, creating all the combinations of motifs, in order to test each pair as primer. Every pair was run over the group of corresponding sequences (the sequences used in the second MEME alignment) in the e-PCR program [Sch97] under the condition that amplicon size must be at least 200 bp in length.

Final selection of primer pairs and synthesis. Motif pairs, which show an adequate amplicon size in e-PCR, were manually verified in their corresponding MEME alignment. First, the motif pair with the highest score was evaluated. The presence of the original query sequence and the number of degenerations was verified. If the query sequence was absent in motif alignments and/or the number of degenerations was greater than 12 between both motifs, the pair was discarded and the test continued with the next pair. If we saw some pattern associated with taxa (for example an A shared between all members of Panicoideae) in a degenerate position, the nucleotide present in Panicoideae was left, and no degeneration was considered. When one motif pair was acceptable, it was tested in NetPrimer (`http://www.premierbiosoft.com/netprimer/netprlaunch/netprlaunch.html`), in order to verify annealing temperatures, second structure formation and possible primer dimers. If the pair had undesirable Tm, a Tm difference greater than 5ºC, or formed dimers or secondary structures with high ΔG (over

-8KJ/mol), bases of one or both motifs were eliminated, until the conditions were acceptable. The minimium size allowed for each motif was 17 nt. If the motif pair, even with these adjustments, did not show to be adequate, the pair was discarded and the test continued with the next pair. Finally, when one pair was accepted, the degenerations were put in their positions and it was synthesized as degenerate primers (Integrated DNA Technologies, Coralville, Iowa). If neither motif pair had the required conditions, the analysis for this particular sequence stopped.

Plant material. Brachiaria decumbens accession 16494 (CIAT code) was employed. Pistils of 1,7-2,2 mm in length were collected with all RNA extraction cautions, and they were deposited in 600 µl of RNAlater (Ambion Inc, Austin, Texas). A total of 200 pistils were taken. This sample was maintained in -80°C until RNA extraction. Leaves, roots and stems of plants were also taken.

RNA extraction and cDNA synthesis. The pistils sample was centrifuged at 14000rpm by at least 20 minutes to allow deposit of pistils at the bottom. RNAlater solution was discarded and pistils were macerated in liquid nitrogen. Total RNA was extracted using Picopure RNA Isolation Kit (Molecular Devices, Sunnyvale, CA), following manufacturer's instructions. Leaves, roots and stems were also macerated in liquid nitrogen and total RNA was extracted using Trizol (GibcoBRL, Carlsbad, CA). cDNA of all type of samples were synthesized using the Creator SMART cDNA Library Construction Kit (Clontech, Mountain View, CA).

Amplification with the designed degenerate primers. PCR reactions were carried with cDNA from pistils (reproductive tissue), in one hand, and a bulk of cDNA from leaves, roots and stems (vegetative tissue), on the other hand, both diluted at 5ng/ μl. PCR reactions were initially tested with the next mix: Tris-HCl 20mM, KCl 50mM, MgCl2 1,4mM, dNTPs 0,2mM each one, 1 U Taq Polimerase, each primer at 0,6 μM, 5 μl of diluted DNA, final volume 25 μl. PCR cycling program was 94ºC 2 minutes, followed by 40 cycles of 94ºC 30 seconds, the theoretical annealing temperature of each primer pair by 30 seconds, and 72ºC 1 minute 30 seconds, and finally 72ºC 5 minutes. If amplification was not observed under these conditions for a particular primer pair, the DNA concentration was increased, making a lower DNA dilution. When more DNA did not improve results, the annealing temperature in the program was reduced another 5ºC. The primer pairs that did not amplify with these changes were not assayed again.

Sequencing of resulting amplicons. Amplicons obtained in the PCR reactions were cloned using pGEM-T Easy Cloning Kit (Promega, Madison, WI) and sequenced with ABI PRISM Big Dye kit (Applied Biosystems, Foster City, CA). Every different amplicon was sequenced three times. Sequences were edited and analyzed using Sequencher 4,8 (Gene Codes Corporation, Ann Arbor, MI).

Bioinformatic analysis of obtained sequences. The obtained sequences were analyzed using BLAST against the GenBank database. Each sequence was compared against its original query, obtained initially from GenBank, using BLAST2Seqs (`http://www.ncbi.nlm.nih.gov/blast/bl2seq/wblast2.cgi`) (blastn algorithm) and LALIGN (`http://www.ch.embnet.org/software/LALIGN_form.html`) (global alignment, the other parameters by default). This comparison was also made when we obtained, with the same primer pair, amplicons of different sizes when the PCR was made in reproductive or vegetative tissue.

3 RESULTS

Sequences obtained from GenBank database. After the literature review, only 27 sequences associated to apomixis in GenBank were obtained (Table 1). Many of the sequences are short sequences (200-400 bp) and are not highly similar to known sequences. Two sequences from Saccharomyces cerevisiae were also included; their corresponding two genes were reported as linked with a yeast meiotic phenomenon also called apomixis [BMG83]. When BLAST analysis was made, only 11 of the 27 sequences showed results with similarities with e-value under 10-25. For this reason, the parameters of algorithm were changed and similarities with e-value under 10-20 were accepted. This allowed the inclusion of another 5 sequences in the analysis. With a final change in the limits of BLAST search to tolerate results with similarities with e-value less than 10-6, nine more sequences were included, and three sequences (u65386, u65387 and yscspoa) did not show any acceptable results and were no more analyzed. For the two sequences of S. cerevisiae, BLAST was made in GenBank database too and plant results were also included in further examination.

Multiple alignments. Many of the sequences did not show very high similarity with their respective BLAST results. So, as we expected, T-COFFEE alignments showed no satisfactory global similarity in any case (data not shown). MEME alignments, however, were capable to reveal motifs with good levels of conservation. More than half of the results exhibited a common arrangement of motifs between the analyzed sequences, indicating a global similarity that could not be well resolved with common algorithms of multiple global alignments (Figure 1a). This was a good precedent for primer design, because it proved the existence of well-conserved regions and similarity in these cases. Nevertheless, 5 of the remaining 24 query sequences did not have a satisfactory similarity in its alignment, especially because the query did not share motifs with the others (Figure 1b). The first eleven motifs have an acceptable level of degeneracy, and they were preferred to make subsequent scrutiny.

e-PCR and primer design. The MEME motifs were virtually combined in pairs and assayed in electronic PCR against every set of sequences (the query plus their BLAST results). An average between 30 and 50 pairs of motifs had an amplicon of at least 200 bp,

GenBank code	Name
AB000809	Panicum maximum A2-134 mRNA, complete cds
AF242539	Paspalum notatum clone apo417 apomixis-related protein 1 mRNA, partial cds
AF475105	Triticum aestivum apomixis-associated protein mRNA, partial cds
AJ786393	Poa pratensis APOSTART2 gene for START domain-containing protein, exons 1-22
AJ810708	Poa pratensis mRNA for RAB1-like (rab1-2 gene)
AJ810709	Poa pratensis mRNA for Ankyrin protein kinase-like (apk gene)
AJ810710	Poa pratensis mRNA for Armadillo-like (arm gene)
AJ841698	Poa pratensis serk1 gene for somatic embryogenesis receptor-like kinase 1, exons 1-11, allele 1.
AY375366	Pennisetum squamulatum Opie-2-like retrotransposon, partial sequence
D37938	Pennisetum ciliare apomixis-associated mRNA, clone:psb C
D37939	Pennisetum ciliare apomixis-associated mRNA, clone:psb3-1a
D37940	Pennisetum ciliare apomixis-associated mRNA, clone:psb H.
EF517497	Cenchrus ciliaris apomixis-related protein Pca21 (Pca21) mRNA, complete cds
EF517498	Cenchrus ciliaris apomixis-related protein Pca24 (Pca24) mRNA, partial cds
EF530198	Hieracium caespitosum DMC1 gene, complete cds
EF530199	Hieracium piloselloides putative ubiquitin associated/TS-N domain-containing protein (UBA) mRNA, complete cds
M32653	S. cerevisiae sporulation protein (SPO16 and SPO12) genes, complete cds
M38357	Saccharomyces cerevisiae meiosis-specific protein (SPO13) gene, complete cds, clone p(SPO13)1
U40219	Pennisetum ciliare possible apospory-associated protein mRNA, complete cds
U65082	PCU65382 Buffelgrass obligatory sexual ovary (M.A.Hussey) Cenchrus ciliaris cDNA, mRNA sequence
U65383	PCU65383 Buffelgrass obligatory sexual ovary (M.A.Hussey) Cenchrus ciliaris cDNA, mRNA sequence
U65384	PCU65384 Buffelgrass obligatory sexual ovary (M.A.Hussey) Cenchrus ciliaris cDNA, mRNA sequence
U65385	PCU65385 Buffelgrass obligatory sexual ovary (M.A.Hussey) Cenchrus ciliaris cDNA, mRNA sequence
U65386	PCU65386 Buffelgrass obligatory apomictic ovary (M.A.Hussey) Cenchrus ciliaris cDNA, mRNA sequence
U65387	PCU65387 Buffelgrass obligatory apomictic ovary (M.A.Hussey) Cenchrus ciliaris cDNA, mRNA sequence
U65388	PCU65388 Buffelgrass obligatory apomictic ovary (M.A.Hussey) Cenchrus ciliaris cDNA, mRNA sequence
U65389	PCU65389 Buffelgrass obligatory apomictic ovary (M.A.Hussey) Cenchrus ciliaris cDNA, mRNA sequence

Table 1: Sequences retrieved from GenBank and used for degenerate primer design

which were organized by their score. The MEME alignment of every pair, starting with the pair with the highest score, was manually revised. We saw some cases in which one tribe or genus has one nucleotide and the others had another nucleotide, indicating a taxonomic pattern of the degeneracy (Figure 2). In that case, that degeneracy was not considered and the nucleotide shared by members of the tribe Panicoideae, to which Brachiaria belong, is putted in that position.

Many motif pairs, however, were discarded in the verification step with NetPrimer, because their Tms were very different and/or the predicted dimers had ΔG over 8KJ/mol. So, the finding of the adequate motif pair took a long time. This manual process of verification was particularly very laborious and time consuming.

Finally, primer pairs were designed for 22 sequences (Table 2), and for another two this was impossible. The degeneracies given by MEME were put in their corresponding posi-

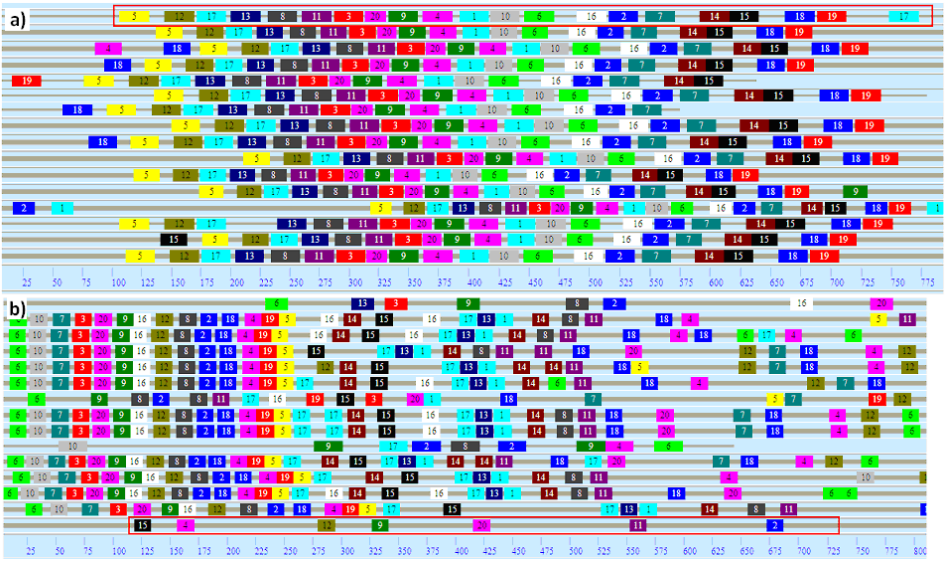


Figure 1: MEME general results (motif general arrangement) for a) the sequence aj810708.1, which serves as an example of MEME results, and b) for the sequence yscspo13. In the red rectangle, the original query sequence is shown; the other sequences, which were taken from its corresponding BLAST results, are represented below it. Every numbered color box represents a specific motif. MEME general motif arrangement results for the sequence yscspo13. In the red rectangle, the original query sequence is shown; the other sequences, which were taken from its corresponding BLAST results, are represented over it. Every numbered color box represents a specific motif.

tions. The strict process of selection followed in this analysis allowed primers to have 120 of degeneracy as maximium. For each primer pair, the accession number of the original query sequence was left as name.

PCR assays in vegetative and reproductive tissues. PCR reactions were assayed using the cDNA of pistils or vegetative tissue (leaves, roots and stems). The length of the pistils was chosen in basis of previous observations [AMP$^+$00, DW99]; in them, the embryo sac formation at this specific pistil size is reported. So, if the query sequences are specific of a process related to reproductive development, there is more chance to obtain them if we assay this specific part and moment of maturity.

Initially, we tried with 25 ng of starting cDNA as template, but neither of the primers showed positive results. It could imply that all amplified transcripts have low expression levels. So, we had to work with more concentrated DNA. A ten-dilution fold of the original cDNA gave amplification of 12 primer pairs in vegetative tissues (ab000809, af475105, aj810709, aj841698, penpsbca, penps31ab, ef517497, ef517498, ef530198, yscspo13, pcu40219, u65082) and only six in pistils (ab000809, aj810709, aj841698, ef517497, ef517498, yscspo13). As we expected always the presence of the product in reproductive tissue, we assayed again with a five-dilution fold. In this case, one primer (penps-

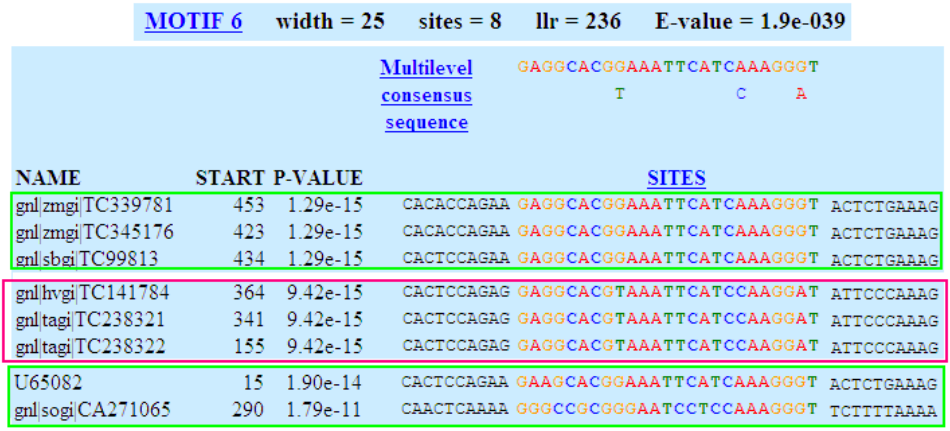


Figure 2: MEME alignment for motif #6 for the sequence u65082, as an example of alignments with taxonomic patterns. In green rectangles, the sequences of Panicoideae are shown; in the red rectangle there are sequences of Pooideae. Two letters followed by gi are employed to annotate the species associated with a particular sequence: zm = Zea mays; sb = Sorghum bicolor; so = Saccharum officinarum, hv = Hordeum vulgare; ta = Triticum aestivum. In this example, in the three degeneracies, the 9th base is always G in Panicoideae and T in Pooideae; the 19th is A in Panicoideae and C in Pooideae, and the 24th is G in Panicoideae and A in Pooideae; so, the consensus considered in this case don't include the T, A and A of Pooideae and the considered motif was GAGGCACGGAAATTCATCAAAGGGT.

bca) showed amplification in reproductive tissue and another primer pair in both samples (aj810708). Finally, we employed cDNA directly for the remaining ones, allowing positive results in pistils for the other five primer pairs that amplified only in vegetative tissue in 1:10 cDNA dilutions and another 2 primer pairs (aj786393, u65384) in both tissues. This may means that some sequences have greater expression levels in vegetative tissues that in reproductive ones. Seven primer pairs (aj810710, penpsbhc, ef530199, u65383, u65385, u65388 and u65389) never gave amplicons under neither of the assayed conditions.

Many of the obtained products had no more than 300bp in size (Figure 3). This is in concordance with the amplicon predicted by ePCR. In many cases, the amplicons obtained from vegetative and reproductive tissue were undistinguishable. However, 4 primer pairs (aj786393, aj810708, penpsbca and ef517498) gave products of different size according to the tissue assayed; the fragment in these situations was always of larger molecular weight in pistils. Events of specific tissue expression and differential splicing are plausible.

One very interesting result about these reactions is the fact that almost neither primer pair gave multiple bands, as could happen if the primer is highly degenerate. In most cases, a unique band was observed. The primer pair aj786393 is the only one that had multiple bands. But even that, a predominant band is observed, not only in the vegetative bulk but also in pistils. These bands could be obtained apart in the cloning tests (data not shown).

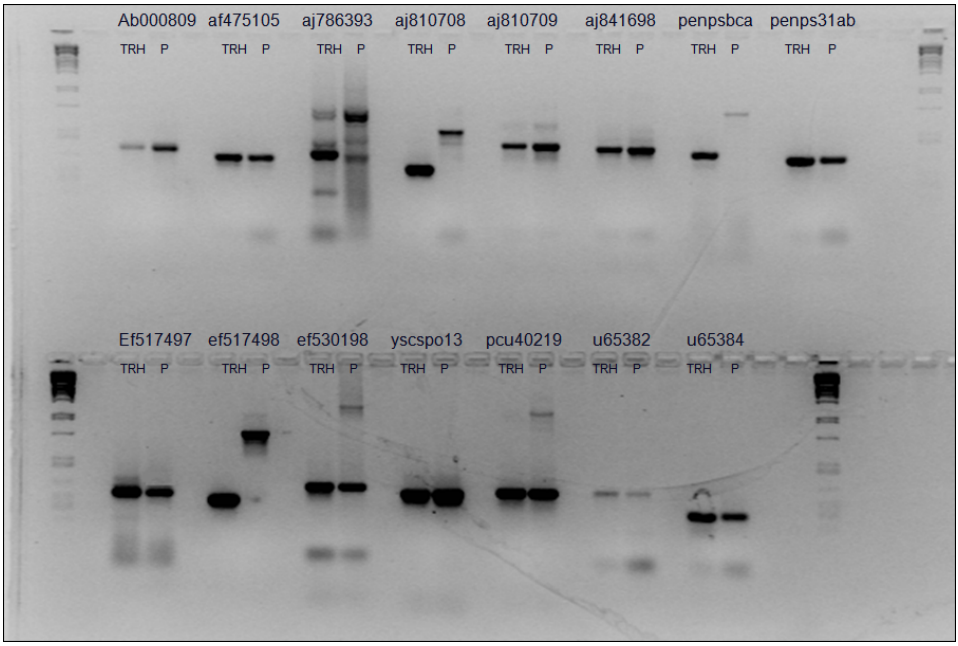


Figure 3: PCR products obtained for the 5 primer pairs that gave positive results. The name of each primer pair (equal to the accession number of the original query sequence) is shown above its corresponding result. The TRH title over one well indicates that the product there is from vegetative tissue, and a P, from pistils. Ladder: 1 kb DNA ladder (Invitrogen).

Bioinformatic analysis of sequenced amplicons. The PCR products were cloned and sequenced. Comparisons between the obtained sequences and the original query sequences using BLAST2SEQS and LALIGN showed high similarity including the entire acquired product and covering at least part of the query in all cases, except three. The query also appeared as heading result when BLAST search was made in these cases.

The three cases that did not show the expected similarity with queries were yscspo13, ab000809 and ef517498. The BLAST results of yscspo13 were overall sequences of the family of N-acetyltransferases, having similarities above 95% over the entire query. Ab000809 sequence matched with a group of functionally uncharacterized transcripts with high resemblance with threonin aldolase gene. Ef517498 is 88% similar to diverse transcripts of Oryza sativa related to the DMC1 gene in S. cerevisiae, a gene specific to meiotic events in that yeast.

Comparisons between the amplicons of different tissues were made. In the situations where both amplicons had the same molecular weight, the sequences were also the same. In the four cases when the PCR product showed different size, the sequence from vegetative tissue was always longer than from pistils. This supports the idea of differential splicing according to tissue, indicating a high spatial and, possibly, functional specificity.

4 DISCUSSION

In the present study, we developed a new method to design degenerate primers to target specific genes in Brachiaria. In contrast to earlier approaches, the method employed here does not require to start from large conserved blocks, extracted from multiple global alignments. Instead, it employs directly conserved motifs obtained from multiple local alignments. This allows us to work with poorly characterized sequences at the biological level and with few or neither homologues previously verified. The MEME algorithm [BWML06] was directly employed for this purpose. To the best of our knowledge, this is the first time that this program is used in this way. Also and for the first time, MEME was combined with ePCR (electronic PCR) [Sch97], which served as an initial filter for the motif pairs. On whole, the method developed in this study combines the use of bioinformatic programs which were not integrated before, and with a goal for which they were not originally made.

Our approach amplified finally and correctly 12 sequences (44% with respect to the 27 original sequences and 54% with respect to the 22 sequences for which primers could be made). These results are considered very positive, not only because the low knowledge about them, but also because many times we started from few similar sequences.

The three false positive results and, in general terms, the cases where an amplicon cannot be obtained, were associated to low similarity patterns in MEME analysis. As a consequence, the respective genetic family could not be detected or one family, different from the target group, was amplified, producing unspecific results. Another consequence of using these patterns is the obtention of high degeneracy levels (ej. penpsbhc, see Table 1). In some situations (u65082, u65383 and u65384), the small size of the fragments made difficult to find similar sequences, and this caused low similarity arrangements. Patterns like yscspo13 (Figure 1b), and in general where some conservation in the motif arrangement cannot be seen, should be avoided as much as possible, in order to increase the successfulness of the technique.

Excluding these cases, the method reported here showed good results. This is reflected in a low difference in the Tm of the primer pairs and between theoretical and experimental Tm, and in low degeneracy levels. The careful selection process of the degeneracies, which took into account taxonomic relationships, contributed to the favorable results obtained. It is desirable to look for ways to automatize and optimize this phase. Bioinformatic analysis could verify the amplification of the correct homologues. Thus, in general terms, the method has a good efficiency, which could be improved in further studies being more rigorous with the differences in Tm, the length of the sequences and the local and global similarity seen.

Qualitative differences in the expression of some of the evaluated transcripts could be observed, when sequences from somatic and reproductive tissues were compared. The loss of DNA segments could suggest specific differential splicing events. This suggests the existence of factors which alter mRNA in a tissue – specific manner, a possibility that should be deeply explored in Brachiaria. In events like this and in differential splicing, part of the explanation to apomixis could be found. So, the long sequence (from pistils)

could have a specific role in the reproductive development, and it must be analyzed in major detail.

In conclusion, we could develop a new bioinformatic method which allows us to amplify homologous genes associated to a very poorly characterized phenomenon at the molecular level like apomixis. The obtention of these sequences by this method is a very important step in order to stablish a clear and concrete molecular model of apospory, in this case in the plant genus Brachiaria. This method could help to make amplifications in other poorly understood biological events, in which there are few related sequences. In general terms, the technique can be used for sequences that have very few known homologues, or to confirm them, and to design degenerate primers when the classical methods do not work. The method, however, needs to be improved, for example with the automatization of some time consuming steps and the avoiding of patterns of low similarity. The results obtained here point to possible events related to differential splicing that could help to explain this very interesting trait. Our results will allow the analysis of all those proposed candidate genes in a unique plant species. In additional studies, the differences in expression and functional characteristics of the obtained sequences must be evaluated, like microarray analysis and real-time PCR. These sequences are being included in microarray analysis in our laboratory.

References

[ACA01] E. R. Alves, V. T. Carneiro, and A. C. Araujo. Direct evidence of pseudogamy in apomictic Brachiaria brizantha (Poaceae). *Sexual Plant Reproduction*, 14(4):207–212, 2001.

[AHOA05] Yukio Akiyama, Wayne W. Hanna, and Peggy Ozias-Akins. High-resolution physical mapping reveals that the apospory-specific genomic region (ASGR) in Cenchrus ciliaris is located on a heterochromatic and hemizygous region of a single chromosome. *Theoretical and Applied Genetics*, 111(6):1042–1051, 2005.

[AJ92] Sven Asker and Lenn Jerling. *Apomixis in Plants*. CRC Press Inc. Boca Raton, Florida, USA, 1 edition, 1992.

[AMB+04] Emidio Albertini, Gianpiero Marconi, Gianni Barcaccia, Lorenzo Raggi, and Mario Falcinelli. Isolation of candidate genes for apomixis in Poa pratensis L. *Plant Molecular Biology*, 56(6):879–894, 2004.

[AMP+00] A. C. Araújo, S. Mukhambetzhanov, M. T. Pozzobon, E. F. Santana, and V. T. Carneiro. Female gametophyte development in apomictic and sexual Brachiaria brizantha (Poaceae). *Rev. Cytol. Biol. Vég. Bot.*, 23:13–28, 2000.

[AMR+05] Emidio Albertini, Gianpiero Marconi, Lara Reale, Gianni Barcaccia, Andrea Porceddu, Francesco Ferranti, and Mario Falcinelli. SERK and APOSTART. Candidate Genes for Apomixis in Poa pratensis. *Plant Physiology*, 138(4):2185–2199, 2005.

[BBK00] R. A. Bicknell, N. K. Borst, and A. M. Koltunow. Monogenic inheritance of apomixis in two Hieracium species with distinct developmental mechanisms. *Heredity*, 84(2):228–237, 2000.

[BMG83] C. A. Bilinski, J. J. Miller, and S. C. Girvitz. Events associated with restoration by zinc of meiosis in apomictic Saccharomyces cerevisiae. *Journal of Bacteriology*, 155(3):1178–1184, 1983.

[BWML06] Timothy L. Bailey, Nadya Williams, Chris Misleh, and Wilfred W. Li. MEME: discovering and analyzing DNA and protein sequence motifs. *Nucleic Acids Research*, 34(suppl_2):W369–373, 2006.

[CCA+99] L. Chen, M. Chikara, K. Akio, S. Akira, and A. Taiji. Isolation and Characterization of a Gene Expressed during Early Embryo Sac Development in Apomictic Guinea Grass (Panicum maximum). *Journal of Plant Physiology*, 154:55–62, 1999.

[CGS+05] Lanzhuang Chen, Liming Guan, Misuk Seo, Franz Hoffmann, and Taiji Adachi. Developmental expression of ASG-1 during gametogenesis in apomictic guinea grass (Panicum maximum). *Journal of Plant Physiology*, 162(10):1141–1148, 2005.

[CHZ+07] Kevin L. Childs, John P. Hamilton, Wei Zhu, Eugene Ly, Foo Cheung, Hank Wu, Pablo D. Rabinowicz, Chris D. Town, C. Robin Buell, and Agnes P. Chan. The TIGR Plant Transcript Assemblies database. *Nucleic Acids Research*, 35(suppl_1):D846–851, 2007.

[dVSJ89] C. B. do Valle, Y. H. Savidan, and L. Jank. Apomixis and sexuality in Brachiaria decumbens Stapf. In *Proceedings of the XVI International Grassland Congress*, pages 407–408, 1989.

[DW99] D. M. A. Dusi and M. T. M. Willemse. Apomixis in Brachiaria decumbens Stapf.: Gametophytic Development and Reproductive Calendar. *Acta Biologica Cracoviensia Series Botanica*, 41:151–162, 1999.

[DWCB01] Zhao-Jun Ding, T. Wang, Kang Chong, and Shunong Bai. Isolation and characterization of OsDMC1 , the rice homologue of the yeast DMC1 gene essential for meiosis. *Sexual Plant Reproduction*, 13(5):285–288, 2001.

[GDY+05] P. J. Gulick, S. Drouin, Z. Yu, J. Danyluk, G. Poisson, A. F. Monroy, and F. Sarhan. Transcriptome comparison of winter and spring wheat responding to low temperature. *Genome*, 48(5):913–23, 2005.

[GMDK05] Michael D. Gadberry, Simon T. Malcomber, Andrew N. Doust, and Elizabeth A. Kellogg. Primaclade–a flexible tool to find conserved PCR primers across multiple species. *Bioinformatics*, 21(7):1263–1264, 2005.

[GNvD01] Ueli Grossniklaus, Gian A. Nogler, and Peter J. van Dijk. How to Avoid Sex: The Genetic Control of Gametophytic Apomixis. *Plant Cell*, 13(7):1491–1498, 2001.

[GRR+00] Jenny Guerin, Jan Bart Rossel, Stanley Robert, Tohru Tsuchiya, and Anna Koltunow. A DEFICIENS homologue is down-regulated during apomictic initiation in ovules of Hieracium. *Planta*, 210(6):914–920, 2000.

[GSH93] D. L. Gustine, R. T. Sherwood, and D. A. Hulce. A strategy for cloning apomixis-associated cDNA markers from buffelgrass. In *Proceedings of the XVII International Grassland Congress*, pages 1033–1034, 1993.

[KCSW94] S. Kwok, S. Y. Chang, J. J. Sninsky, and A. Wang. A guide to the design and use of mismatched and degenerate primers. *Genome Research*, 3(4):S39–S47, 1994.

[KG03] Anna M. Koltunow and Ueli Grossniklaus. APOMIXIS: A Developmental Perspective. *Annual Review of Plant Biology*, 54(1):547–574, 2003.

[LAP+97] O. Leblanc, I. Armstead, S. Pessino, J. P. A. Ortiz, C. Evans, C. do Valle, and M. D. Hayward. Non-radioactive mRNA fingerprinting to visualise gene expression in mature ovaries of Brachiaria hybrids derived from B. brizantha, an apomictic tropical forage. *Plant Science*, 126(1):49–58, 1997.

[LBC+02] P. Labombarda, A. Busti, M. E. Caceres, F. Pupilli, and S. Arcioni. An AFLP marker tightly linked to apomixis reveals hemizygosity in a portion of the apomixis-controlling locus in Paspalum simplex. *Genome*, 45(3):513–519, 2002.

[Mat89] F. Matzk. Genetic studies on parthenogenesis in Poa pratensis L. *Apomixis Newsletter*, 1:32–34, 1989.

[ME90] M. J. Malavasic and R. T. Elder. Complementary transcripts from two genes necessary for normal meiosis in the yeast Saccharomyces cerevisiae. *Molecular and Cellular Biology*, 10(6):2809–2819, 1990.

[OARH98] Peggy Ozias-Akins, Dominique Roche, and Wayne W. Hanna. Tight clustering and hemizygosity of apomixis-linked molecular markers in Pennisetum squamulatum implies genetic control of apospory by a divergent locus that may have no allelic form in sexual genotypes. *Proceedings of the National Academy of Sciences of the United States of America*, 95(9):5127–5132, 1998.

[OCJ+07] Takashi Okada, Andrew S. Catanach, Susan D. Johnson, Ross A. Bicknell, and Anna M. Koltunow. An Hieracium mutant, loss of apomeiosis 1 (loa1) is defective in the initiation of apomixis. *Sexual Plant Reproduction*, 20(4):199–211, 2007.

[OPL+97] J. P. A. Ortíz, S. C. Pessino, O. Leblanc, M. D. Hayward, and C. L. Quarín. Genetic fingerprinting for determining the mode of reproduction in Paspalum notatum, a subtropical apomictic forage grass. *Theoretical and Applied Genetics*, 95(5-6):850–856, 1997.

[PEM+01] Silvina C. Pessino, Francisco Espinoza, Eric J. Martínez, Juan Pablo A. Ortiz, Estela M. Valle, and Camilo L. Quarín. Isolation of cDNA Clones Differentially Expressed in Flowers of Apomictic and Sexual Paspalum Notatum. *Hereditas*, 134(1):35–42, 2001.

[PEO+98] Silvina C. Pessino, Clive Evans, Juan Pablo A. Ortiz, Ian Armstead, Cacilda B. do Valle, and Michael D. Hayward. A Genetic Map of the Apospory-Region in ¡i¿Brachiaria¡/i¿ Hybrids: Identification of two Markers Closely Associated with the Trait. *Hereditas*, 128(2):153–158, 1998.

[PMB+04] F. Pupilli, E. J. Martinez, A. Busti, O. Calderini, C. L. Quarin, and S. Arcioni. Comparative mapping reveals partial conservation of synteny at the apomixis locus in Paspalum spp. *Molecular Genetics and Genomics*, 270(6):539–548, 2004.

[PNT+99] Bogdan Polevoda, Joakim Norbeck, Hikaru Takakura, Anders Blomberg, and Fred Sherman. Identification and specificities of N-terminal acetyltransferases from Saccharomyces cerevisiae. *The EMBO Journal*, 18(21):6155–6168, 1999.

[POL+97] S. C. Pessino, J. P. A. Ortiz, O. Leblanc, C. B. do Valle, C. Evans, and M. D. Hayward. Identification of a maize linkage group related to apomixis in Brachiaria. *Theoretical and Applied Genetics*, 94(3-4):439–444, 1997.

[QLH+00] John Quackenbush, Feng Liang, Ingeborg Holt, Geo Pertea, and Jonathan Upton. The TIGR Gene Indices: reconstruction and representation of expressed gene sequences. *Nucleic Acids Research*, 28(1):141–145, 2000.

[RCB+02] D. Roche, A. Conner, A. Budiman, D. Frisch, R. Wing, W. Hanna, and P. Ozias-Akins. Construction of BAC libraries from two apomictic grasses to study the microcolinearity of their apospory-specific genomic regions. *Theoretical and Applied Genetics*, 104(5):804–812, 2002.

[RCD+03] Júlio C. M. Rodrigues, Gláucia B. Cabral, Diva M. A. Dusi, Luciane V. de Mello, Daniel J. Rigden, and Vera T. C. Carneiro. Identification of differentially expressed cDNA sequences in ovaries of sexual and apomictic plants of Brachiaria brizantha. *Plant Molecular Biology*, 53(6):745–757, 2003.

[SBF07] Manjit Singh, Byron L. Burson, and Scott A. Finlayson. Isolation of candidate genes for apomictic development in buffelgrass (Pennisetum ciliare). *Plant Molecular Biology*, 64(6):673–682, 2007.

[Sch97] Gregory D. Schuler. Sequence Mapping by Electronic PCR. *Genome Research*, 7(5):541–550, 1997.

[VCNB+96] J.-Philippe Vielle-Calzada, Michael L. Nuccio, Muhammad A. Budiman, Terry L. Thomas, Byron L. Burson, Mark A. Hussey, and Rod A. Wing. Comparative gene expression in sexual and apomictic ovaries of Pennisetum ciliare (L.) Link. *Plant Molecular Biology*, 32(6):1085–1092, 1996.

Accession number (GenBank)	Theoretical amplicon length (bp)[a]	Sequence Primer Fwd (5'-3')	Tm Fwd (ºC)	Degeneration Fwd[b]	Sequence Primer Rev (5'-3')	Tm Rev (ºC)	Degeneration Rev	Total degeneration[c]
AB000809	380	TGCGCSACGTCGCTGGAG	63.7	2	CATACGGCAKGARGTGGCAG	58.5	4	8
AF475105	371	GAAGTACTGGAAYGACCCTG	53.7	2	GGGGTGTTCTTGTTCTTRTC	52.2	2	4
AJ786393	380	GTTGGACAGTACCTGAYAGCA	55.5	2	CGCACTATCCATGAKCCCT	55.6	2	4
AJ810708	301	CAAATYTGGGAYACTGCTGG	53.9	4	TTCTTKGCACTGGTCTCCATG	56.3	2	8
AJ810709	431	GGCATGTGCTGCTTCTTCTACAT	57.7	1	GCTTKCTCGACYGTCAAGCT	57.8	4	4
AJ810710	326	TACACAAGGTATTCAGACAGAGTGA	55.2	1	ATRATTTCCTTGGTCCGC	51.3	2	2
AJ841698	407	ATGGGTTTGGTATTCTCCTGTTGGA	58.1	1	GGACCAGATAGCTCAATTGGCTC	57.5	1	1
D37938 (penpsbca)	416	AYACATTTGCATATCACACATACTT	52.1	2	TCTTCRCCAGGYTTCAGAG	54.3	4	8
D37939 (penpsb31ab)	375	TGAAGTAYTGGAATGACCCTG	53.1	2	CARYGGGRTGTTCTTGTTCT	54	4	8
D37940 (penpsbhc)	214	AATGTTCAGTTACACYNGNCTATGC	58	32	ATTGARRATTTTCAGGGAGGA	51.5	4	128
EF517497	423	GGCAAAWTTAGAARAWGAGG	48	8	ACTTRCTYRGTTTTACATACCATAA	51.3	8	64
EF517498	256	GAATTGGGTCCACNATTKCTC	54.8	8	TCGTTCAGCATGTCTWGGTAGAACT	57.6	2	16
EF530198	330	GAYGTGAAGAAGCTGMARGAT	53.9	8	AAGCTGAGTKGARACACAWAGAGT	56.2	8	64
EF530199	336	ATTCCATAACGCKCCGG	53.9	2	GCASTGATGTAGTTTGTATCTTTMA	52.3	4	8
M38357 (yscspo13)	305	ATGCAGGCGTGCAACCTGAT	60.3	1	GGACGTGGAGGGASACGTACT	60.5	8	8
U40219 (pcu40219)	342	GATTTCCTCACCAAGGAGGT	54.3	1	CATGCYTTYGCCTTYTTCT	52.3	2	2
U65082	355	GARGCACGKAAATTCATCAAA	51.5	4	GGATTCCWTGATSAGCTTCTCC	55.3	4	16
U65383	248	CRRCGCTACATTGATGATGATG	54.1	4	CATCAGAGRARGTAAGCSATCA	54.2	8	32
U65384	205	ATTCGGCTTYGGTCSAGA	55.1	4	TTCAACAGCTTTCTCSACACAT	54.8	2	8
U65385	243	TTGCMAGCTTCCTGCTGCTCTT	60.8	2	CACYGCRGCTCCAGCTGG	61.9	4	8
U65388	246	ATAYGAKACTCTGCATTGATAA	48.9	4	GCTATGTAATGTACTGCTACTACTA	51.5	1	4
U65389	261	ATGCAGGCGTGCAACCTGAT	63.1	1	AGCTCAAGCCTTGAGCTTGCCGTAG	60.3	1	1

Table 2: Sequences obtained for the degenerate primers in this work, theoretical amplicon, Tm and level of degeneration. a) Average length calculated by ePCR in the group of sequences with which alignments were made. b) The degeneracy of every primer is determinated according to the existent degeneracies. Every two-base degeneracy (S, Y, W, R, K, M) sums 2 to total degeneracy. Every three-base degeneracy (B, D, H, V) sums 3 to total degeneracy. Each N sums 4 to total degeneracy. Total primer degeneracy is calculated multiplying the contribution of each base position. c) Combined degeneracy of the primer set, calculated by multiplying the degeneracy of forward and reverse primers.

Discovering temporal patterns of differential gene expression in microarray time series

Oliver Stegle[1], Katherine J. Denby[2], Stuart McHattie[2], Andrew Mead[2],
David L. Wild[2], Zoubin Ghahramani[3], Karsten M. Borgwardt[1]
[1]Max Planck Institutes Tübingen, [2]University of Warwick, [3]University of Cambridge

Abstract: A wealth of time series of microarray measurements have become available over recent years. Several two-sample tests for detecting differential gene expression in these time series have been defined, but they can only answer the question *whether* a gene is differentially expressed across the whole time series, not *in which intervals* it is differentially expressed. In this article, we propose a Gaussian process based approach for studying these dynamics of differential gene expression. In experiments on *Arabidopsis thaliana* gene expression levels, our novel technique helps us to uncover that the family of WRKY transcription factors appears to be involved in the early response to infection by a fungal pathogen.

1 Introduction

Microarray data are a major resource for studying the response of an organism to external conditions and stimuli. In the past, the majority of studies considered only a single measurement in each condition. Recent advances in microarray technology and falling costs have led to an increasing number of studies where expression levels are measured in different conditions over time rather than in a single snapshot.

A range of techniques to test for differential expression have been proposed in the computational biology and statistics communities. In statistics, this task is often referred to as the two-sample problem. The majority of these existing methods are aimed at identifying differentially expressed genes from static microarray experiments, for example (KMC00, DYCS02, ETST01).

More recent approaches are specifically designed for time series (JGS$^+$03; SXL$^+$05; TS06; CDCMP07; ACC$^+$08), and a range of desired properties of a two-sample test for microarray time series have been established. First, the test should explicitly address the dependencies between consecutive measurements. Second, the method should not make overly strong assumptions about functions describing the time series, such as assuming a linear or finite model basis (Yua06). Third, to accommodate data characteristics specific to the microarray platform, it is beneficial to handle missing values and deal with multiple replicates. Finally, robustness with respect to outliers has proven useful for reliable results on microarray datasets (CDCMP07; ACC$^+$08).

To address all of these issues, we defined a robust Bayesian two-sample test for differential gene expression using Gaussian processes (GP) in (SDW$^+$09). In addition to solving the basic two-sample problem, the presented method can also be used to decide whether

differential expression occurs at a specific time point in the time series.

However, the test from (SDW$^+$09) does not reflect'smoothness' between decisions at consecutive time points. That is, there can be abrupt switches from non-differential gene expression to differential gene expression (and vice versa) from one time point to the next. If one wants to detect meaningful temporal *intervals* of differential gene expression rather than individual time steps, it is vital to incorporate this smoothness assumption into the formulation of the statistical model. This is exactly the goal of this article.

The remainder of this article is organised as follows. We start by reviewing how Gaussian processes can be applied to test for differential expression in microarray time series (SDW$^+$09). In Section 3, this basic test is extended to a temporal model detecting intervals of differential gene expression. Finally, in Section 4, we demonstrate how this additional information can be useful to gain insights into regulatory mechanisms involved in the response of *Arabidopsis* to an infection by a fungal pathogen.

2 Gaussian Process-based two-sample test

The task of detecting differential gene expression is defined as follows: Given observed gene expression levels from two biological replicates that are exposed to different conditions, the goal is to determine whether a given gene probe is differentially expressed in these conditions or not.

The principle underlying the Gaussian process-based two-sample test (GPTwoSample) from (SDW$^+$09) is the comparison of two models: The first model assumes that the microarray time series in both conditions are samples drawn from an identical *shared* distribution. An alternative model describes the time series in both conditions as samples from two *independent* distributions. As these distributions need to be defined over functions, a Gaussian process is an appealing model. A GP incorporates beliefs about smoothness and allows all model parameters except for a handful of hyperparameters to be integrated out analytically, allowing for tractable model comparison. The two alternatives, the *shared* model ($\mathcal{H}_\mathrm{S}$) and the *independent* model ($\mathcal{H}_\mathrm{I}$) can then be objectively compared using the logarithm of the Bayes factor

$$\text{Score} = \log \frac{P(\mathcal{D}_A, \mathcal{D}_B \,|\, \mathcal{H}_\mathrm{I})}{P(\mathcal{D}_A, \mathcal{D}_B \,|\, \mathcal{H}_\mathrm{S})}, \tag{1}$$

where $\mathcal{D}_A$ and $\mathcal{D}_B$ are observed expression levels in two conditions A and B. Writing out the GP models explicitly leads to

$$\text{Score} = \log \frac{P(\mathbf{Y}^A \,|\, \mathcal{H}_\mathrm{GP}, \mathbf{T}^A, \boldsymbol{\theta}_\mathrm{I}) P(\mathbf{Y}^B \,|\, \mathcal{H}_\mathrm{GP}, \mathbf{T}^B, \boldsymbol{\theta}_\mathrm{I})}{P(\mathbf{Y}^A \cup \mathbf{Y}^B \,|\, \mathcal{H}_\mathrm{GP}, \mathbf{T}^A \cup \mathbf{T}^B, \boldsymbol{\theta}_\mathrm{S})}, \tag{2}$$

where $Y^{A/B}$ are observed expression levels and $T^{A/B}$ are the corresponding time points in both conditions and $\boldsymbol{\theta}_\mathrm{I}, \boldsymbol{\theta}_\mathrm{S}$ are hyperparameters of both models. For details see (SDW$^+$09).

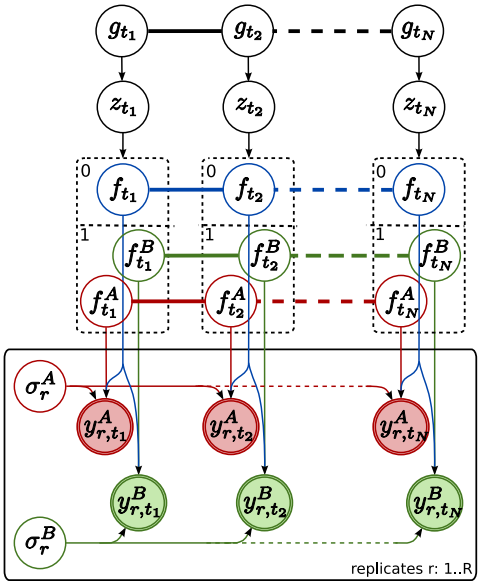


Figure 1: Bayesian network for the temporal GPTwoSample model. At observed time points $\{t_n\}$, binary indicator variables $\{z_{t_n}\}$ determine the state of a gate and hence which expert explains the corresponding observations. If the state of the indicator is 1 the *independent* expert is used, while if the switch is 0, the *shared* expert is used. The *shared* expert uses a single GP $f(t)$ to model both conditions. The *independent* expert uses two GPs $f^A(t), f^B(t)$. Smoothness of the GP priors is indicated by the thick bands coupling function values at different time points. A logistic Gaussian process, $g(t)$, incorporates smoothness over the state of the indicator variables.

3 Detecting intervals of differential gene expression

Once we know that a particular gene is differentially expressed, it is interesting to ask in which intervals of the time series this effect is present. Such detailed analysis is particularly valuable for longer time series, where differential behaviour might be present only temporarily or occur after a certain time delay.

To address this question, we propose a mixture model over time, where one mixture component (expert) corresponds to the *shared* model and a second mixture component to the *independent* model. A Bayesian network representation of this temporal GPTwoSample model is shown in Figure 1. The model is closely related to mixtures of Gaussian process experts (RG01). The *shared* expert is a single GP explaining expression levels in both conditions, while the *independent* expert uses a separate GP for each condition.

At observed time points binary switches $\mathbf{z} = \{z_{t_1}, \ldots, z_{t_n}\}$ determine which expert explains the corresponding expression levels. In the same way that the expression levels vary smoothly over time, we also believe that the states of the indicators follow a smooth trend, typically reflecting a transition from the *shared* expert to the *independent* expert. This belief about smoothness is expressed in a gating network, implying a joint probability

distribution over all indicators $P(\mathbf{z} \mid \mathbf{T}, \boldsymbol{\theta}_{\mathrm{G}})$.

The coupling of the observed expression levels $\mathbf{Y}$ by the GP experts renders inference in this mixture model difficult. While the latent function values $\mathbf{f}$ can be integrated out in closed form, marginalising over the state of the indicators $\mathbf{z}$ yields an exponential sum over all possible configurations:

$$P(\mathbf{Y} \mid \mathbf{T}, \boldsymbol{\theta}_{\mathrm{S}}, \boldsymbol{\theta}_{\mathrm{I}}, \boldsymbol{\theta}_{\mathrm{G}}) = \sum_{\mathbf{z}} P(\mathbf{z} \mid \boldsymbol{\theta}_{\mathrm{G}}) \big[P(\mathbf{Y}_{\{y_n : z_{t_n}=0\}} \mid \mathcal{H}_{\mathrm{S}}, \mathbf{T}_{\{t_n : z_{t_n}=0\}}, \boldsymbol{\theta}_{\mathrm{S}}) \times P(\mathbf{Y}_{\{y_n : z_{t_n}=1\}} \mid \mathcal{H}_{\mathrm{I}}, \mathbf{T}_{\{t_n : z_{t_n}=1\}}, \boldsymbol{\theta}_{\mathrm{I}}) \big]. \quad (3)$$

The two terms in the sum are the data likelihoods from both GP experts introduced in Section 2. Here we follow (RG01) and exploit tractable conditional distributions. Conditioned on a particular configuration of the indicators $\mathbf{z}$, the likelihood factorises into a product over the two experts, where the data are split between the experts according to the state of the indicator variables.

A Gibbs sampler is well suited for this inference task. The latent function values of the experts can be integrated out or collapsed and hence Gibbs sampling steps reduce to updates of one indicator at a time, conditioning on the current state of all remaining indicators and data. The conditional distribution over a particular indicator z_{t_i} is

$$P\left(z_{t_i} = s \mid \mathbf{z}^{\setminus t_i}, \mathbf{T}, \mathbf{Y}, \boldsymbol{\theta}_{\mathrm{S}}, \boldsymbol{\theta}_{\mathrm{I}}, \boldsymbol{\theta}_{\mathrm{G}}\right) \propto P\left(\mathbf{Y} \mid z_{t_i} = s, \mathbf{z}^{\setminus t_i}, \mathbf{T}, \boldsymbol{\theta}_{\mathrm{I}}, \boldsymbol{\theta}_{\mathrm{S}}\right) \times P\left(z_{t_i} = s \mid \mathbf{z}^{\setminus t_i}, \boldsymbol{\theta}_{\mathrm{G}}\right), \quad (4)$$

with $s \in \{0, 1\}$. The first term is the conditional data likelihood. Rewriting this term as

$$P\left(\mathbf{Y} \mid z_{t_i} = s, \mathbf{z}^{\setminus t_i}, \mathbf{T}, \boldsymbol{\theta}_{\mathrm{I}}, \boldsymbol{\theta}_{\mathrm{S}}\right) = P(y_{t_i} \mid z_{t_i} = s, \mathbf{z}^{\setminus t_i}, \mathbf{Y}^{\setminus y_{t_i}}, \mathbf{T}) \times P(\mathbf{Y}^{\setminus y_i} \mid \mathbf{z}^{\setminus t_i}, \mathbf{T}^{\setminus t_i}) \quad (5)$$

reveals that for Gibbs sampling it is sufficient to calculate the probability of y_{t_i} under the leave-one-out predictive distribution of both GP experts.

The second term in (4) is the probability of the indicator z_{t_i} under the predictive distribution of the gating network, given all other indicators. We choose a logistic Gaussian process as a gating network, where smoothness is expressed by a GP prior on a latent function $g(t)$ (Figure 1). Bernoulli predictions of an indicator z_{t_i} are related to the Gaussian predictive function values by a probit likelihood model (full details in accompanying technical report),

$$P(z_{t_i} = 1 \mid \mathbf{z}^{\setminus t_i}, \mathbf{T}) = \int_{g_{t_i}} \Phi(g_{t_i}) \mathcal{N}\left(g_{t_i} \,\middle|\, \mu_{t_i}, \sigma_{t_i}^2\right) \mathrm{d}g_{t_i}. \quad (6)$$

The likelihood models of both Gaussian process experts as well as the gating network are all non-Gaussian and hence predictive distributions are not available in closed form. Expectation Propagation (EP) (see tech report, (SDW$^+$09)) is applied to all these cases to obtain tractable approximate predictive densities.

Sampling of the indicators is repeated for a number of randomised sweeps through all indicators. After every full sweep, the GP hyperparameters from the GP experts and the gating function are sampled using Hamiltonian Monte Carlo (e.g. (Mac03)). The complete sampling scheme is summarized in Algorithm 1.

Algorithm 1 Sampling scheme for the temporal GPTwoSample model

for $n_g = 1 \dots N_g$ Gibbs sweeps **do**
  for $n \in 1, \dots, N$ measurements **do**
    Resample indicator z_{t_n} (Equation (4)).
  end for
  Sample the hyperparameters $\boldsymbol{\theta}_{\mathrm{S}}, \boldsymbol{\theta}_{\mathrm{I}}$ and $\boldsymbol{\theta}_{\mathrm{G}}$ conditioned on $\mathbf{z}$.
end for

To identify temporal patterns of differential expression, we are most interested in the inferred states of the indicators. After a burn-in period, the generated samples yield an empirical posterior distribution over the indicator variables $\mathbf{z}$. Predictions of the gating network at test times $t_\star$ can be obtained by integrating out $\mathbf{z}$ using a set of S samples

$$P(z_\star = 1 \,|\, \mathbf{Y}, \mathbf{T}, t_\star) \approx \frac{1}{S} \sum_{s=1}^{S} P(z_\star = 1 \,|\, \mathbf{Y}, \mathbf{T}, t_\star, \mathbf{z}^{(s)}, \boldsymbol{\theta}_{\mathrm{G}}^{(s)}), \tag{7}$$

yielding a mixture of Bernoulli distributions. These marginal predictions ignore the coupling at different time points that is introduced from the sampled states $\{\mathbf{z}^{(s)}\}$. However, after a sufficient burn-in period, most of the indicators are constant across samples $\{\mathbf{z}^{(s)}\}$ and hence marginal predictions are appropriate. The same argument applies to predictions of the latent function values of the GP experts. These mixtures of Gaussians are well approximated by a Gaussian predictive distribution.

4 Detecting transition points in *Arabidopsis* microarray time series

We applied the temporal GPTwoSample model[1] to detect intervals of differential expression of gene probes from an *Arabidopsis* time series dataset.

In this particular experiment, the stress response of interest is an infection of *Arabidopsis thaliana* by the fungal pathogen *Botrytis cinerea*. The ultimate goal is to elucidate the gene regulatory networks controlling the plant defense against this pathogen. The identification of intervals of differentially expressed genes is an important first step towards this goal.

Data were obtained from an experiment in which detached *Arabidopsis* leaves were inoculated with a *B. cinerea* spore suspension (or mock-inoculated) and harvested every 2 hr up to 48 hr post-inoculation for a total of 24 time points. *B. cinerea* spores (suspended in half-strength grape juice) germinate, penetrate the leaf and cause expanding necrotic lesions. Mock-inoculated leaves were treated with droplets of half-strength grape juice. At each time point and for both treatments, one leaf was harvested from four plants under identical

[1] Software will be made available with the accompanying tech report.

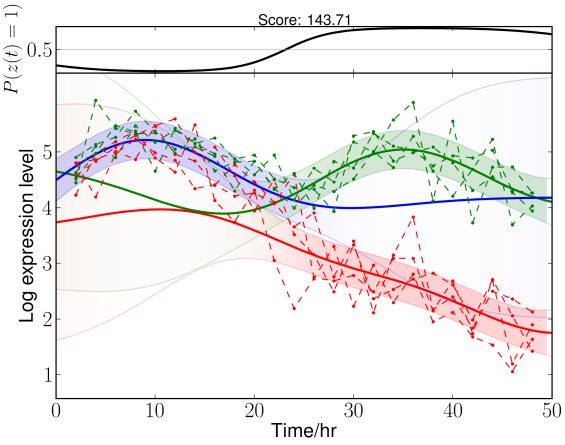


Figure 2: An example result produced by the GPTwoSample temporal test. **Top:** The posterior probability of differential expression as a function of time. **Bottom:** Dashed lines represent replicates of gene expression measurements for control (green) and treatment (red). Thick solid lines are Gaussian process mean predictions of the latent process traces; error bars of plus or minus 2 standard deviations are indicated by shaded areas. The intensity of the shaded areas is modulated by the posterior probability of the respective Gaussian process expert. The score in the figure title is the Bayes factor of the standard GPTwoSample test.

conditions (i.e. there were 4 biological replicates). Full genome expression profiles were generated from these whole leaves covering a total of 30,336 gene probes.

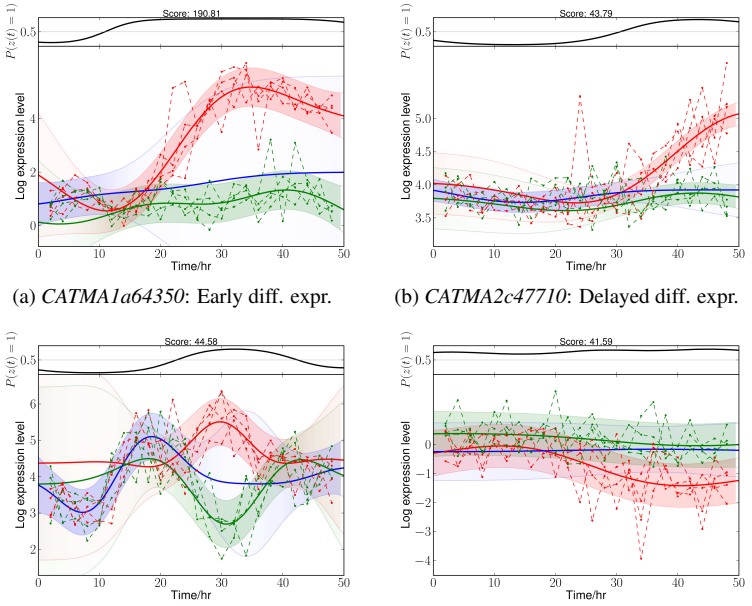
In the experiments, we used our novel test for detecting intervals of differential gene expression for each of the 30,336 probes. In the computations, a total of 50 Gibbs sweeps were performed. After every Gibbs sweep 5 Hamiltonian Monte Carlo updates were interleaved. To allow for a burn-in period, posterior parameters were estimated from samples of the last 25 sweeps.

Figure 2 gives an example result of the temporal GPTwoSample model. The top panel shows the marginal predictive distribution for the indicator state $z(t)$, choosing between the *shared*, $z(t) = 0$, and the *independent*, $z(t) = 1$, expert. The bottom panel shows the raw data and marginal predictions of latent function values from both GP experts. For this particular gene the test identified intervals of clear differential expression that started at around 22 hr post inoculation and lasted until the end of the time series recording.

Additional results for a representative selection of gene probes are shown in Figure 3.

Delayed differential expression

Applying the temporal GPTwoSample test to a large set of differentially expressed genes, it is possible to study the distribution of their start and stop times of differential expression.



(a) *CATMA1a64350*: Early diff. expr.

(b) *CATMA2c47710*: Delayed diff. expr.

(c) *CATMA1c71184*: Transient diff. expr.

(d) *CATMA5a14990*: Permanent diff. expr.

Figure 3: Example results of the temporal GPTwoSample model applied to the *Arabidopsis* data. Panels **(a)** and **(b)** show examples of particularly early and late differential expression. In **(c)** a gene probe is shown for which differential expression appeared to be transient. Example **(d)** shows a probe with weak evidence for differential expression throughout the time series.

For this analysis, the top 6000 genes that had a score suggesting significant differential expression were used. For each gene the start time of differential expression was determined as the first time point at which the posterior probability of differential expression, $P(z_{t_n} = 1)$, exceeded 0.5, evaluated at a discretisation of 100 points in the interval $[0, 50]$ hr. Analogously the stop time was deduced as the time point where differential expression ended, i.e. $P(z_{t_n} = 0) \leq 0.5$. The lower panel in Figure 4 shows the histogram of the start time for the considered 6000 gene probes. The identification of transition points for individual gene expression profiles shows that a significant change in the transcriptional program began at around 17 hr post-inoculation. This program of gene expression change appeared to have two strong waves peaking around 21 hr and 25 hr. For a small fraction of genes this change in the transcriptional program started at either significantly earlier or later times; Figures 3a and 3b give examples of such genes. Figure 3d shows results for one of the approximately 200 genes that were identified as differentially expressed right from the start of the time series. Most of these genes were weakly expressed and an offset between the measurements in both conditions triggered the early classification as differentially expressed.

The top panel of Figure 4 shows the stop time of differential expression for 13 genes for

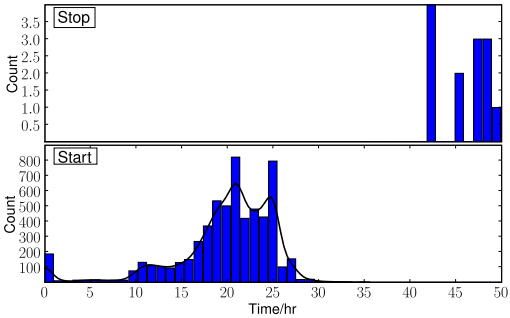


Figure 4: Histogram of the most likely start and stop of differential expression for the top 6000 differentially expressed genes. Stop time are shown for a total of 13 genes that appear to exhibit transient differential expression ending within the observed time window.

which the differential expression program ended within the measured time interval. An example of one of these genes with transient differential expression is given in Figure 3c.

Interpreting waves of differential expression

It is interesting to understand the causes for the different onset-timings of differential expression for individual genes. We expect regulators (if involved in the response to the fungus infection) to be expressed at earlier times than the downstream genes they control. In the *Arabidopsis* response to stress, several relevant regulatory mechanisms have been established in the literature. These include transcription factors (CPG$^+$02; SYS00) as well as kinases (FFN$^+$06; CRBX05).

Figure 5 shows histograms of the start time of differential expression for groupings of the 6000 genes that correspond to different gene categories. Tentatively, transcription factors and kinases appeared to be stronger represented in the earlier wave; however application of a Kolmogorov-Smirnov (KS) test revealed that these differences were not significant (transcription factors: $p = 0.092$, kinases: $p = 0.964$).

The differential expression onset-timing can be broken down further, for instance into subfamilies of transcription factors. The family of WRKY transcription factors is known to play a role in response to biotic stresses (CPG$^+$02). The onset times of transcription factors in this family appeared to be overrepresented in early differential expression compared to other transcription factors. A KS-test revealed that this subset of 26 transcription factors exhibited a significantly different distribution of onset-times than other genes ($p = 3.3 \cdot 10^{-6}$). This results demonstrates the usefulness of the time-local two-sample test. By analysing the onset timing it is possible to narrow down the set of interesting candidate genes to study. When designing further experiments to elucidate transcriptional networks mediating the defense response against *B. cinerea*, regulatory genes whose expression first changes in the 21 hr wave or earlier would be of particular interest.

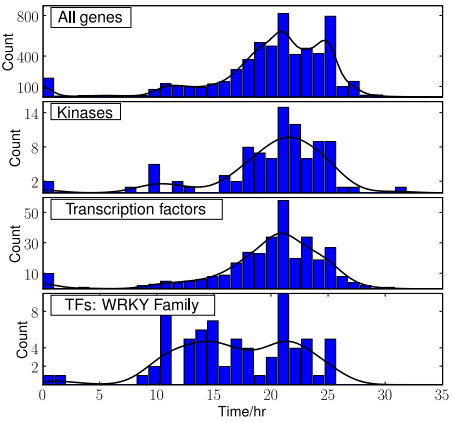


Figure 5: Histogram of the most likely start differential expression for the top 6000 differentially expressed genes split up into different gene categories. From top to bottom the histograms show results for all 6000 genes, kinases , known and putative transcription factors and WRKY transcription factors.

5 Discussion and outlook

The temporal GPTwoSample model, which we presented in this article, extends the standard paradigm of the two-sample problem and our previous work (SDW$^+$09) to the identification of smooth intervals of differential expression. The proposed method is computationally efficient and can be applied to large datasets with thousands of genes using a standard desktop PC. Experimental results on 6000 differentially expressed *Arabidopsis thaliana* gene probes revealed patterns in the timing of the response to a fungal infection (Figure 3). As an example application we studied the distribution of the start and stop times of differential expression (Figure 4) that led to insights on waves of differential expression in *Arabidopsis* genes (Figure 5).

Several extensions of the method developed in this article would be of interest. First, the current model does not distinguish between different expression patterns and anti correlated genes. Explicit modeling of anti-correlation is an important next step. Second, extensions to model differential gene expression at a network view rather than at the level of individual genes are an interesting direction of future development of differential gene expression models. The presented method provides the required per-gene level model for such future investigation.

References

[ACC$^+$08] C. Angelini, L. Cutillo, D. Canditiis, M. Mutarelli, and M. Pensky. BATS: a Bayesian user-friendly software for Analyzing Time Series microarray experiments. *BMC Bioinformatics*, 9(1):415, 2008.

[CDCMP07] Angelini C., D. De Canditiis, M. Mutarelli, and M. Pensky. A Bayesian Approach to

Estimation and Testing in Time-course Microarray Experiments. *Statistical Applications in Genetics and Molecular Biology*, 6, September 2007.

[CPG+02] W. Chen, N.J. Provart, J. Glazebrook, F. Katagiri, H.S. Chang, T. Eulgem, F. Mauch, S. Luan, G. Zou, S.A. Whitham, et al. Expression profile matrix of Arabidopsis transcription factor genes suggests their putative functions in response to environmental stresses. *The Plant Cell Online*, 14(3):559, 2002.

[CRBX05] M. Cvetkovska, C. Rampitsch, N. Bykova, and T. Xing. Genomic analysis of MAP kinase cascades in Arabidopsis defense responses. *Plant Molecular Biology Reporter*, 23(4):331–343, 2005.

[DYCS02] S. Dudoit, Y. H. Yang, M. J. Callow, and T. P. Speed. Statistical methods for identifying differentially expressed genes in replicated cDNA microarray experiments. *Statistica Sinica*, 12:111–140, 2002.

[ETST01] B. Efron, R. Tibshirani, J. D. Storey, and V. Tusher. Empirical Bayes Analysis of a Microarray Experiment. *Journal of the American Statistical Association*, 96:1151–1160, 2001.

[FFN+06] M. Fujita, Y. Fujita, Y. Noutoshi, F. Takahashi, Y. Narusaka, K. Yamaguchi-Shinozaki, and K. Shinozaki. Crosstalk between abiotic and biotic stress responses: a current view from the points of convergence in the stress signaling networks. *Current Opinion in Plant Biology*, 9:436–442, August 2006.

[JGS+03] Z.V. Joseph, G. Gerber, I. Simon, D.K. Gifford, and T.S. Jaakkola. Comparing the continuous representation of time-series expression profiles to identify differentially expressed genes. *Proceedings of the National Academy of Sciences of the United States of America*, 100:10146–51, September 2003.

[KMC00] M.K. Kerr, M. Martin, and G.A. Churchill. Analysis of Variance for Gene Expression Microarray Data. *Journal of Computational Biology*, 7(6):819–837, 2000.

[Mac03] D. J. C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003.

[RG01] C. E. Rasmussen and Z. Ghahramani. Infinite Mixtures of Gaussian Process Experts. In *Advances in Neural Information Processing Systems*, volume 13, pages 881–888. MIT Press, 2001.

[SDW+09] O. Stegle, K. Denby, D. L. Wild, Z. Ghahramani, and K. M. Borgwardt. A robust Bayesian two-sample test for detecting intervals of differential gene expression in microarray time series. *Lecture Notes in Computer Science (RECOMB)*, 2009.

[SXL+05] J. D. Storey, W. Xiao, J. T. Leek, R. G. Tompkins, and R. W. Davis. Significance analysis of time course microarray experiments. *Proceedings of the National Academy of Sciences of the United States of America*, 102:12837–42, September 2005.

[SYS00] K. Shinozaki and K. Yamaguchi-Shinozaki. Molecular responses to dehydration and low temperature: differences and cross-talk between two stress signaling pathways. *Current Opinion in Plant Biology*, 3(3):217–223, 2000.

[TS06] Y. C. Tai and T. P. Speed. A multivariate empirical Bayes statistic for replicated microarray time course data. *Annals of Statistics*, 34:2387–2412, 2006.

[Yua06] M. Yuan. Flexible temporal expression profile modelling using the Gaussian process. *Computational Statistics and Data Analysis*, 51:1754–1764, 2006.

Comparative Generalized Logic Modeling Reveals Differential Gene Interactions during Cell Cycle Exit in *Drosophila* Wing Development

Mingzhou (Joe) Song†, Chung-Chien Hong†, Yang Zhang†, Laura Buttitta‡, Bruce A. Edgar‡
†Department of Computer Science, New Mexico State University, Las Cruces, U.S.A.
‡Division of Basic Sciences, Fred Hutchinson Cancer Research Center, Seattle, U.S.A.

Abstract: A comparative interaction detection paradigm is proposed to study the complex gene regulatory networks that control cell proliferation during development. Instead of attempting to reconstruct the entire cell cycle regulatory network from temporal transcript data, differential interactions – represented by generalized logic – are detected directly from time course transcript data under two distinct conditions. This comparative approach is scale- and shift-invariant and is capable of detecting nonlinear differential interactions. Simulation studies on *E. coli* circuits demonstrated that the proposed comparative method has substantially increased statistical power over the intuitive reconstruct-then-compare approach. This method was therefore applied to a microarray experiment, profiling gene expression in the fruit fly wing as cells exit the cell cycle, and under a condition which delays this exit, over-expression of the cell cycle regulator E2F. One statistically significant differential interaction was identified between two gene clusters that is strongly influenced by E2F activity, and suggests the involvement of the Hippo signaling pathway in response to E2F, a finding that may provide additional insights on cell cycle control mechanisms. Furthermore, the comparative modeling can be applied to both static and dynamic gene expression data, and is extendible to deal with more than two conditions, useful in many biological studies.

1 Introduction

Comparative experimental designs for gene expression studies have yet to be explored for their full potential in understanding *differential* and *conserved* gene interactions in important biological phenomena such as cell cycle control. An interaction is an association from one or more parent genes to a child gene. Because complex interactions may only stand out when contrasted, we have developed a comparative modeling paradigm to detect novel gene interactions, represented by generalized logic (**glog**), for such experiments. Our strategy, based on heterogeneity and homogeneity chi-square tests, extends meta-analysis which has been traditionally used for comparing data sets of similar studies from different researchers. Our new comparative modeling approach is designed to uncover novel gene interactions, missed by other approaches.

Our goal is to fundamentally increase sensitivity in detecting how gene interactions may be either conserved or shifted in a comparative experiment. As microarray technologies mature, many approaches to gene expression analysis have been developed. Some per-

form single-gene differential expression analysis [TTC01], ignoring either dynamics or gene interactions; while others carry out gene regulatory network reconstruction [Fri04], relying on exhaustive genome-wide perturbation experiments for mathematical accuracy. As pointed out by Bonneau [Bon08], reconstruction is often cost-ineffective and believed to be "beyond our current reach". A step forward is the strategies summarized in [TBB07] that identify conserved and differential interactions by shifted Pearson linear correlation coefficients, which do not integrate temporal associations, nonlinear interactions, or interactions involving more than two genes. Still no rigorous statistical framework exists for comparative gene interaction detection beyond pair-wise linear correlation.

Our innovation is to extend the heterogeneity and homogeneity chi-square tests by associating child gene expression with potential parent gene expression at the same or previous time points. This association takes a non-parametric form that can be highly nonlinear. Our approach generalizes the correlation-based comparisons [BL06], which can be considered a single-parent, linear, zero-delay, and static interaction. We use a glog to represent an interaction. Our approach directly assesses the contrastive strength of a pair of potential interactions, instead of reconstructing-then-comparing the interaction under each condition. An interaction will be selected if it consistently shows either similar or differential patterns. Such a strategy embraces uncertainty in glog, while other approaches assume zero variance. A remarkable property of this strategy is its determination of parents without having to estimate accurately the actual glog. Although this paper explores discrete differential gene interactions, we have also developed a nice analogous approach for continuous differential interactions [OS09]. The discrete approach captures switch-like behaviors of interactions, while the continuous approach is effective for subtle and gradual interactions, complementarily.

The biological phenomenon we examine with this approach is cell cycle exit, an event critical during the process of organism development, and mis-regulated in cancers. Normally, cells differentiating into their final fates exit the cell cycle and become unresponsive to proliferative cues, but this process is somehow blocked or disrupted in cancer. To uncover how differentiation so potently blocks the cell cycle, we have examined the process of cell cycle exit in the model organism *Drosophila melanogaster*. *Drosophila* has been a key organism for studies of the cell cycle and provides an excellent system for a wide array of genetic manipulations. The *Drosophila* wing is particularly useful for studies of cell cycle control because it is highly homogenous with over 90% of the cells consisting of a single epithelial cell type, which undergo a well-characterized temporally synchronized cell cycle exit [SP87, MCGB96, BKP$^+$07]. Due to this synchrony, it is an excellent system for a time-course study of differential genetic interactions upon cell cycle exit *in vivo*.

The final cell cycle in the wing occurs between 122-144 hours of development. Exactly how this relatively synchronous cell cycle exit is controlled remains unknown, but restraining the activity of the transcription factor complex E2F has been shown to be critical for the proper timing of exit *in vivo* [BKP$^+$07]. The E2F transcription factor complex is a master regulator of cell cycle genes, promoting expression of many genes for G1-S as well as G2-M cell cycle transitions. Consistent with its role in promoting the cell cycle, the E2F complex is a well-established target for negative regulation by tumor suppressor proteins such as Retinoblastoma. It is also positively regulated by oncogenes such as SV40 Large

T and Adenovirus E1A [vdHD08]. We have found the E2F complex to regulate the expression of ∼900 genes, covering a number of cell cycle regulators, chromatin modifiers and other factors comprising the "E2F transcriptional program".

By comparative modeling on gene expression under normal conditions and conditions where E2F activity is high, we successfully identified a significant differential interaction between two clusters of genes influenced by E2F activity during cell cycle exit. We propose that this approach uncovers novel genetic networks that are perturbed upon aberrant E2F activity, providing insight into the global function of this transcription factor *in vivo*.

2 Interactions in generalized logic and their reconstruction

Let child node X have Q quantization levels ranging from 0 to $Q-1$, controlled by K parents $z_1, z_2, \ldots, z_K$ of $Q_1, Q_2, \ldots, Q_K$ quantization levels, respectively. The glog H of node X is a function that maps all possible combinations of parent node values to values of X. We also call glog H an *interaction*. The glog can incorporate temporal dependencies by introducing time t and delays of each parent $\tau_1, \ldots, \tau_K$.

We apply chi-square test to detect an interaction from a contingency table obtained from experimental data. The number of rows in the table is $R = Q_1 Q_2 \cdots Q_K$ and the number of column is Q $n_{r,c}$ is the number of observations in which the parents take the values in the r-th row and X takes the value of c. Let $n_{\cdot,c}$ be the sum of column c. Let $n_{r,\cdot}$ be the sum of row r. Let $\bar{n}_{r,c} = n_{r,\cdot} n_{\cdot,c}/n$ be the expected count when the parents are not associated with X. Then, $\chi^2 = \sum_{r=0}^{R-1} \sum_{c=0}^{Q-1} \frac{(n_{r,c}-\bar{n}_{r,c})^2}{\bar{n}_{r,c}}$ is asymptotically chi-square distributed with $(R-1)(Q-1)$ degrees of freedom (d.f.) when the parents do not influence the child. Further details can be found in [SLLea09].

3 Differential interactions and their detection by heterogeneity tests

An interaction is *conserved* if it does not change from one condition to another; otherwise, it is *differential* if any change occurs in parent identity or strength for any parent. An interaction under two conditions can have both *homogenous* and *heterogenous* components: the former represents an overall agreement of the interaction under the two conditions; the

	0	1	2
0	X	0	0
1	0	X	0
2	0	0	X

	0	1	2
0	0	0	X
1	0	X	0
2	X	0	0

(a) Detectable linear differential interaction: 1 versus -1 for Pearson coefficients.

	0	1	2
0	X	0	X
1	X	0	X
2	0	X	0

	0	1	2
0	0	X	0
1	X	0	X
2	X	0	X

(b) Undetectable nonlinear differential interactions: 0 versus 0 for Pearson coefficients.

Figure 1: Linear correlation differential interaction detection: Detectable and undetectable.

latter represents deviation from the overall agreement.

Existing comparative methods compare interactions numerically, ignoring the variance in the estimated models. For example, pair-wise linear correlation based approaches will be effective on linear differential interaction detection (Fig. 1(a)), but not nonlinear ones

(Fig. 1(b)). Our strategy will instead consider both nonlinearity and uncertainty in two data sets collected under comparative experimental conditions. Such a consideration enables much greater statistical power than other approaches.

3.1 Detect the differential interaction of a child with known parents

We develop a procedure based on chi-square statistics to determine whether a fixed topology interaction shows any significant shift under two conditions. The null hypothesis assumes no interaction between the parents and the child. The test statistics measure the homogenous and heterogenous components in interactions, illustrated in Fig. 2.

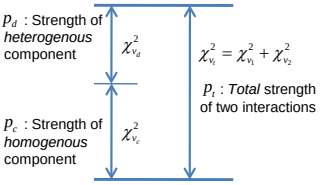


Figure 2: Components in an interaction.

Let the two data sets (temporal or static), collected under two different conditions, be $\mathcal{T}_1$ and $\mathcal{T}_2$. Let $\pi \rightarrow X$ represent an interaction from parents π to child X. We first obtain the contingency tables C_1 and C_2 from $\mathcal{T}_1$ and $\mathcal{T}_2$, respectively, associated with $\pi \rightarrow X$. $\chi^2_{v_1}$, d.f. v_1, and p-value p_1 are computed from C_1, so do $\chi^2_{v_2}$, v_2, and p_2.

The total chi-square from two interactions is $\chi^2_{v_t} = \chi^2_{v_1} + \chi^2_{v_2}$, with d.f. $v_t = v_1 + v_2$ and p-value p_t. This statistic measures by p_t the total strength of any interaction under either conditions, regardless of differential or conserved.

The homogenous component is the conserved portion of an interaction under two conditions. A contingency table C_{pool} is filled, using parent and child values, from both $\mathcal{T}_1$ and $\mathcal{T}_2$. From C_{pool}, one can compute $\chi^2_{v_c}$ with d.f. v_c and p-value p_c, which is the strength of interaction homogeneity under different conditions.

Algorithm 1 Decide-Interaction-Type(X, π, $\mathcal{T}_1$, $\mathcal{T}_2$, α)

1: Form contingency table C_1 for $\pi \rightarrow X|\mathcal{T}_1$, C_2 for $\pi \rightarrow X|\mathcal{T}_2$, and C_{pool} for $\pi \rightarrow X|\mathcal{T}_1, \mathcal{T}_2$
2: Calculate heterogenous component $\chi^2_{v_d}$ and strength p_d
3: Calculate homogenous component $\chi^2_{v_c}$ and strength p_c
4: Calculate total chi-square $\chi^2_{v_t}$ and total strength p_t
5: **if** heterogenous component is significant ($p_d \leq \alpha$) **then**
6: **if** total chi-square is significant ($p_t \leq \alpha$) **then**
7: comparative interaction type $\leftarrow$ absolute differential
8: **else**
9: comparative interaction type $\leftarrow$ relative differential
10: **end if**
11: **else if** homogenous component is significant ($p_c \leq \alpha$) **then**
12: comparative interaction type $\leftarrow$ conserved
13: **else**
14: comparative interaction type $\leftarrow$ null
15: **end if**
16: Return the comparative interaction type and $\{C_1, C_2, C_{\text{pool}}, \chi^2_{v_d}, \chi^2_{v_c}, \chi^2_{v_t}, p_d, p_c, p_t\}^{\pi}$

The heterogenous component is the differential portion of an interaction under two con-

ditions, defined by $\chi^2_{v_d} = \chi^2_{v_1} + \chi^2_{v_2} - \chi^2_{v_c}$ with d.f. $v_d = v_1 + v_2 - v_c$ and p-value p_d, which is the strength of interaction heterogeneity under different conditions.

For a parent set π, Algorithm 1 determines the interaction type: conserved, absolute or relative differential, and null. Our principle is that a pair of interactions is considered differential if it has a significant heterogenous component regardless of the significance of its homogenous component. We further classify a differential pair to be *relative* differential if the total chi-square is insignificant and otherwise *absolute* differential.

A simulation study to demonstrate the power advantage is shown in Fig. 3. The power gain can be as high as about 40% when the noise is at an intermediate level.

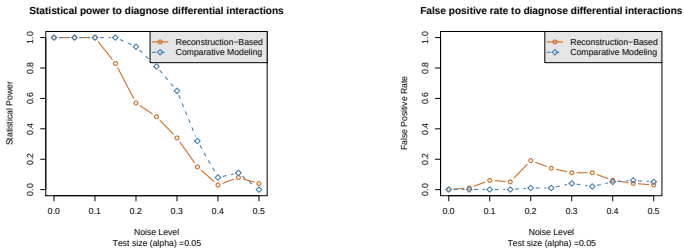


Figure 3: Advantage of comparative modeling versus reconstruct-then-compare in statistical power and false positive rate of differential interactions when parent are fixed. The example was based on a 2-parent binary interaction.

3.2 *Ab initio* comparative interaction modeling in networks of unknown topology

In *ab initio* comparative modeling, we find differential or conserved interactions of each child when parents identities are unknown. This is thus both a modeling problem to determine the most likely parents for each child, as well as a detection problem to check differential interactions. The rationale of such an approach lies in that it is unlikely for non-parents to show consistently differential or conserved interactions with a child.

Table 1: Selection of parent sets.

Interaction $\pi_1 \to X$	**Interaction** $\pi_2 \to X$	**Condition**	**True**	**False**
Conserved	Conserved	$p_c^{\pi_1} \leq p_c^{\pi_2}$	π_1	π_2
Abs. differential	Abs. differential	$p_t^{\pi_1} \leq p_t^{\pi_2}$	π_1	π_2
Rel. differential	Rel. differential	$p_d^{\pi_1} \leq p_d^{\pi_2}$	π_1	π_2
Conserved	Abs. differential	$p_c^{\pi_1} \leq p_t^{\pi_2}$	π_1	π_2
Abs. differential	Conserved	$p_c^{\pi_2} \leq p_t^{\pi_1}$	π_2	π_1
Conserved or abs. diff.	Rel. diff.	-	π_1	π_1
Rel. diff.	Conserved or abs. diff.	-	π_2	π_2
null	null	$p_t^{\pi_1} \leq p_t^{\pi_2}$	π_1	π_2
non-null	null	-	π_1	π_1
null	non-null	-	π_2	π_2

We compare two parent sets, π_1 and π_2, for child X, using Table 1. The interaction types under each parent set are determined first. We assume all p-values have been adjusted for multiple comparisons. The selection of parent set is based on four interaction types. The same type is compared by the p-values associated with that type. When the types differ, p-values are not compared but a prioritized list of conserved or absolute differential, relative

differential, and null is used. The principle is that a conserved or absolute differential parent set is selected over a relative differential one and non-null is over null. If both are null, the parent set with smaller p_t is selected.

4 Simulation study on comparing 78 pairs of *E. coli* circuits

The simulation study in Fig. 3 indicates that for known parents, the statistical power for comparative modeling is higher to reconstruct-then-compare under the same false positive rate. We now evaluate the performance of *ab initio* comparative modeling (Section 3.2), in reference to the reconstruct-then-compare approach. We used 13 *E. coli* networks [GEHL02], also called circuits, to form 78 pairs of circuits for comparison. Each circuit has four binary nodes: the inducible repressors (LacI and TetR), λ CI, and GFP. All circuits are first Markovian with a maximum of one parent for each child.

We evaluate the performance node-wise. An interaction for a node is true negative (TN) if it is conserved and announced so, false positive (FP) if conserved announced differential, false negative (FN) if differential announced conserved, and true positive (TP) if differential announced so. The performance of comparing two networks is accumulated over all the nodes in them. The network TNs is the total number of TN children, FPs the total number of FP children, FNs the total number of FN children, and TPs the total number of TP children.

The noise model is defined such that one node at a particular expression level is more likely to jump to its adjacent levels:

$$P(j|i,\theta) = \begin{cases} \left(1 - \frac{|j-i|}{\sum_{d=0}^{K-1}|d-i|}\right)\frac{\theta}{K-1}, & j \neq i \\ 1-\theta, & j = i \end{cases} \quad (1)$$

where θ denotes the noise level (from 0 to 1)[1], j denotes the noisy version of true value i, and K is the number of quantization levels.

Figure 4 shows the performance advantage of comparative modeling versus the reconstruct-then-compare approach.[2] When noise level is relatively high (0.1), comparative modeling significantly outperformed: its TPs is almost twice of reconstruct-then-compare. This implies comparative modeling can detect differential interactions more accurately without increasing FPs.

5 Differential gene interactions in cell cycle exit in *Drosophila* wings

We next applied comparative modeling to the study of cell cycle control during development *in vivo*. For this study, we obtained transcriptomic profiles of *Drosophila* wings

[1] For comparative modeling, the worst noise is 0.5.

[2] The non-monotonic ROC of the reconstruct-then-compare approach is expected as a pair of differential interactions involving a null and a non-null can become conserved of two null interactions when α increases.

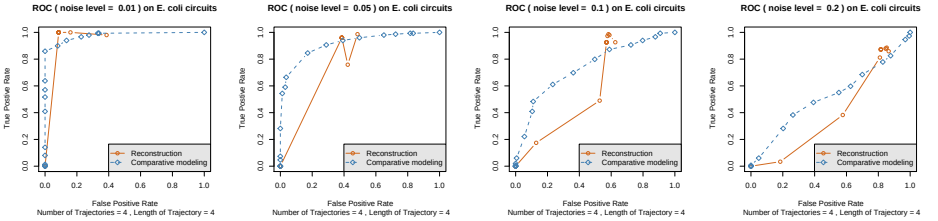


Figure 4: ROC advantage of comparative modeling versus reconstruct-then-compare, demonstrated by a simulation study on 78 pairs of *E. coli* circuits.

during cell cycle exit under normal conditions, and conditions of excessive E2F activity.

Microarray experiments – Ten pupal wings from either control animals (E2F-) or animals expressing the E2F/DP transcription factor complex under the control of the Gal4/UAS system (E2F+) were dissected at 0h, 24h, and 36h after pupa formation (APF). RNA was isolated using Trizol, and cDNA synthesis was performed with one subsequent round of T7-dependent linear RNA amplification using the commercially available Message AmpTM kit from Ambion. Amplified RNA labeled and hybridized to Nimblegen *Drosophila* expression arrays of 15,473 probes according to the manufacturer's specifications. Hybridizations were repeated 4 times with independently obtained samples. Microarray scanning and normalization was performed as recommended by the manufacturer. Importantly, cell cycle exit occurs at 24h APF under normal conditions (E2F-), while under E2F+ conditions cells go through an extra cycle and instead exit at 36h APF [BKP$^+$07].

Preprocessing – Two-way ANOVA on time (24h/36h), condition (E2F+/-), and their interaction was applied to filtered out genes insignificantly differentially expressed, resulting in 5,867 selected out of 15,473. We performed hierarchical clustering to form 127 groups of linearly correlated transcripts at 24h and 36h. A total of 127 representatives that best represent transcripts in each cluster were selected. Genes in the same cluster are considered mathematically equivalent and only the representatives were used in the subsequent modeling. A joint quantization was applied to convert continuous gene expression levels at 0h, 24h, and 36h to discrete levels of low, intermediate, and high.

Comparative modeling – Comparative glog interaction modeling was applied to data at 24h and 36h to contrast the interactions under E2F+ versus E2F-. The α-level used was 0.05. The maximum number of parents is 1.

Differential interactions – The only significant differential interaction detected is from cluster C125(22) to C34(59). The number in the parentheses is the total number of genes in that cluster. The original gene expression levels in the two clusters are shown in Fig. 5.

Table 2 shows the observed differential interaction between C125(22) to C34(59). The C125(22)→C34(59) interaction contains a significant heterogenous component ($p_d = 0.031$) and is also overall significant ($p_t = 0.039$), indicating a consistent shift in the way the two clusters interact under E2F+ or - conditions.

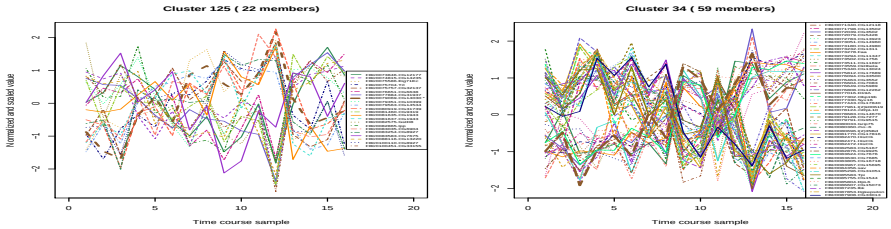


Figure 5: Expression levels (scaled and shifted) of transcripts in clusters C125(22) and C34(59). Time course sample index 1 to 8 represent 8 replicates under E2F- (1 to 4 at 24h; 5 to 8 at 36h). Sample index 9 to 16 represent 8 replicates under E2F+ (9 to 12 at 24h; 13 to 16 at 36h).

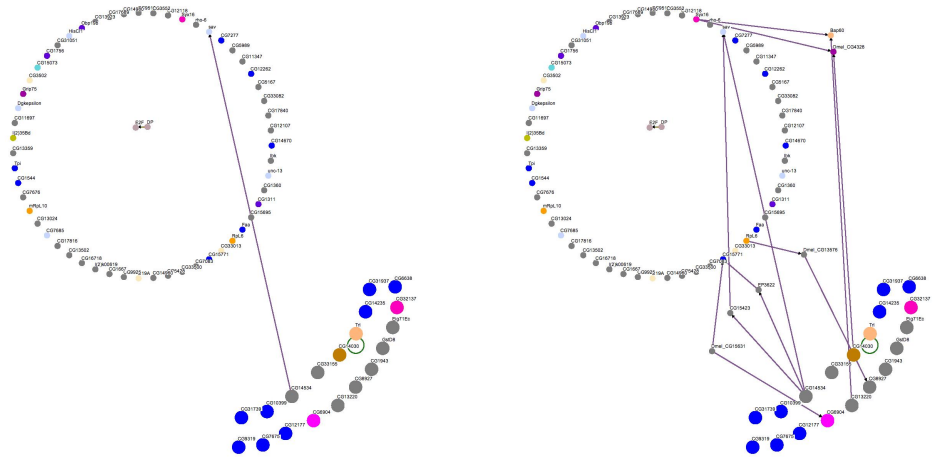
Table 2: Interaction of E2F and C125(22) with C34(59). The numbers in the table represent the occurrences of the associations in the observed expression data.

		C34(59)		
E2F	C125(22)	Low	Intermediate	High
-	Low	4	1	0
-	Intermediate	0	1	1
-	High	1	0	0
+	Low	0	0	2
+	Intermediate	0	2	0
+	High	4	0	0

6 Discussion

From this preliminary analysis we identified two clusters of genes, C125(22) and C34(59), that display a differential interaction under high E2F activity. Surprisingly, few of the genes in these clusters have known roles in cell cycle control, and none have known interactions with E2F. We have examined whether any genes within these clusters have any known genetic or physical interactions with each other, either directly or through secondary partners, using the FlyGRID database and the Osprey network visualization program. Figs. 6(a) and 6(b) show the results of the analysis. While we found no direct known interactions between E2F and the two clusters, we do find a single known direct interaction between the clusters, via CG14534 binding to Salvador (Sav) in a yeast two-hybrid protein binding assay (Fig. 6(a)). Sav is a scaffolding protein, known to be a key component of the Hippo pathway, a pathway involved in cell growth and proliferation [Edg06]. This interaction will therefore have the highest priority for further validation.

Additionally, these two clusters have multiple interactions through secondary partners. For example, CG14534 interacts with 3 targets in C34: CG15771, Syx16 and Sav, via protein-protein binding through secondary partners. CG14030 interacts with Syx16 (C34) through a secondary partner CG4328 and CG13220 (C125) also interacts with Syx16 (C34) via the chromatin modifier Bap60. Thus additional connections can be drawn through intermediate partners with Sav and Syx16 being the most highly connected targets in C34. However, two more interactions between these clusters are independent of the highly connected Syx16 and Sav nodes. They are: the CG6904 (C125) interaction with CG15771 (C34) via CG15631 and the CG8927 (C125) interaction with RpL6 (C34) through CG13576. These



(a) A known genetic and protein-protein interaction from CG14534 in C125(22) to Sav in C34(59).

(b) Genetic and protein-protein interactions including secondary interactors.

Figure 6: Known direct and secondary interactions among genes in C125(22) and C34(59) were provided by FlyGRID and Osprey. Genes in C34 are displayed as small nodes in a circular array where colors indicate different gene ontology annotations. Genes in C125 are displayed as large nodes aligned at right. E2F and DP are displayed as small nodes in the center of the C34 circular array. Known genetic or physical interactions are represented as edges between nodes with purple edges indicating a physical interaction via yeast two-hybrid assays.

results suggest several potential networks for further investigation (Fig. 6(b)).

Interestingly, a genetic interaction between the Hippo signaling pathway and E2F was recently described [NF08], where repression of Hippo signaling resulted in increased E2F expression and activity. In contrast, our work suggests that activation of E2F also alters the level of Hippo signaling via changes in Sav expression. Together these genetic interactions could result in a feedback loop, stably coordinating changes in E2F activity during development *in vivo* with compensatory alterations in Hippo signaling. We plan to further test this hypothesis by direct genetic experiments examining Hippo signaling *in vivo*.

Importantly, our modeling approach allows new interactions present only under certain conditions to be uncovered. Therefore we do not expect that many of the important interactions will be identified by the genome-wide analyses present in the database, which are done exclusively under normal conditions. To address this in future work, we can systematically test the requirement for certain genes in cluster C125 on the induction of genes in C34 under high E2F activity at 36h. This could be carried out using gene specific RNAis to knock-down the levels of highly connected genes in C125 to test the effects on transcripts in C34 by quantitative RT-PCR.

We have demonstrated that novel genetic interactions can be proposed from modeling gene expression associations at the same time point. However our total sample size of 16 for the comparative analysis is small. In future work, by doubling the sample size, the statistical

power is expected to improve substantially. By increasing the number of time points, we will expand our efforts to detect differential temporal interactions. We anticipate comparative modeling will enable more fundamental understanding of gene expression programs either within a species under different conditions or across species under same conditions.

References

[BKP+07] L Buttitta, A Katzaroff, C Perez, A de la Cruz, and BA Edgar. A double-assurance mechanism controls cell cycle exit upon terminal differentiation in *Drosophila*. *Dev. Cell*, 12(4):631–643, 2007.

[BL06] J Berg and M Lässig. Cross-species analysis of biological networks by Bayesian alignment. *Proc Natl Acad Sci U S A*, 103(29):10967–10972, 2006.

[Bon08] R Bonneau. Learning biological networks: from modules to dynamics. *Nat Chem Biol*, 4(11):658–64, 2008.

[Edg06] BA Edgar. From cell structure to transcription: hippo forges a new path. *Cell*, 124(2):267–273, 2006.

[Fri04] N Friedman. Inferring cellular networks using probabilistic graphical models. *Science*, 303:799–805, 2004.

[GEHL02] CC Guet, MB Elowitz, W Hsing, and S Leibler. Combinatorial synthesis of genetic networks. *Science*, 296:1466–1470, 2002.

[MCGB96] M Milan, S Campuzano, and A Garcia-Bellido. Cell cycling and patterned cell proliferation in the *Drosophila* wing during metamorphosis. *Proc Natl Acad Sci U S A*, 93(21):11687–11692, 1996.

[NF08] BN Nicolay and MV Frolov. Context-dependent requirement for dE2F during oncogenic proliferation. *PLoS Genet.*, 4(10):e1000205, 2008.

[OS09] Z Ouyang and M Song. Comparative identification of differential interactions from trajectories of dynamic biological networks. In *Proceedings of German Conference on Bioinformatics*, Halle, Germany, September 2009.

[SLLea09] M Song, CK Lewis, ER Lance, and et al. Reconstructing generalized logical networks of transcriptional regulation in mouse brain from temporal gene expression data. *EURASIP J Bioinform Syst Biol*, 2009. Article ID 545176, 13 pages.

[SP87] M Schubiger and J Palka. Changing spatial patterns of DNA replication in the developing wing of *Drosophila*. *Dev Biol.*, 123(1):145–153, 1987.

[TBB07] I Tirosh, Y Bilu, and N Barkai. Comparative biology: beyond sequence analysis. *Curr Opin Biotechnol*, 18(4):371–377, 2007.

[TTC01] VG Tusher, R Tibshirani, and G Chu. Significance analysis of microarrays applied to the ionizing radiation response. *PNAS*, 98:5116–5121, 2001.

[vdHD08] S van den Heuvel and N Dyson. Conserved functions of the pRb and E2F families. *Nat Rev Mol Cell Biol.*, 9(9):713–724, 2008.

Expression profiles of metabolic models to predict compartmentation of enzymes in multi-compartmental systems

Achuthanunni Chokkathukalam, Mark Poolman, Chiara Ferrazzi and David Fell

cbaunni@brookes.ac.uk

Abstract: Enzymes and other proteins coded by nuclear genes are targeted towards various compartments in the plant cell. Here, we describe a method by which localisation of enzymes in a plant cell may be predicted based on their transcription profile in conjunction with analysis of the structure of the metabolic network. This method uses reaction correlation coefficients to identify reactions in a metabolic model that carry similar flux.

First a correlation matrix for the expression of genes of interest is calculated and the columns clustered hierarchically using the correlation coefficient. The rows clustered using reaction correlation coefficients. In the resulting matrix, we show that the genes in a particular compartment are clustered together and compartmental predictions, with respect to a reference gene can be readily made.

1 Introduction

Spatial organisation of metabolism and other cellular functions is a well known feature of plant cells. Enzymes and other proteins coded by nuclear genes are targeted towards various compartments in the plant cell with the help of the targeting information within their amino acid sequence. Identifying the localisation of proteins is thus an important step towards a broader understanding of the cellular function as a whole and may help in determining the role of thousands of uncharacterised proteins predicted by the genome sequencing projects. Modern organelle-focused experimental approaches can identify proteins in a given compartment. However, reliable protein localisation requires that the technique used must be able to distinguish between genuine organelle residents and contaminating proteins [DDWL04]. Although reasonably pure preparations of some organelles can be achieved, there are many difficulties associated with measuring and characterising proteins that are in a compartment [DHS$^+$06]. Nevertheless, a variety of experimental methods are currently being used to identify protein localisation. Recently chimeric fusion proteins (FPs) and mass spectrometry (MS) techniques have been successfully employed to deduce the localisation of approximately 1100 and 2600 proteins, respectively [HVTF$^+$06]. Although these techniques have accelerated the flow of protein localisation information, the subcellular location of the majority of proteins in a plant cell is still not known.

A relatively simple, low-cost and rapid means to tackle this issue is to employ bioinfor-

matic targeting algorithms to predict protein localisation from amino acid sequence. A number of software tools exists, including TargetP [EBvHN07], Predotar [SPLL04], iPSORT [BTM+02], SubLoc [HS01], MitoProt II [CV96], MITOPRED [GFS04], PeroxiP [EEvHC03], and WoLF PSORT [HPO+07], which can predict proteins targeted towards plastid, cytosol, nucleus, mitochondria, peroxisome or the endoplasmic reticulum. However, the output of such programs has been found to be somewhat inconsistent with each other, or with experimentally determined results [HVTFM05], making them unreliable for some analyses.

The advent of whole-system approaches such as microarrays and metabolomics and the accumulation of such high-throughput data have created new opportunities for studying how reactions are coordinated to meet cellular demands. Microarray experiments monitor the expression of thousands of genes simultaneously. Grouping together genes of similar expression pattern is a general starting point in the analysis of expression data. Similarity between genes is measured by the correlation of their expression profiles and hierarchical clustering methods are used to partition data into clusters of genes exhibiting similar expression patters [IBB04]. Numerous studies have shown that co-expression patterns of gene expression across many microarray datasets form modules of genes that are functionally correlated [WPM+06, MDO+08]. Recently this approach was successfully employed in identifying new genes involved in cellulose synthesis in plants [PWM+05].

Here, we describe a method by which localisation of enzymes may be predicted based on the co-expression profiles of genes coding for reactions in a structural model of plant carbon metabolism. Structural models contain stoichiometries of reactions in a metabolic system. Based on the correlation between these reactions, it can be represented hierarchically as a metabolic tree in which the root node represents the complete system, leaf nodes represent individual reactions, and the intermediate nodes represent metabolic modules capable of the net interconversion of metabolites common to reactions inside and outside the module [PSPF07]. Our technique uses reaction correlation profiles generated from metabolic models together with expression correlation profiles obtained from the microarray data to identify the distribution of enzymes in a particular compartment with respect to the experimentally determined location of a protein representing that compartment.

2 Materials and methods

2.1 Construction of the model of plant carbon metabolism

A structural model of plant carbon metabolism including plastid and cytosol compartments was constructed (Figure 1). The model contains reactions of the Calvin cycle, light reactions and glycolysis and is based, in part, on previous models of plant metabolism constructed in our group [PFR03, Ass05]. Protons, CO_2, pyruvate and sucrose were made external (metabolites that are in constant exchange with the extracellular environment) yielding a model with a total of 53 reactions and 49 metabolites. Reversibility of the reactions was determined based on literature. All modelling and model analysis were performed

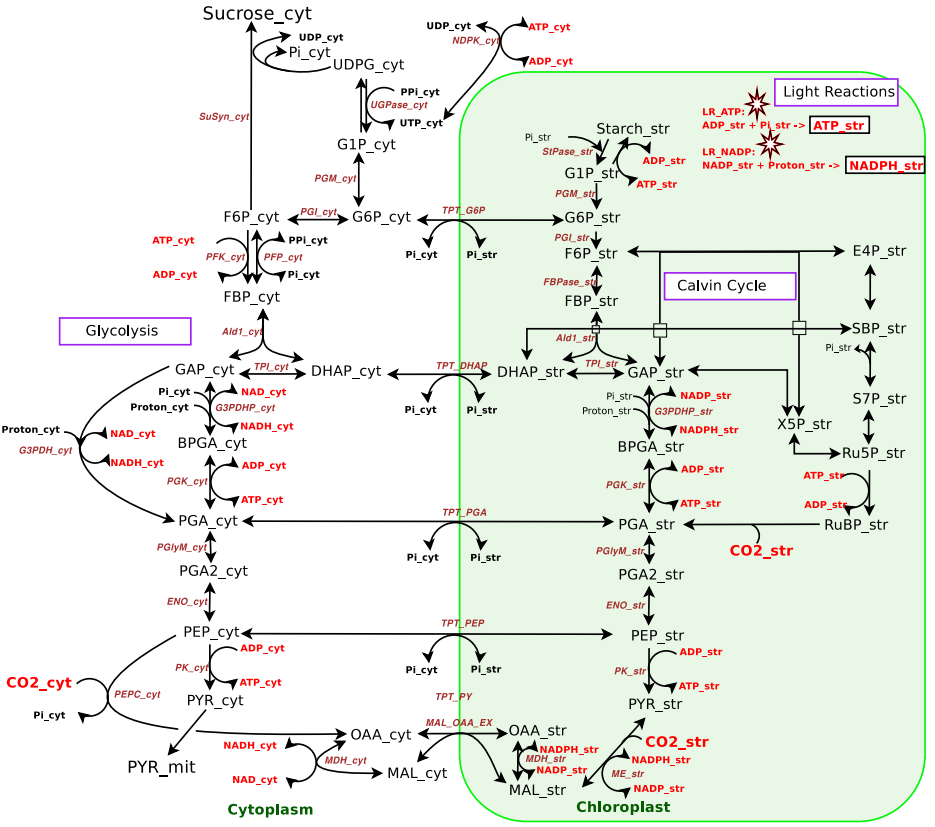


Figure 1: Reaction schema of the model of plant carbon metabolism. For simplicity, the light reactions are depicted here as two separate reactions producing ATP and NADPH. Protons, CO_2 and sucrose are considered external. '_str' and '_cyt' represent the compartments stroma and cytosol, respectively. Notice the transporters connecting reactions of the plastid and the cytosol.

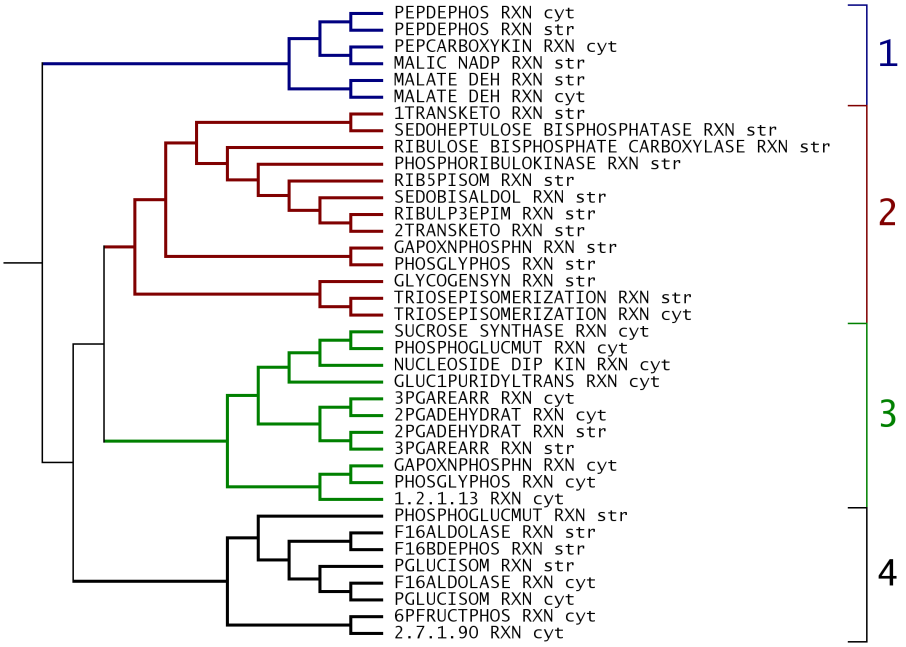


Figure 2: Metabolic tree constructed from the model showing four separate clusters containing reactions capable of net interconversion of metabolites; A. Reactions of the Malate/Oxaloacetate shuttle. B. Calvin cycle reactions. C. Reactions of glycolysis. D. Reactions involved in the regeneration of cytosolic UDP. '_str' and '_cyt' represent the compartments chloroplast and cytosol, respectively.

using the metabolic modelling tool ScrumPy (http://mudshark.brookes.ac.uk) [Poo06].

The model represents the formation of sucrose and pyruvate from the Calvin cycle intermediates transported to the cytosol via specific transport proteins. It contains several reactions such as phosphoglyceromutase, enolase, pyruvate kinase and malate dehydrogenase that are active in both the chloroplast and cytosol. Presence of these reactions in the model will enable us not only to identify their distribution between the compartments but also to distinguish isoforms of genes that code for same reactions in both the compartments. This model is publically available as SBML or in the ScrumPy '.spy' format (http://mudshark.brookes.ac.uk/index.php/User:Cbaunni).

2.2 Expression data analysis of genes coding for reactions in the model

The gene to reaction associations describe the dependence of reactions on genes. The gene to reaction associations in the model were mapped using the AraCyc [ZFT+05] database (http://www.arabidopsis.org/biocyc/index.jsp). The result is a set of genes that potentially code for all the reactions in the model.

The expression data for analysing these genes were obtained from the Nottingham *Arabidopsis* Stock Centre's (NASC) microarray database (http://affymetrix.arabidopsis.info/). The 'super bulk gene' file containing nearly 3500 hybridisations, each with expression level measurements for over 22000 genes represented on the ATH1 array was downloaded (http://affymetrix.arabidopsis.info/narrays/help/usefulfiles.html, March 2009). Expression data from individual experiments were log-transformed; no further modification or scaling was made on the data unless otherwise specified. All microarray data analysis was performed using custom modules designed for ScrumPy.

Expression data for genes ultimately coding for reactions in the model were extracted and a large-scale correlation analysis of expression values between these genes were performed essentially as described by Causton *et al.* [CQB03] by calculating the Pearson's correlation coefficient.

2.3 Clustering and analysis of the correlation matrix

A metabolic tree was generated from the model using the method described in [PSPF07] (Figure 2). The order of the reactions in this tree was used to sort the genes along the rows of the correlation matrix.

The columns of the matrix were hierarchically clustered based on the Pearson's correlation coefficient and an expression correlation tree was generated (Figure 3). Leaves of this tree represent genes in the model and the intermediate nodes are clusters that represent genes sharing similar functions. The columns of the correlation matrix were then sorted in the order of the leaves of the expression correlation tree.

The correlation matrix was imported into TM4-MeV (http://www.tm4.org/mev.html) for visualisation as heatmap [ESBB98]. The metabolic trees were visualised using MEGA phylogenetic tree editor (http://www.megasoftware.net/) [KNDT08].

3 Results and Discussion

3.1 Identification of correlated genes sharing similar flux

Metabolic tree generated from the model contain four separate clusters, each representing reactions capable of net interconversion of metabolites (Figure 2). It is notable that reactions of the Calvin cycle and glycolysis are represented as separate nodes on the tree. Clustering the rows of the correlation matrix based on the genes coding for reactions represented in these nodes can rearrange the heatmap vertically based on the similarities in flux. On the other hand, hierarchically clustering the columns of the correlation matrix grouped genes horizontally depending on their levels of expression. Doing so resulted in the formation of clusters in the heatmap representing genes that are expressed together and code for enzymes that share a similar flux (Figure 4).

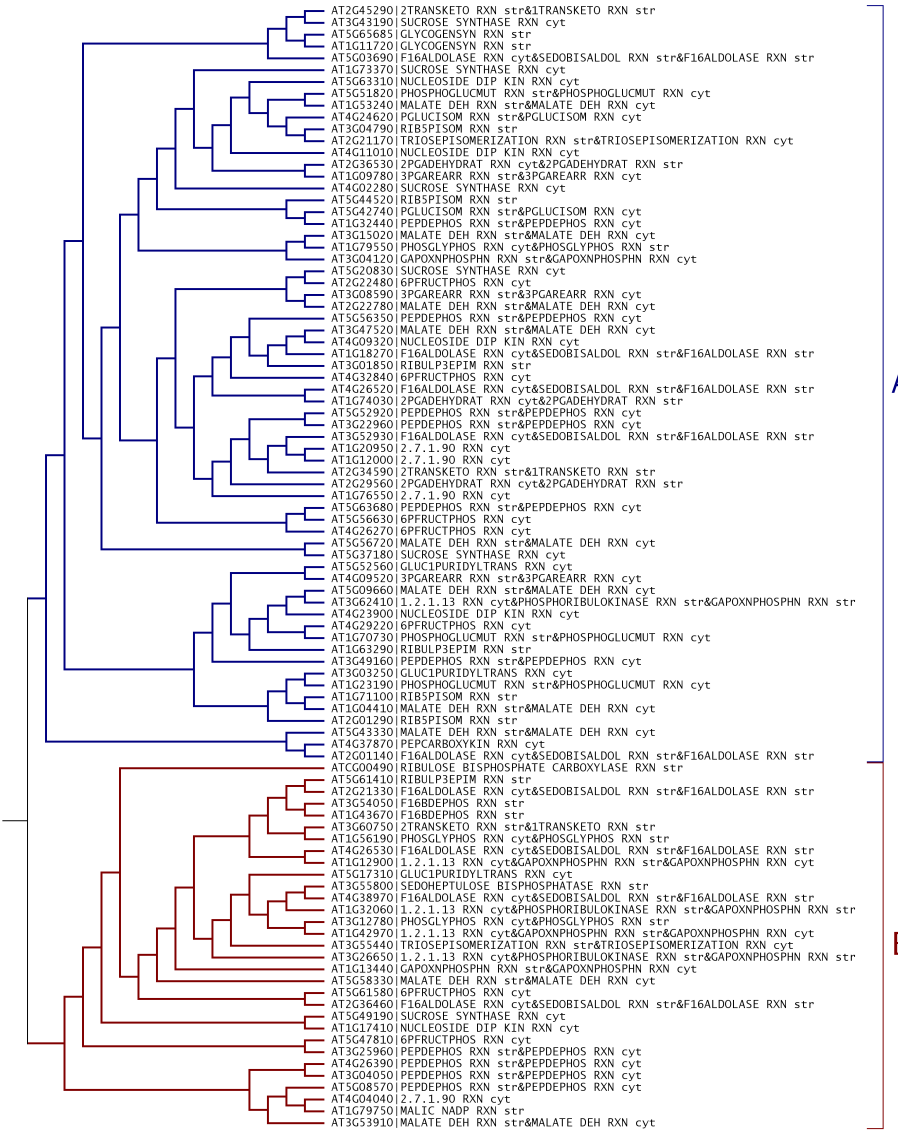


Figure 3: Expression correlation tree generated by hierarchically clustering correlation coefficients of genes coding for reactions in the model showing two separate clusters. A. Genes that predominantly code for reactions in the cytosol correlate with each other B. Genes coding for Calvin cycle intermediates cluster together. '_' is used to separate genes from reactions and '&' is used to distinguish reactions that the gene code for. '_str' and '_cyt' represent the compartments chloroplast and cytosol, respectively.

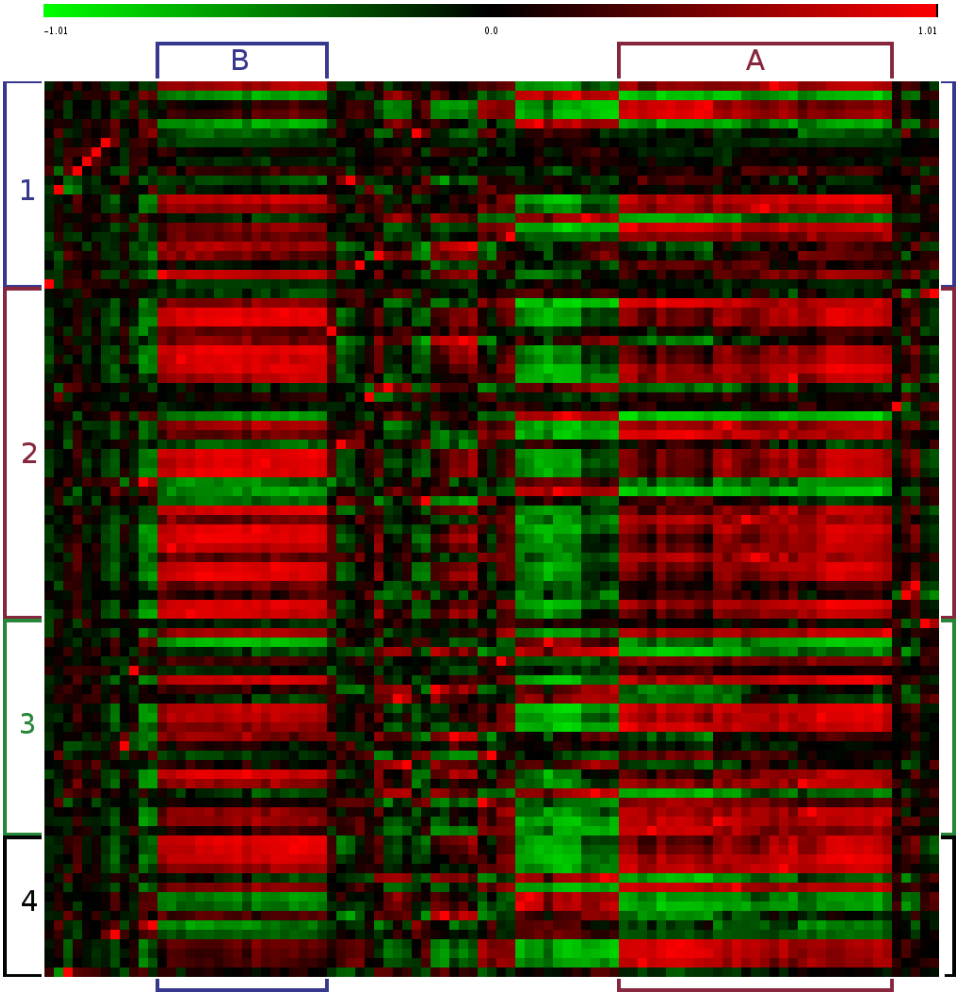


Figure 4: Correlation matrix generated from the expression values of genes coding for reactions in the steady state model. The correlation coefficient ranges from -1 (green) for perfect anticorrelation to +1 (red) for perfect correlation, with zero (black) indicating no relationship. Columns were sorted based on the clustering expression correlation coefficient and rows sorted by clustering based on reaction correlation coefficient. 'A' and 'B' represent two distinct clusters observed in the correlation matrix (Figure 3). Correlated genes in cluster 'A' were found to be highly correlated with reference genes known to be localised in the chloroplast. Whereas correlated genes in cluster 'B' showed higher correlation with genes localised in the cytoplasm. 1, 2, 3 and 4 represent clusters in the metabolic tree representing reactions capable of net interconversion of metabolites (Figure 2).

We found that genes coding for reactions in the Calvin cycle are found to be tightly correlated between each other and they cluster together. The same holds true for genes coding for glycolysis reactions. Isoforms of some Calvin cycle genes anticorrelate with other genes coding for reactions of the Calvin cycle. However, those genes that were anticorrelated with the genes of Calvin cycle reactions are found to be tightly correlated with genes of the glycolysis reactions, and vice versa. Similar cases can also be observed in case of the isoforms of glycolytic genes.

A previous study on the transcriptional coordination of metabolic network in *Arabidopsis* suggested that genes coding for reactions in a pathway show tighter levels of correlation [WPM+06]. Results from our study correlates with the above observation and also suggests that the expression profiles of genes can be used to distinguish their compartmentation.

3.2 Identifying compartmentation of genes

Though, this technique is efficient in clustering genes based on their compartmentation, identification of the compartment itself requires a reference gene whose localisation is already known. For example, the plastidic ribulose biphosphate carboxylase (Rubisco) gene ATCG00490 was used as the reference to identify genes localised in the chloroplast. Compartments are identified by filtering out genes that are highly correlated with the reference gene.

The results were compared with the various bioinformatic tools described in Section 1. Comparison with predictions made by bioinformatic tools as a whole was not possible as many of these tools were directed towards particular compartments. Compartmentation of genes that were predicted to be in the chloroplast showed good agreement with tools such as TargetP and Predotar, whereas mitochondrial predictions correlated with MITOPRED and MitoProt II predictions.

This approach was used to predict the localisation of the complete set of genes coding for the reactions in a model containing reactions of the chloroplast, cytosol and mitochondria. Given a good quality microarray expression data containing sufficient experiments that allow reliable statistical analysis, this technique can be used more generically. With the large number of publically available metabolic networks and expression data, this approach may significantly contribute to the identification of enzyme localisation in many different eukaryotic systems.

References

[Ass05] H. Assmus. *Modelling Carbohydrate Metabolism in Potato Tuber Cell*. PhD thesis, Oxford Brookes University, 2005.

[BTM+02] H. Bannai, Y. Tamada, O. Maruyama, K. Nakai, and S. Miyano. Extensive feature detection of N-terminal protein sorting signals. *Bioinformatics*, 18(2):298–305, 2002.

[CQB03] Helen C. Causton, John Quackenbush, and Alvis Brazma. *Microarray gene expression data analysis: a beginner's guide*. Wiley-Blackwell, 2003.

[CV96] M.G. Claros and P. Vincens. Computational Method to Predict Mitochondrially Imported Proteins and their Targeting Sequences. *European Journal of Biochemistry*, 241:779–786, 1996.

[DDWL04] T.P.J. Dunkley, P. Dupree, R.B. Watson, and K.S. Lilley. The use of isotope-coded affinity tags (ICAT) to study organelle proteomes in Arabidopsis thaliana. *Biochem. Soc. Trans.*, 32(3):520–523, 2004.

[DHS+06] T.P.J Dunkley, S. Hester, I.P. Shadforth, J. Runions, T. Weimar, S.L. Hanton, J.L. Griffin, C. Bessant, F. Brandizzi, C. Hawes, R.B Watson, P. Dupree, and K.S. Lilley. Mapping the Arabidopsis organelle proteome. *Proc. Natl. Acac. Sci. USA*, 103(17):6518–6523, 2006.

[EBvHN07] O. Emanuelsson, S. Brunak, G. von Heijne, and H. Nielsen. Locating proteins in the cell using TargetP, SignalP, and related tools. *Nature Protocols*, 2:953–971, 2007.

[EEvHC03] O. Emanuelsson, A. Elofsson, G. von Heijne, and S. Cristobal. In Silico Prediction of the Peroxisomal Proteome in Fungi, Plants and Animals. *Journal of Molecular Biology*, 330:443–456, 2003.

[ESBB98] M.B. Eisen, P.T. Spellman, P.O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *Proc. Natl. Acad. Sci. USA*, 95:14863–14868., 1998.

[GFS04] C. Guda, E. Fahy, and S. Subramaniam. MITOPRED: a genome-scale method for prediction of nucleus-encoded mitochondrial proteins. *Bioinformatics*, 20(11):1785–1794, 2004.

[HPO+07] P. Horton, K-J. Park, T. Obayashi, N. Fujita, H. Harada, C.J. Adams-Collier, and K. Nakai. WoLF PSORT: Protein Localization Predictor. *Nucleic Acids Research*, pages 1–3, 2007.

[HS01] S. Hua and Z. Sun. Support vector machine approach for protein subcellular localization prediction. *Bioinformatics*, 17(8):721–728, 2001.

[HVTF+06] J.L. Heazlewood, R.E. Verboom, J. Tonti-Filippini, I. Small, and A.H. Millar. SUBA: The Arabidopsis Subcellular Database. *Nucleic Acids Research*, 00:1–6, 2006.

[HVTFM05] J.L. Heazlewood, R.E. Verboom, J. Tonti-Filippini, and A.H. Millar. Combining experimental and predicted datasets for determination of the subcellular location of proteins in Arabidopsis. *Plant Physiol.*, 139(2):598–609, 2005.

[IBB04] J. Ihmels, S. Bergmann, and N. Barkai. Defining transcription modules using large-scale gene expression data. *Bioinformatics*, 20(13):1993–2003, 2004.

[KNDT08] S. Kumar, M. Nei, J. Dudley, and K. Tamura. MEGA: A biologist-centric software for evolutionary analysis of DNA and protein sequences. *Briefings in Bioinformatics*, 9(4):299–306, 2008.

[MDO+08] M. Menges, R. Doczi, L. Okresz, P. Morandini, L. Mizzi, M. Soloviev, J.A.H. Murray, and L. Bogre. Comprehensive gene expression atlas for the Arabidopsis MAP kinase signalling pathways. *New Phytologist*, 179(3):643–662, 2008.

[PFR03] M.G. Poolman, D.A. Fell, and C.A. Raines. Elementary modes analysis of photosynthate metabolism in the chloroplast stroma. *Eur. J. Biochem*, 270:430–439, 2003.

[Poo06] M.G. Poolman. ScrumPy - metabolic modelling with Python. *IEE Proceedings Systems Biology*, 153(5):375–378, 2006.

[PSPF07] M.G. Poolman, C. Sebu, M.K. Pidcock, and D.A. Fell. Modular decomposition of metabolic systems via null-space analysis. *Journal of Theoretical Biology*, 249(4):691–705, 2007.

[PWM+05] S. Persson, H. Wei, J. Milne, G.P. Page, and C.R. Somerville. Identification of genes required for cellulose synthesis by regression analysis of public microarray data sets. *Proc Natl Acad Sci USA*, 102:8633–8638, 2005.

[SPLL04] I. Small, N. Peeters, F. Legeai, and C. Lurin. Predotar: A tool for rapidly screening proteomes for N-terminal targeting sequences. *Proteomics*, 4(6):1581–90, 2004.

[WPM+06] H. Wei, S. Persson, T. Mehta, V. Srinivasasainagendra, L. Chen, G.P. Page, C. Somerville, and A. Loraine. Transcriptional Coordination of the Metabolic Network in Arabidopsis. *Plant Physiol.*, 142(2):762–774, 2006.

[ZFT+05] P. Zhang, H. Foerster, C.P. Tissier, L. Mueller, S. Paley, P.D. Karp, and S.Y. Rhee. MetaCyc and AraCyc. Metabolic pathway databases for plant research. *Plant Physiol.*, 138:27–37, 2005.

Comparative Identification of Differential Interactions from Trajectories of Dynamic Biological Networks

Zhengyu Ouyang and Mingzhou (Joe) Song
Department of Computer Science
New Mexico State University
Las Cruces, NM88003, U.S.A.
oyoung@nmsu.edu, joemsong@cs.nmsu.edu

Abstract: It is often challenging to reconstruct accurately a complete dynamic biological network due to the scarcity of data collected in cost-effective experiments. This paper addresses the possibility of comparatively identifying qualitative interaction shifts between two dynamical networks from comparative time course data. An innovative approach is developed to achieve differential interaction detection by *statistically* comparing the trajectories, instead of *numerically* comparing the reconstructed interactions. The core of this approach is a statistical heterogeneity test that compares two multiple linear regression equations for the derivatives in nonlinear ordinary differential equations, statistically instead of numerically. In detecting any shift of an interaction, the uncertainty in estimated regression coefficients is taken into account by this test, while it is ignored by the reconstruction-based numerical comparison. The heterogeneity test is accomplished by assessing the gain in goodness-of-fit from using a single common interaction to using a pair of differential interactions. Compared with previous numerical comparison methods, the proposed statistical comparison always achieves higher statistical power. As sample size decreases or noise increases in a certain range, the improvement becomes substantial. The advantage is illustrated by a simulation study on the statistical power as functions of the noise level, the sample size, and the interaction complexity. This method is also capable of detecting interaction shifts in the oscillated and excitable domains of a dynamical system model describing cdc2-cyclin interactions during cell division cycle. Generally, the described approach is applicable to comparing dynamical systems of additive nonlinear ordinary differential equations.

1 Introduction

Reconstruction of gene regulatory networks or metabolic pathways from time course observations has been a sustaining focus of efforts [BBAIdB07]. Data-driven deterministic and non-deterministic mathematical modeling methods [KWKK08] have been developed to reconstruct biological networks. Examples include Bayesian networks, Boolean networks, and ordinary/partial differential equations (ODEs/PDEs). However, accurate and complete biological network reconstruction is considered beyond our current reach. This is due to several reasons, among which are the combinatory nature of the problems, limited system perturbation and un-captured dynamical measurements [Bon08]. To remediate these limitations, we take advantage of the comparative nature of many biology

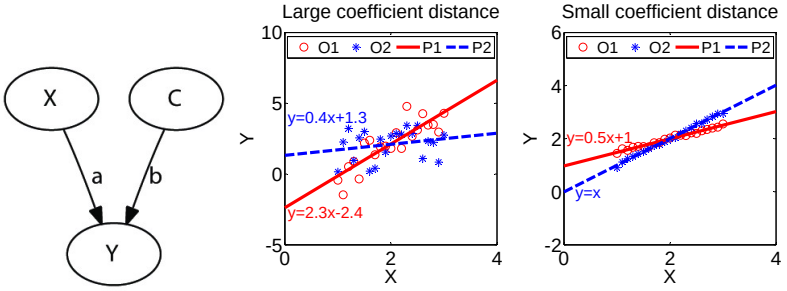


Figure 1: Unreliable NC estimation of the heterogeneity of an interaction from observations under two conditions. Left: An interaction with a coefficient vector, (a, b), which might shift under two conditions; Center: The observations, $(O1, O2)$, led to estimation of a large, but insignificant, difference between the coefficient vectors. $(P1, P2)$ are model predictions; Right: The observations, $(O1, O2)$, let to estimation of a small, but significant, difference between the coefficient vectors.$(P1, P2)$ are model predictions.

experimental designs, to pursue comparative identification of differential interactions.

When one inspects two models, numerical comparison (NC) of their coefficients is intuitive. As model databases such as BioModels [NBB+06] do not provide the data from which models are built, researchers have no choice but to use numerical comparison if they want to compare their models with those in a database. This becomes a problem when coefficients in a model have great uncertainty due to the data used to derive them. As NC methods do not consider variance in the comparison, they are effective only for accurately reconstructed networks. Therefore, several biological comparative analysis approaches have been developed [TBB07]. Differential gene expression analyses [NS04] of each single gene ignore regulatory interactions which might cause differential expression. More recently, pair-wise gene expression correlations [TBB07] utilize patterns of gene co-expression to detect gene interaction shifts. To the contrary of the NC method, correlation-based comparison considers only variance but not the way two variables interact, and would not be effective in telling the shift in interactions. Co-expressing genes can be considered as a simple model involving only two genes. Consider a network shown in the left pane of Fig. 1. A node X and a constant node C control the change of a node Y by a coefficient vector, (a, b), which are related to the correlation coefficient vector between X, C and Y. Two sets of observations $\{X^{(1)}, Y^{(1)}\}$ and $\{X^{(2)}, Y^{(2)}\}$ are obtained under different conditions which could cause changes in (a, b). In order to detect any change in (a, b), the NC method will first estimate two coefficient vectors from two data sets respectively, then compare the distance between the two estimated coefficient vectors with an experienced threshold. However, this numerical comparison may be unreliable, as illustrated by an example in the center and right panes of Fig. 1. Although the distance between the two estimated coefficient vectors is greater in the center pane than in the right pane, the data sets do not support such an interaction difference in the center pane as strong as in the right pane considering the uncertainty in the estimated coefficients. A threshold in between the two distances will lead to false negative differential interaction detection on the case in the right pane.

We establish a new paradigm of statistical comparison (SC) to detect interaction shifts in a biological network. An SC method can be considered a generalization of NC by extending the zero-variance assumption to non-zero. The significance takes the uncertainty into account that will be reported by a statistical heterogeneity test. The null hypothesis is that the interactions keep the same. A differential interaction in a network will be detected by rejecting the null hypothesis through analyzing the goodness-of-fit between a common interaction and several differential interactions. The proposed method will take uncertainty into account while focusing on the identification of interaction changes. In this paper, considering two comparable biological networks, we assume the true topology of the network is given but coefficients. We define an interaction by an ODE. A method is also given based on statistical comparison of multiple linear regression equations. Then, the performance of the proposed method will be illustrated by simulation studies under various noise levels, data sizes and interaction complexities. We also demonstrate our SC approach on a real biological network delineating cdc2-cyclin interactions in the cell division cycle.

2 Interactions in dynamical systems

We focus on detecting the interaction shifts in biological networks represented by dynamical system models (DSMs) composed of ODEs. We choose DSMs for two reasons: first, ODEs are widely used in kinetic models; second, biological model databases including a large number of DSMs, such as BioModels [NBB$^+$06], have been created, which can be used to test our methods. In gene regulatory network modeling, ODEs has been used to describe transcriptional kinetics [dLD09], where gene regulation is modeled by reaction-rate equations expressing the rate of a gene product as a function of concentrations of other gene products or metabolites in the system. The general mathematical form is

$$\frac{dx_i(t)}{dt} = f_i(\mathbf{x}(t), \boldsymbol{\beta}) \tag{1}$$

where $\mathbf{x}(t) = (x_0(t), x_1(t), ..., x_{N-1}(t))^\top$ is a vector of the concentrations of N variables at time t, $x_i(t)$ is a target variable which can represent the concentration of a gene product or a metabolite, $\boldsymbol{\beta}$ is a coefficient constant vector, and f_i is a linear combination of either linear (e.g. x_j) or nonlinear (e.g. quadratic x_j^2 or $x_j x_k$, or sigmoidal $\frac{x_j^2}{1+x_j^2}$) terms, with coefficients $\boldsymbol{\beta}$. The pair-wise linear correlation model is a special case of the above model as $0 = \beta_0 + x_i(t) + \beta_1 x_j(t)$. Coefficient vector $\boldsymbol{\beta}$ in model f_i can be estimated using multiple linear regression. We also refer to this estimation process as reconstruction and estimated coefficient vector as $\hat{\boldsymbol{\beta}}$.

In a pair of *differential* interactions for a variable, the two coefficient vectors, $\boldsymbol{\beta}^{(1)}$ and $\boldsymbol{\beta}^{(2)}$, differ from each other under two experimental conditions. Take Fig. 1 as an example. The values of node X, Y, and C can be considered gene concentrations. The rate of change in gene Y is regulated by X and C through a coefficient vector $\boldsymbol{\beta} = (a, b)$. Thus, any change in the values of (a, b) implies an interaction shift.

3 Detecting differential interactions via heterogeneity tests

We introduce the SC method and compare it with the more intuitive NC method. The NC method identifies interaction shifts in biological networks by numerically comparing with a threshold the distance between estimated coefficient vectors of individually reconstructed models based on several comparative data sets. On the other hand, the SC approach tests the interaction shifts by analyzing the goodness-of-fit of individual models (together called a heterogenous model) and a pooled model (a homogenous model) which is assumed to produce all data sets. A p-value as the significance will be reported by SC method finally.

How two methods work will be introduced by using the ODEs described in Eq. (1). Let matrix $\mathbf{X} = (\mathbf{x}[0]^\top, \mathbf{x}[1]^\top, ..., \mathbf{x}[T-1]^\top)$ be one observation set from T discrete time points. The concentration change rate, $y[t]$, of an interested variable i at discrete time t,

$$y[t] = \frac{dx_i[t]}{dt} \tag{2}$$

will be obtained from observations by using a smoothing spline technique in this paper. Let a vector $\mathbf{Y} = (y[0], y[1], ..., y[T-1])^\top$ represent one derivative set of the interested element at T time points. Assume two sets of the concentration observations in a network under different conditions are obtained, $\{\mathbf{X}^{(1)}, \mathbf{Y}^{(1)}\}$ and $\{\mathbf{X}^{(2)}, \mathbf{Y}^{(2)}\}$, respectively. The comparative methods can be utilized to check if these two sets come from two differential interactions while all observations contain the noise.

3.1 The numerical comparison approach

The NC method will produce a score based on the distance between individual models. After using the model reconstruction method to obtain $\hat{\boldsymbol{\beta}}^{(1)}$ and $\hat{\boldsymbol{\beta}}^{(2)}$ from two observation sets, respectively, the score will be calculated by

$$Score_R = \sqrt{(\hat{\boldsymbol{\beta}}^{(1)} - \hat{\boldsymbol{\beta}}^{(2)})^\top (\hat{\boldsymbol{\beta}}^{(1)} - \hat{\boldsymbol{\beta}}^{(2)})} \tag{3}$$

Based on the result of comparing this score with an experienced threshold, a differential interaction will be identified while the calculated score is larger.

There are several drawbacks of this method. Take Fig. 1 as an example, a large distance between estimated coefficients is not always associated with a difference between the true coefficients, as noise can distort the estimated difference. Furthermore, if the scales within a coefficient vector are not the same, normalization has to be applied as shown in Eq. (7).

3.2 The statistical comparison approach

We propose the SC method to identify interaction shifts in dynamic biological networks, based on a statistical heterogeneity test to compare model coefficients in two multiple linear regression models. The method detects the interaction shifts by analyzing the goodness-of-fit of a heterogenous interaction model versus a homogenous interaction model.

We formulate differential interaction detection as a statistical inference problem and obtain the best estimators first. Considering two sets of observations, $\{\mathbf{X}^{(1)},\mathbf{Y}^{(1)}\}$ and $\{\mathbf{X}^{(2)},\mathbf{Y}^{(2)}\}$, we assume the best model estimators for them are $\hat{\boldsymbol{\beta}}^{(1)}$ and $\hat{\boldsymbol{\beta}}^{(2)}$ respectively, which together form a heterogenous model (with a complexity of $P_{he} = \sum_{j=1}^{2} dim(\hat{\boldsymbol{\beta}}^{(j)})$). Under the null hypothesis that two data sets are from a single homogenous interaction model (with a complexity of P_{ho}), its best estimator is $\hat{\boldsymbol{\beta}}_{ho}$ ($P_{ho} = dim(\hat{\boldsymbol{\beta}}_{ho})$) which is calculated based on the pooled data set.

A test is now presented to test the null hypothesis of non-interaction-shift by comparing the performance of two models, though their modeling residuals respectively

$$R_{he} = \sum_{j=1}^{2} (\hat{\mathbf{Y}}^{(j)} - \mathbf{Y}^{(j)})^{\top} (\hat{\mathbf{Y}}^{(j)} - \mathbf{Y}^{(j)}),\ R_{ho} = \sum_{j=1}^{2} (\hat{\mathbf{Y}}_{ho}^{(j)} - \mathbf{Y}^{(j)})^{\top} (\hat{\mathbf{Y}}_{ho}^{(j)} - \mathbf{Y}^{(j)}) \quad (4)$$

where first derivatives $\hat{\mathbf{Y}}^{(j)} = f_i(\mathbf{X}^{(j)}, \hat{\boldsymbol{\beta}}^{(j)})$ is estimated by the heterogenous model with the jth data set and model coefficients; one element in $\mathbf{Y}^{(j)}$ which is obtained from observation directly is defined in Eq. (2); $\hat{\mathbf{Y}}_{ho}^{(j)} = f_i(\mathbf{X}^{(j)}, \hat{\boldsymbol{\beta}}_{ho})$ is obtained by the homogenous model. We notice that heterogenous models become a homogenous one when $\hat{\boldsymbol{\beta}}^{(1)} = \hat{\boldsymbol{\beta}}^{(2)}$ implying that the homogenous model is nested within the heterogenous one ($P_{ho} \leq P_{he}$). While the model complexities are taken into account, the proportion of the performance improvement achieved by heterogenous model versus its own performance can be inspected by using the ratio

$$F = \frac{(R_{ho} - R_{he})/df_1}{R_{he}/df_2} \quad (5)$$

where $df_1 = P_{he} - P_{ho}$ and $df_2 = T - P_{he}$. Under the null hypothesis, which is that both data sets are from a homogenous interaction model, the test statistic F follows an F-distribution with df_1 and df_2 degrees of freedom while the data size is asymptotic [Zar98]. If the test size α is given, using the above F-test, we can determine if two data sets arise from differential interactions. The F value is considered as the score of SC method. The significance level (p-value) in the application could be reported after obtaining the distribution of the test statistic by permutation when the sample size is small. We also point out that this method already works for comparing two additive nonlinear ODE models and can be extended to identify interaction shifts under more than two conditions.

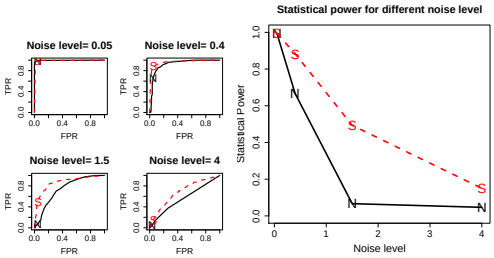


Figure 2: ROC and power advantage of the SC over the NC method under various noise levels. Left: ROC curves under four noise levels; Right: Power as a function of noise level. Solid curves marked by "N" represent NC, while the dashed ones marked by "S" represent SC. The noise level shown is the standard deviation of the noise.

4 Performance evaluation on simulated and real biological networks

To compare the performance of the NC and SC methods, we use the receiver operating characteristic (ROC) curve. The ROC curve is a graphical plot of the true positive rate (TPR) vs. the false positive rate (FPR) for a binary classifier as its discrimination threshold is varied. TPR is the detected fraction of all true differential interactions; while FPR is the fraction of all true non-differential interactions, that are incorrectly announced, also known as Type I error. The statistical power which is TPR at $FPR = 0.05$, is a function of population parameters, including but not limited to noise level, sample size, and the complexity of the classifier. A classifier whose ROC curve is closer to the top left corner has better performance, while the one that has an ROC of a diagonal line from (0,0) to (1,1) is equivalent to random guessing. One can also quantify the area under an ROC curve - a larger area indicates a better performance.

4.1 Simulation studies on the statistical power

We generated ROC curves for the SC and NC methods via a simulation study. Two types of ODE pairs were created randomly: the first type contains two identical ODEs representing conserved interactions; the second type contains two different ODEs representing differential interactions. Three coefficient vectors of dimension N+1 or complexity N (number of independent variables) were randomly sampled from uniform distribution from -5 to 5. The 1^{st} coefficient vector is shared by the pair of identical ODEs; the 2^{nd} and 3^{rd} are used for the pair of different ODEs. We randomly generated 300 ODE pairs for each type. For each ODE, T observations for each independent variable on the right hand side of Eq. (1) were sampled from the uniform distribution from -10 to 10; the left hand-side's first derivative was calculated directly from the ODE. Trajectories were simulated from the ODEs using T observations and first derivatives. Additive noise of zero-mean normal distributions is applied repeatedly to obtain noisy replicates of the trajectories. The SC and NC methods were applied to each pair of data sets to detect differential in-

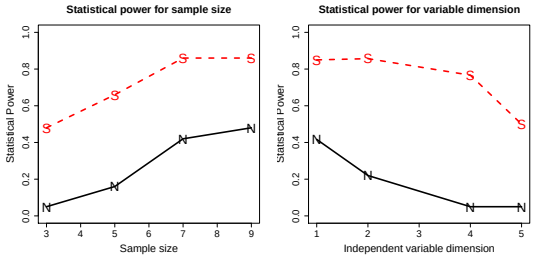


Figure 3: Power advantage of the SC over the NC method under various sample size and interaction complexity. Left: Statistical Power as a function of the sample size; Right: Statistical Power as a function of independent variable dimension. The solid curves marked by "N" represent NC, while the dashed ones marked by "S" represent SC.

teractions. Detection results on those pairs of data sets from the identical/different ODE pair were used to compute the FPR/TPR for plotting the ROC curve.

The performance of NC and SC methods under different noise levels are given in Fig. 2 by setting $N = 1$, $T = 3$ and $\sigma = (0.05, 0.4, 1.5, 4)$. The ROC curves are displayed in the left pane of Fig. 2. Both methods had good performance when the noise was low, while when the noise was high neither had any useful result. It is evident that the SC had consistently better performance under the intermediate noise levels. The statistical power as a function of noise level, shown in the right pane of Fig. 2, is another way to visualize the SC advantage of the TPR at $FPR = 0.05$.

The SC method also achieved a better performance on sample sizes and variable dimensions than the NC method, as illustrated in Fig. 3. We obtained the statistical power curves of the sample size, shown in the left pane of Fig. 3, by setting $\sigma = 1.5$ and $N = 1$. The statistical power curves of the independent variable dimension, shown in right pane of Fig. 3, were obtained by setting $\sigma = 1.5$ and $T = 7$. The statistical power gain of SC over NC in the two situations is up to 50% and 70%, respectively - an extraordinary advantage.

4.2 Differential interactions between cdc2 and cyclin in a cell division cycle model

As difference in trajectories under comparative conditions is insufficient to imply mechanism shifts in a dynamic biological network, one must examine the interactions at each node and check if any of them have changed in their coefficients. Thus we examine the heterogeneity of interactions at each node one by one in the network. We use a dynamical system model (Fig. 4) of cdc2-cyclin interaction in the cell division cycle [Tys91] to illustrate the performance of the two comparison methods on detecting mechanism shifts in the network. The cdc2-cyclin dynamical system model consists of six kinetic equations, shown in Table 1. Following recommendations for coefficient values in [Tys91], we set $k_1[aa]/[CT] = 0.015min^{-1}$, $k_2 = 0$, $k_3[CT] = 200min^{-1}$, $k_4' = 0.018min^{-1}$,

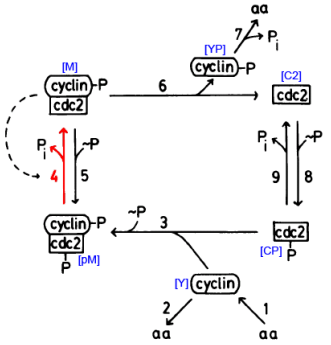


Figure 4: The cdc2-cyclin interaction dynamic network involved in the cell division cycle [Tys91]. The variable names used in the dynamical system model (Table 1) are marked next to the proteins or protein complexes they represent. The network shifts in the interaction change marked by #4.

$d[C2]/dt$	=	$k_6[M] - k_8[\sim P][C2] + k_9[CP]$
$d[CP]/dt$	=	$-k_3[CP][Y] + k_8[\sim P][C2] - k_9[CP]$
$d[pM]/dt$	=	$k_3[CP][Y] - k'_4[pM] - k_4[pM]([M]/[CT])^2$
$d[M]/dt$	=	$k'_4[pM] + k_4[pM]([M]/[CT])^2 - k_5[\sim P][M]$ $-k_6[M]$
$d[Y]/dt$	=	$k_1[aa] - k_2[Y] - k_3[CP][Y]$
$d[YP]/dt$	=	$k_6[M] - k_7[YP]$

Table 1: ODEs governing the cdc2-cyclin interaction in the cell division cycle [Tys91]: t, time; k_i, rate constant; aa, amino acids. The concentrations $[aa]$ and $[\sim P]$ are assumed to be constant. Variable $[C2]$ is for cdc2, $[CP]$ for cdc2-P, $[pM]$ for preMPF=P-cyclin-cdc2-P, $[M]$ for active MPF (P-cyclin-cdc2), $[Y]$ for cyclin and $[YP]$ for cyclin-P, and $[CT]$ for total cdc2.

$k_5[\sim P] = 0$, $k_6 = 1min^{-1}$, $k_7 = 0.6min^{-1}$, $k_8[\sim P] = 1000000min^{-1}$ and $k_9 = 1000min^{-1}$, where $[CT]$ was assumed to be a constant of 1. In this study, we perturb the coefficient k_4, a rate constant associated with the autocatalytic activation of MPF by dephosphorylation of the cdc2 subunit [Tys91], marked as reaction 4 in Fig. 4. After cell division becomes growth controlled, $k_4 > 150min^{-1}$, MPF enters the oscillation domain in which it alternates between active and inactive forms with a period of 35 min, roughly the cell cycle length in early frog embryos [Tys91]. While $k_4 < 100min^{-1}$, MPF, being maintained in inactive forms, goes into the excitable domain (as in the resting phase of non-proliferating somatic cells). As cells grow, k_4 increases (activator accumulates) and drives the regulatory system into the oscillation domain. The subsequent burst of MPF activity triggers mitosis, causes k_4 to decrease (activator degrades), and brings the regulatory system back into the excitable domain (steady-state behavior). We use the comparison methods to detect changed interactions due to k_4, which implicates two differential interactions for preMPF and active MPF. The remaining interactions for other four proteins are conserved.

As the observed trajectories for the 6 involved proteins are distinctive in the two domains of the cell division cycle, differential gene expression analysis would report statistically significant changes in all proteins. We applied the NC and SC methods to detect differential interactions for $[pM]$ and $[M]$ as well as conserved interactions of other proteins. k_4 was set to be 180 min^{-1} in the oscillation domain, while in the excitable domain, k_4 was randomly chosen from a uniform distribution from 70 to 80. After 20 observations of 40-min long trajectories were obtained for each domain, noises were added three times to generate replicates. A smoothing spline technique was utilized to obtain the first derivatives for each variable from the noisy observations. Assuming that the forms of kinetic equations were known but not the coefficients, both methods were applied to detect differ-

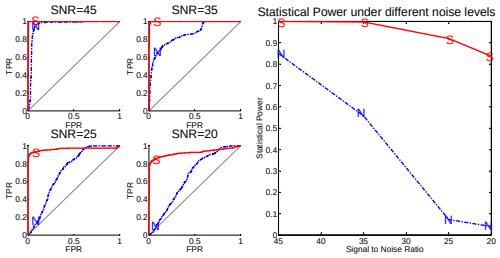


Figure 5: The advantage in ROC curves and statistical power of the SC (marked by "S") versus the NC methods (marked by "N") to detect differential interactions in the cdc2-cyclin cell division cycle model. Left: The ROC curves under different noise levels. Right: The statistical power of both methods as a function of the noise level.

ential interactions of each protein in the network with varying thresholds under four noise levels. Then we compared the detection results with the two true differential interactions at $[pM]$ and $[M]$ (true positives) and four true conserved interactions at $[C2]$, $[CP]$, $[Y]$ and $[YP]$ (true negatives) to compute the overall TPR and FPR at each noise level.

The noise level in this study is represented by signal-to-noise ratio (SNR), defined as ten times $\log_{10}$ of the sum of squares of the signal divided by the sum of squares of the noise. When the data sets contain replicates, the SNR can be estimated by

$$SNR = 10\log_{10}\left(\frac{\sum_{i=1}^{K}\bar{O}_{i\cdot}}{\sum_{i=1}^{K}(\sum_{j=1}^{M_i}(O_{ij}-\bar{O}_{i\cdot})/M_i)}\right)^2 \tag{6}$$

where O_{ij} means the j-th replicate of i-th observation, among a total of K observations, the i-th observation contains M_i replicates, and $\bar{O}_{i\cdot} = \frac{1}{M_i}\sum_{j=1}^{M_i} O_{ij}$.

As the scales of coefficients were different, we modified the NC score from Eq. (3) to

$$Score_R = \max_{i=1,\ldots,N}(|\hat{\beta}_i^{(1)} - \hat{\beta}_i^{(2)}|/(2\max(\hat{\beta}_i^{(1)}, \hat{\beta}_i^{(2)}))) \tag{7}$$

where N is the dimension of $\hat{\boldsymbol{\beta}}$.

The SC method achieved consistently and significantly better performance than the NC method on differential interaction detection in networks. From Fig. 5, we can see when the noise was low ($SNR = 45dB$), both methods achieved good performance. However, when the noise strength increased ($SNR = 35, 25, 20dB$), the performance of NC method dropped very quickly, while the SC maintained above 80% power at a type I error of 0.05.

5 Discussion

We have proposed an SC method to identify differential interactions in nonlinear dynamic biological networks, based on a statistical test to compare multiple linear regression equations. In addition to be able to announce two networks are different, the method can also

detect which node in the network has experienced an interaction shift. Our simulation studies demonstrated that the performance of SC approach is substantially superior to the NC method under various noise levels, sample sizes, and interaction complexities. The cdc2-cyclin interaction cell division cycle model was also used to test the proposed method on real biological networks and our method achieved much improved identification accuracy of differential interactions over the NC method. The SC method is much more sensitive to detect consistent interaction changes while keeping a low Type I error.

Our comparative modeling is capable of generating two lists: one is a list of the genes whose regulatory relationships from a common concerting theme under the comparative conditions, such as $[C2]$, $[CP]$, $[Y]$ and $[YP]$ in the cell cycle model; the other is the list of genes whose regulatory relationships consistently demonstrated distinctive signatures under the comparative conditions, such as $[pM]$ and $[M]$ in the cell cycle model.

We are working on applying the proposed method on studying differential gene interactions between embryonic and postnatal stages in mouse cerebellar development. We anticipate our approach widely applicable to many comparative experimental designs in life science research.

References

[BBAIdB07] M. Bansal, V. Belcastro, A. Ambesi-Impiombato, and D. di Bernardo. How to infer gene networks from expression profiles. *Mol Syst Biol.*, 3(78), 2007.

[Bon08] R. Bonneau. Learning biological networks: from modules to dynamics. *Nature Chemical Biology*, 4(11):658–664, Nov. 2008.

[dLD09] S. Ben-Tabou de Leon and E. H. Davidson. Modeling the dynamics of transcriptional gene regulatory networks for animal development. *Developmental Biology*, 325(2):317–328, Jan. 2009.

[KWKK08] H. A. Kestler, C. Wawra, B. Kracher, and M. Kühl. Network Modeling of Signal Transduction: Establishing the Global View. *Bioassays*, 30(11-12):1110–1125, Nov. 2008.

[NBB+06] N. Le Novre, B. Bornstein, A. Broicher, M. Courtot, M. Donizelli, H Dharuri, L. Li, H Sauro, M. Schilstra, B. Shapiro, J. L. Snoep, and M. Hucka. BioModels Database: a free, centralized database of curated, published, quantitative kinetic models of biochemical and cellular systems. *Nucleic Acids Res*, 34(Database Issue):D689–D691, Jan. 2006.

[NS04] M. Neuhäuser and R. Senske. The Baumgartner-Weiβ-Schindler test for the detection of differentially expressed genes in replicated microarray experiments. *Bioinformatics*, 20(18):3553–3564, 2004.

[TBB07] I. Tirosh, Y. Bilu, and N. Barkai. Comparative biology: beyond sequence analysis. *Current Opinion in Biotechnology*, 18(4):371–377, Aug. 2007.

[Tys91] J. J. Tyson. Modeling the cell division cycle: cdc2 and cyclin interactions. *Proceedings of the National Academy of Sciences*, 88(16):7328–7332, Aug. 1991.

[Zar98] J. H. Zar. *Biostatistical Analysis*. Prentice Hall, 4^{th} edition, Oct. 1998.

Traceable Analysis of Multiple-Stage Mass Spectra through Precursor-Product Annotations

Hisayuki Horai[1,*], Masanori Arita[1,2,3,*], Yuya Ojima[1], Yoshito Nihei[1], Shigehiko Kanaya[3,4] and Takaaki Nishioka[1]

[1]Institute for Advanced Biosciences, Keio University, Tsuruoka 997-0035, Japan
[2]Department of Computational Biology, Graduate School of Frontier Sciences, The University of Tokyo, Kashiwa 277-8561, Japan
[3]RIKEN Plant Science Center, Yokohama 230-0045, Japan
[4]Laboratory of Comparative Genomics, Graduate School of Information Science, Nara Institute of Science and Technology, Ikoma 630-0192, Japan
[*]Both authors contributed equally

Abstract: We present a wiki-based interface for multiple-stage mass spectra with molecular structures and their physicochemical properties. Spectra for 453 metabolites were measured on QqTOF-MS^n and their ion peaks were annotated with consideration of fragmentation patterns, especially bond cleavages. The resulting information was classified on wiki pages, where related molecular formulas and their relationships were likewise accumulated. Each page is rendered with search operation(s) using formulas as keys, and related information is automatically updated as database contents increase. Our data management model allows internet beginners to collaboratively input and organize information in a multi-user environment. The system, with links to our MassBank database (`http://massbank.jp/`), is available at `http://metabolomics.jp/wiki/Index:MassBank`.

1 Introduction

Metabolomics has become a standard technology in analyzing natural products [VBNS+07], and metabolite identification from mass spectra (MS) is a much investigated research topic. Identification from electrospray-ionization (ESI) spectra has been hampered by practical problems, however, because metabolites share similar molecular structures and physicochemical properties. First, there is no comprehensive database for ESI-MS. Fragmentation pattern of ESI has been considered machine-type dependent, and few research groups have attempted to accumulate and provide freely downloadable spectral information [Nat08, SRL+08, WKG+09]. Second, fragmentation rules have not been well understood compared to what has been accomplished in electron-ionization (EI) MS [McL59]. To overcome these difficulties, we have designed and implemented a distributed database called MassBank (`http://massbank.jp/`) for ESI spectra. Over 10 institutions joined our consortium and share spectra as well as data management systems. In this short article, we introduce a recent activity on our wiki-based interface to MassBank. Hereafter, MS

stands for a deconvoluted set of ion peaks p whose m/z (mass to charge ratio) and scaled intensity (from 0 to 1000) can be accessed by functions $\mathrm{mass}(p)$ and $\mathrm{intensity}(p)$, respectively. We use these functions rather informally for elements other than ion peaks when the context is unambiguous. We also use the word 'mass' to refer to m/z hereafter.

2 Data Acquisition

2.1 Statistics of Mass Spectral Data

As of June 2009, twelve laboratories contribute their mass spectra to MassBank (see `http://massbank.jp/en/published.html`). The total number of ESI spectra is $> 10,000$ for over $1,500$ molecules with overlap. All records are accessible for free, and software programs are also available under the GNU General Public License. Because the overview of the database including supported search methods will be presented elsewhere, we focus on the analysis of precursor-product ion relationships here.

2.2 Peak Annotation

The current study used spectra of 453 metabolite standards measured on QqTOF-MS^n (Applied Biosystems Japan, Tokyo) at Keio University. Peaks were annotated with consideration of fragmentation patterns, especially bond cleavage. Let us assume that, for a standard compound M, a set of spectral peaks

$$\{P_M | \forall p \in P_M \ \mathrm{intensity}(p) > 5\}$$

is obtained. Our annotation process consisted of the following steps.

1. For each ion peak $p \in P_M$, find a molecular formula f_p that is a subset of molecular composition of M and whose mass is within 50 ppm from the $\mathrm{mass}(p)$. If not, remove p from P_M.
2. For each remaining $p \in P_M$, assign a connected molecular substructure m_p of M that corresponds to f_p. If such a structure is not found, then remove p from P_M.
3. For each remaining $p \in P_M$, find all peaks $q \in P_M$ such that its structure m_q can be obtained by a single fragmentation step (i.e. cleaving up to 2 bonds) from m_p. Output all pairs (p, q). This step tries to list precursor-product pairs only, not an arbitrary pair of fragments.

Assignments were manually checked using two commercial software programs: Mass Frontier (Thermo Fisher Scientific Inc., Waltham MA, USA) and ACD/MS Fragmenter (Fujitsu Inc., Kawasaki, Japan). Through this process, 1,483 different molecular formulas

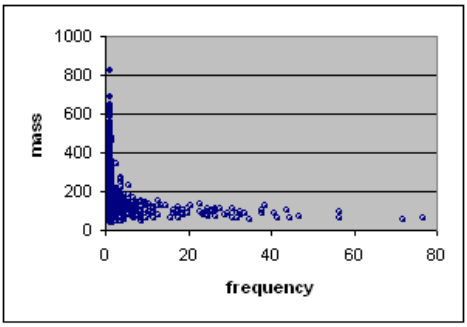


Figure 1: Statistics of Precursor-Product Ions

were identified, among which 5,557 precursor-product ion pairs were assigned. Note that the vast majority of peaks were left unannotated: we could annotate only 3,925 out of 130,246 peaks (3%, 9 peaks per spectra on average). Among the assigned molecular formulas, as much as 985 appeared only once. Most frequently appearing ions (and their frequency in parentheses) were C_6H_5 (100), C_5H_5 (80), C_3H_6N (71), and C_4H_3 (70), respectively. Not surprisingly, many ions were unsaturated fragments of C and H only and the mass of most frequent molecular formulas were under 200. Details of annotation results will be presented elsewhere (Ojima *et al.* in preparation).

3 Results and Discussion

3.1 Statistics of ESI Fragmentation

The notable character of annotated ions is the absence of clear correlation between frequency and mass (Figure 1). Usually MS contain more peaks of smaller mass, and such peaks are not informative for metabolite identification. However, except for some highly frequent masses (lower right dots in Figure 1), annotated ions showed low frequency and were distributed almost evenly up to mass 600. This indicates that annotated ions of smaller mass are equally as informative in structure prediction as those of larger mass. It must be noted that our annotation process is easier than the identification of metabolites [RSGB09]. Since we know the molecular structure in advance, we only need to traverse its possible, connected substructures (we did not consider a coupling of isolated fragments) in steps in Section 2.2. Although the enumeration problem of such substructures is NP-hard [HRM$^+$08], it is feasible for small metabolites under our strict condition.

3.2 Similarity Measure using Fragment Ions

The main purpose of annotation is to use it for metabolite identification from spectra in a future. For a fragment ion p to be used for identification, its frequency $\mathrm{freq}(p)$ should not be high. To identify informative ions, Shannon's information content $\mathrm{H}(p) = -\log(\mathrm{freq}(p))$ was used. Then, the similarity of two molecular structures M_1 and M_2 was defined as $\sum_{p \in P_{M_1} \cap P_{M_2}} \mathrm{H}(p)$. Note that all ions were equally weighted regardless of their mass by considering the result of Section 3.1.

3.3 Wiki-based Interface for Spectral Information

The information source of the analyses is essentially molecular formulas of product-precursor pairs, and all analyses are straightforward. Our novelty is not the analysis contents but the accessibility of processes and results on MediaWiki pages, i.e. traceability of research [AS08]. In other words, all operations are performed at the user-level on the wiki-based system and any user can reproduce, verify, and edit details just like editing a Wikipedia article.

Fragmentations observed in ESI-MS^n are a quite different type of chemical reactions from those observed in EI-MS; all the ions produced in EI-MS have odd-number electrons ("radical ions"), whereas those in ESI-MS have even-number electrons. Empirical rules that have been accumulated for the fragmentations in EI-MS are never applicable to the degradation reactions in ESI-MS^n. Only a few empirical rules are known in ESI-MS^n [Nak02], and we need to accumulate more rules on its degradation reactions. To provide a web-based forum of more chemical discussions among the mass spectral research communities, we provide a wiki-based platform linked with MassBank. Since the wiki part is used for annotation and discussion, not for actual spectra, default page contents should be as succinct as possible. For this reason, our wiki pages contain minimum possible information for drawing fragmentation scheme.

Let us explain an example at

`http://metabolomics.jp/wiki/MassBank:KOX00284`[1].

The page source is the simplest: identified molecular formulas and their precursor-product relationships only. At the time of page access, the minimum information is processed into a precursor-product table as its display, and the search for related pages is performed on demand. This molecule is Glycolate (MassBank ID: KOX00284), and its structural neighbor, Taurocolate (KOX00601), is automatically detected through the similarity of fragment ions. All results are always up-to-date even if other users add or remove data pages asynchronously. By checking such information, users can add comments and questions on precursor-product relationships as ordinary texts on wikis. Its advantage is obvious for a

[1] Currently, related pages are password-protected. To access, please login using the name "MassBank" and password "GCB2009".

collaborative project like our MassBank because page edit is open to all registered contributors.

The difference from conventional approaches such as Semantic Wiki is its simplicity [Ari09, HBB+09]. Data are plainly organized in a tabular form, and are exempted from site-specific predicates or manual addition of page links. Users' task is drastically alleviated since formatting and linking can be delegated to an embedded Lua programming language [Ier06], whose programs are also written inside wiki pages. Its computational power is restricted by running time and by closed I/O libraries to avoid web vandalism. Using Lua functionality, pages can be designed to minimize redundancy of data.

4 Conclusion

We implemented spectral annotation of precursor-product relationships on a MediaWiki based platform. All pages and Lua programs can be managed at the user-level, and this consequently guarantees traceability of research. Wiki users are also encouraged to leave references and traces of thinking in the annotation so that fragmentation rules can be later summarized from input information. A login account can be obtained on request to `massbank@iab.keio.ac.jp`.

Acknowledgments

This work is supported by JST-BIRD and Grant-in-Aid for Scientific Research on Priority Areas "Systems Genomics" from the Ministry of Education, Culture, Sports, Science and Technology, Japan. Authors also thank anonymous referees for their valuable comments.

References

[Ari09] Masanori Arita. A pitfall of wiki solution for biological databases. *Briefings in Bioinformatics*, 10(3):295–296, 2009.

[AS08] Masanori Arita and Kazuhiro Suwa. Search extension transforms Wiki into a relational system: A case for flavonoid metabolite database. *BioData Mining*, 1(1):7, 2008.

[HBB+09] Robert Hoehndorf, Joshua Bacher, Michael Backhaus, Sergio Gregorio, Frank Loebe, Kay Prufer, Alexandr Uciteli, Johann Visagie, Heinrich Herre, and Janet Kelso. BOWiki: an ontology-based wiki for annotation of data and integration of knowledge in biology. *BMC Bioinformatics*, 10(Suppl 5):S5, 2009.

[HRM+08] Markus Heinonen, Ari Rantanen, Taneli Mielikäinen, Juha Kokkonen, Jari Kiuru, Raimo A. Ketola, and Juho Rousu. FiD: a software for ab initio structural identification of product ions from tandem mass spectrometric data. *Rapid Communications in Mass Spectrometry*, 22(19):3043–3052, 2008.

[Ier06] Roberto Ierusalimschy. *Programming in Lua*. Lua.org, 2 edition, 2006.

[McL59] F. W. McLafferty. Mass Spectrometric Analysis. Molecular Rearrangements. *Analytical Chemistry*, 31(1):82–87, 1959.

[Nak02] H. Nakata. A Rule to account for mass shifts in fragmentations of even-electron organic ions in mass spectrometry. *Journal of the Mass Spectrometry Society of Japan*, 50(4):173–188, 2002.

[Nat08] National Institute of Standards and Technology. NIST Chemistry WebBook, 2008.

[RSGB09] Simon Rogers, Richard A. Scheltema, Mark Girolami, and Rainer Breitling. Probabilistic assignment of formulas to mass peaks in metabolomics experiments. *Bioinformatics*, 25(4):512–518, 2009.

[SRL+08] T. R. Sana, J. C. Roark, X. Li, K. Waddell, and S. M. Fischer. Molecular formula and METLIN Personal Metabolite Database matching applied to the identification of compounds generated by LC/TOF-MS. *Journal of Biomolecular Techniques*, 19(4):258–266, 2008.

[VBNS+07] Silas G. Villas-Boas, Jens Nielsen, Jorn Smedsgaard, Michael A. E. Hansen, and Ute Roessner-Tunali. *Metabolome Analysis: An Introduction*. Wiley - Interscience Series on Mass Spectrometry. Wiley-Interscience, 1 edition, 2007.

[WKG+09] David S. Wishart, Craig Knox, An Chi Guo, Roman Eisner, Nelson Young, Bijaya Gautam, David D. Hau, Nick Psychogios, Edison Dong, Souhaila Bouatra, Rupasri Mandal, Igor Sinelnikov, Jianguo Xia, Leslie Jia, Joseph A. Cruz, Emilia Lim, Constance A. Sobsey, Savita Shrivastava, Paul Huang, Philip Liu, Lydia Fang, Jun Peng, Ryan Fradette, Dean Cheng, Dan Tzur, Melisa Clements, Avalyn Lewis, Andrea De Souza, Azaret Zuniga, Margot Dawe, Yeping Xiong, Derrick Clive, Russ Greiner, Alsu Nazyrova, Rustem Shaykhutdinov, Liang Li, Hans J. Vogel, and Ian Forsythe. HMDB: a knowledgebase for the human metabolome. *Nucleic Acids Research*, 37(suppl_1):D603–610, 2009.

EFMEvolver: Computing elementary flux modes in genome-scale metabolic networks

Christoph Kaleta*, Luís Filipe de Figueiredo*, Jörn Behre, and Stefan Schuster†

Department of Bioinformatics, Friedrich Schiller University Jena, Ernst-Abbe-Platz 2, D-07743 Jena, Germany

Abstract: Elementary flux mode analysis (EFM analysis) is an important method in the study of biochemical pathways. However, the computation of EFMs is limited to small and medium size metabolic networks due to a combinatorial explosion in their number in larger networks. Additionally, the existing tools to compute EFMs require to enumerate all EFMs before selecting those of interest. The method presented here extends EFM analysis to genome-scale models. Instead of computing the entire set of EFMs an optimization problem is used to determine a single EFM. Coupled with a genetic algorithm (GA) this allows to explore the solution space and determine specific EFMs of interest. Applied to a network in which the set of EFMs is known our method was able to find all EFMs in two cases and in another case almost the entire set before aborted. Furthermore, we determined the parts of three metabolic networks that can be used to produce particular amino acids and found that these parts correspond to significant portions of the entire networks.
Availability: Source code and an executable are available upon request.

1 Introduction

In the post-genomic era, the analysis of metabolic networks is essential for molecular biology. These networks are complex and the subdivision of a network into pathways makes the analysis more comprehensive. However, the focus only on specific classically known pathways can conceal the view on the actual metabolic capabilities of an organism [KdFS09]. Thus, the construction of genome-scale metabolic networks that model the entire metabolism of organisms has come to importance [FP08].

A method that has been used to comprehensively studying pathways in metabolic networks is elementary flux mode analysis [SDF99]. Elementary flux modes (EFMs) are a systematic definition of the biological concept of a pathway. They correspond to minimal sets of reactions that can perform at steady state [SDF99]. EFM analysis has already been used to study biochemical relevant metabolic pathways [CS04, dFSKF09], to study metabolic network properties such as fragility and robustness [SKB$^+$02, BWvK$^+$08], and to optimize microorganisms with respect to the production of a certain metabolite [TUS08]. However, EFM analysis has been limited to small and medium scale networks because the number of EFMs grows exponentially with the size of the network [KS02]. For instance, Yeung *et*

*Both authors contributed equally

†Corresponding author (stefan.schu@uni-jena.de)

al. [YTP07] estimated that the number of extreme pathways [SLP00], a subset of EFMs, is at the order of 10^{29} for a genome-scale model of human.

Due to this problem, alternative approaches for the identification of pathways based on graph theory have been proposed [RAS+05, CCWvH06, BK08]. These methods abstract from the metabolic network by converting it into a graph and consider only connected paths. While they operate efficiently in genome-scale metabolic networks, they bear the problem that a detected pathway does not automatically imply that a net-conversion of the source metabolite into a specific target metabolite is possible [PB08, dFSKF09].

Here we want to present a method that allows the enumeration of EFMs in genome-scale metabolic models. Starting from an initial pathway, the space of EFMs is explored using a genetic algorithm (GA). GAs have already been used in the analysis of metabolic networks to find combinations of gene knockouts that improve the production of a given metabolite [PRFN05]. We used benchmark models for EFM analysis to validate our new method and applied it to a study of amino-acid synthesis in genome-scale metabolic models.

2 Methods

The aim of our algorithm is, given a metabolic network and an input medium, to find all EFMs producing a certain metabolite. The employed strategy is based on the observation that gene knockouts can force an organism to use pathways alternative to those found under standard conditions. Thus, we are detecting EFMs by evolving a population in which each individual corresponds to a set of knockouts. However, instead of considering the knockouts of genes we here focus on the "knockout" of reactions. By searching for a specific EFM avoiding reactions that are knocked out and iterating over different sets of knockouts we are able to determine different EFMs.

2.1 Detecting a single EFM

A metabolic network comprising *m* metabolites and *n* reactions is defined by the *m*x*n* stoichiometric matrix $\mathbf{N}$. Each metabolite can be defined to be either internal or external. External metabolites differ to internal metabolites in that their concentration is assumed to be buffered by the system. Examples for such external metabolites are energy currency metabolites like ATP, NADH and FADH. Since their concentration is assumed to be constant they are not required to be balanced by an EFM.

To be an EFM, a flux $\mathbf{v} \in \mathbb{R}^n$ through a reaction network has to fulfill the following conditions: (1) steady-state condition, i.e., all internal metabolites are balanced; (2) irreversible reactions have positive fluxes; (3) non-decomposability of the enzyme set, i.e., the non-zero indices of one EFM cannot be a subset of the non-zero indices of another EFM. In our approach reversible reactions are decomposed into two irreversible reactions with opposite directions. Therefore, all fluxes have to be positive.

Given a set K of reactions to be knocked out and an index μ corresponding to a target reaction which produces a certain metabolite of interest, the optimization problem to compute an EFM can be formulated as a linear program by minimizing $\sum_{r=1}^{n} v_r$ subject to

$$\mathbf{N}\mathbf{v} = 0 \tag{1}$$

$$\mathbf{v} \geq 0 \tag{2}$$

$$v_\mu \geq 1 \tag{3}$$

$$\forall i \in K : \; v_i = 0 \tag{4}$$

Using eqs. 1 and 2 we only allow for a strictly positive flux $\mathbf{v}$ that obeys the steady-state condition. Eq. 3 forces the solution to have a positive flux through a given reaction which can be the outflow of the product of interest, i.e., if a solution exists, $\mathbf{v}$ produces the metabolite of interest. Eq. 4 guarantees that we only find a flux that does not use the reactions in K that are knocked out. By minimizing the overall flux and solving the linear program using the simplex algorithm [Sch98] we achieve that $\mathbf{v}$ corresponds to an EFM. This property of $\mathbf{v}$ will be shown in the following.

The solution space of the steady-state and the irreversibility condition (eqs. 1 and 2) in the space of possible fluxes $\mathbb{R}^n$ corresponds to a convex polyhedral cone $\mathcal{P}$ [GK04]. Since, we split reversible reactions, the extreme rays or spanning vectors of $\mathcal{P}$ correspond to the EFMs of the system. Furthermore, a knockout of a reaction only leads to the disappearance of some EFMs [SDF99]. Thus, for every K chosen, the cone is still spanned by EFMs and eq. 4 does not impact the property of the spanning vectors of $\mathcal{P}$ of being EFMs. Furthermore, eq. 3 cuts $\mathcal{P}$ with a hyperplane at $v_\mu = 1$ (Figure 1C). Since $\mathcal{P}$ is unbounded the edges of the solution space of eqs. 1 - 3 correspond to the intersection points between the EFMs defined by eqs. 1 as well as 2 and the hyperplane defined by $v_\mu = 1$. These points can each be written as the corresponding EFM multiplied with a scaling-factor. From linear programming it is known that the simplex algorithm used to solve such problems always returns a solution that can be found at the edges of the solution space [Sch98]. Thus, using the simplex algorithm and minimizing the objective function subject to eqs. 1 - 4 will always return an EFM.

In principle, the described linear program can find all EFMs by testing every possible set of knocked out reactions K. However, this is computationally inefficient and thus we will next outline an algorithm that allows to explore the space of EFMs more efficiently.

2.2 Genetic Algorithm

The aim of the GA is to test different sets of reactions to be knocked out in order to find all EFMs. Each such set of reactions corresponds to an individual. Each individual is represented by a binary genome $\mathbf{G}$ of length n, i.e., the number of reactions in the system. $G_i = 1$ indicates that reaction i can be used by that organism and $G_i = 0$ that this reaction is knocked out. From each genome an EFM can be derived by mapping $\mathbf{G}$ to the

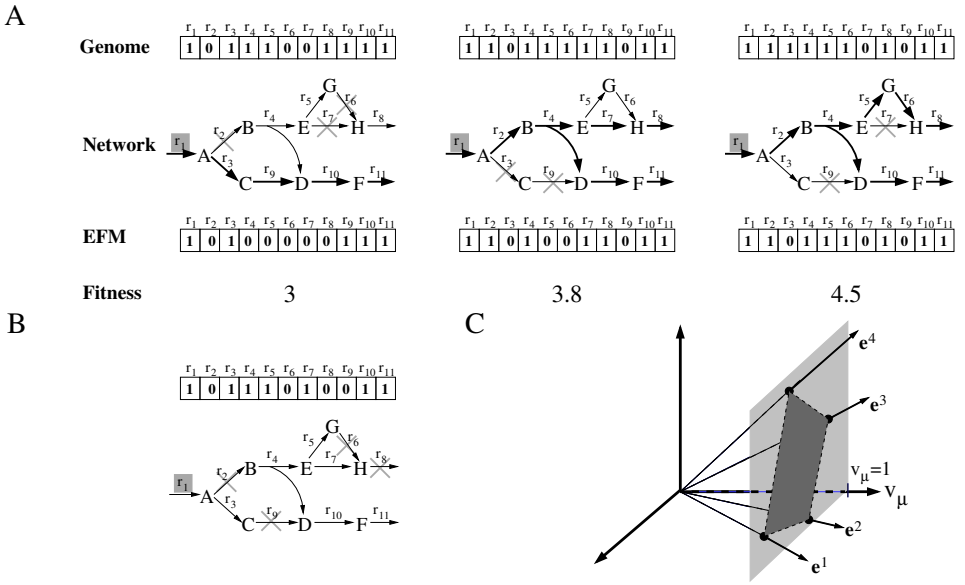


Figure 1: Scheme of the computation of EFMs. **A** Viable individuals. The target reaction $\mu = \mathrm{r}_1$ is shaded in gray. In the upper row the genome of each individual is given. The second row indicates the reactions knocked out in the model and the third row the EFM obtained from the linear program. Even though the EFM of the third individual is also a valid EFM satisfying eqs. 1 - 4 for the second individual it is not minimal since the sum of fluxes is higher. The fourth row gives the fitness of each individual for a population containing the three depicted genomes. **B** Individual for which no EFM can be found. **C** Three-dimensional solution space of eqs. 1 - 3 for 3 reactions (not shown). The solution space is defined by the intersection of the solution space of eqs. 1 and 2, spanned by the EFMs $\mathbf{e}^1$ to $\mathbf{e}^4$, and the half-space defined by eq. 2. Optimal solutions of the linear program can always be found in the edges of the solution space (black circles).

set of knocked out reactions K and solving the linear program described in the previous section. Thus, we can obtain an EFM associated to an individual (Figure 1A and 1B). Solving the linear program described in the last section we can only find a single EFM. In consequence, by specifying different sets of reactions that should not be used by an EFM, that is, by knocking them out, we can sample EFMs.

Central for each GA is the definition of a fitness function that returns a numerical value indicating the quality of an individual. In contrast to other approaches the aim of the GA described here is not to find an individual that is optimal in some sense, but to detect all possible EFMs in a metabolic network. Thus, we attribute higher fitness to individuals whose associated EFMs use reactions which are not frequent in the EFMs of the population. Given a population $\mathbf{G}^1$, ..., $\mathbf{G}^s$ of individuals and the associated EFMs $\mathbf{e}^1$, ..., $\mathbf{e}^s$ the

fitness $f(\mathbf{G}^k)$ of a particular individual $\mathbf{G}^k$ is defined by

$$f(\mathbf{G}^k) = \sum_{i=1}^{n} \frac{sign(e_i^k)}{\sum_{j=1}^{s} sign(e_i^j)} \tag{5}$$

with $sign(x)$ returning '1' if x is non-zero, i.e., if a reaction is used, and '0' otherwise.

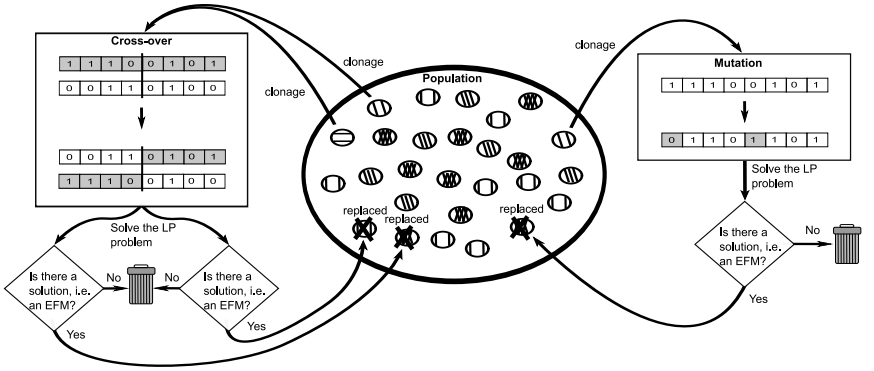


Figure 2: Setup of the GA. Individuals from the population are cloned and subsequently mutated or recombined. Afterward the viability of the individuals is tested by determining an EFM contained in them that uses the target reaction μ. If no such EFM is found, the individual is discarded. Otherwise it is reinserted by replacing a randomly chosen individual in the population.

For the GA we use the setup depicted in Figure 2. We assume a constant population size of s individuals and use two genetic operators: mutation and recombination. Before selecting individuals from the population it is decided whether a mutation or recombination should be performed. With probability $1 - p_{rec}$ one individual is mutated and with probability p_{rec} two individuals are recombined. To apply these operators, individuals are cloned from the population. By cloning we mean that an individual is selected and its genome copied creating a new individual. Thus, the original individual persists in the population. Given the individual fitness values $f(\mathbf{G}^1)$, ..., $f(\mathbf{G}^s)$ the probability of individual k to be cloned is proportional to its fitness:

$$P(\text{“Individual } k \text{ is cloned”}) = \frac{f(\mathbf{G}^k)}{\sum_{i=1}^{s} f(\mathbf{G}^i)} \tag{6}$$

During a mutation event, after cloning a single individual, each position in its genome is mutated with probability p_{mut}. Subsequently, it is tested whether the new individual is “viable” by determining the EFM associated to it. If such an EFM is found, the individual is re-inserted into the population by replacing a randomly chosen individual. Furthermore, the EFM that has been found is compared to all previously found EFMs and is added

to this set if it has not already been detected. If two individuals are recombined, they are first cloned and then the genomes are interchanged starting from a random position. Subsequently it is tested for both if they are viable, and, if this is the case, they are re-inserted replacing two randomly chosen individuals of the population. Thus, EFMs are detected as a side product of checking the viability of new individuals.

An important advantage of GAs is that they can be easily parallelized by the use of separate threads that mutate, recombine, and test individuals. Thus, the multi-processor architecture of modern desktop PCs is fully exploited.

3 Results

We applied our method to compute EFMs producing lysine, threonine, and arginine in two metabolic networks of *Escherichia coli* and one metabolic network of *Corynebacterium glutamicum*. Especially for the industrial production of lysine *C. glutamicum* is of importance [WBE06]. The first network of *E. coli* has been presented in [BWvK$^+$08]. It comprises 220 reactions and models amino acid metabolism. This network has the advantage that we can compute EFMs using Metatool [vKS06]. The second network represents a genome-scale model of *E. coli* metabolism and comprises 3558 reactions [FHR$^+$07]. The model of *C. glutamicum* contains 641 reactions and has been presented in [KN09]. In order to avoid side-pathways used for the balancing of co-factors and to provide an input medium we set the metabolites ammonium, AMP, ATP, CO_2, coenzyme A, glucose, NAD$^+$, NADH, NADP$^+$, NADPH, oxygen, protons, and inorganic ions to external status. As parameters for the computation we used a population size of $s = 100$ individuals, a mutation rate of $p_{mut} = 0.01$ per reaction and a probability of $p_{rec} = 0.3$ for recombination events. Computations were performed on an Intel® Core™2 Quad Q9300 machine with 4096 MB RAM running Linux Kernel 2.6.25 and Java Hotspot VM version 1.6.0. *Clp* version 1.0.6 from the COIN-OR project [LH03] has been used to solve the linear programs. An overview on the results is given in Table 1 and Figure 3.

As a first benchmark we tested to what extend our method can recover EFMs in a system in which they are already known. The model of [BWvK$^+$08] contains 3436 EFMs producing lysine, 444 EFMs producing threonine and 27450 EFMs producing arginine. We found all EFMs producing threonine and lysine after 491 s and 4821 s, respectively. For arginine we recovered 95.6% of all EFMs after a running time of 7200 s. In comparison, Metatool 5.1 took only 61 s to find all 65840 EFMs. However, a direct run-time comparison even to the currently fastest algorithm for the enumeration of EFMs presented in [TS08] does not bear much meaning since these methods in general only return the entire set of EFMs. This is not practicable in genome-scale networks since the number of EFMs exceeds by far current limitations in memory and processing power [YTP07]. An interesting behavior of the GA can be observed from these experiments. First, the time-course shows a kind of saturation when having found most of the EFMs. Furthermore, we observe phases in which only few new EFMs are found and sudden jumps in which the number increases rapidly as in the case of threonine in the model of amino acid metabolism at $t = 320$ s. While this particular behavior is also observable in the case of lysine in the model of *C. glutamicum*, a saturation

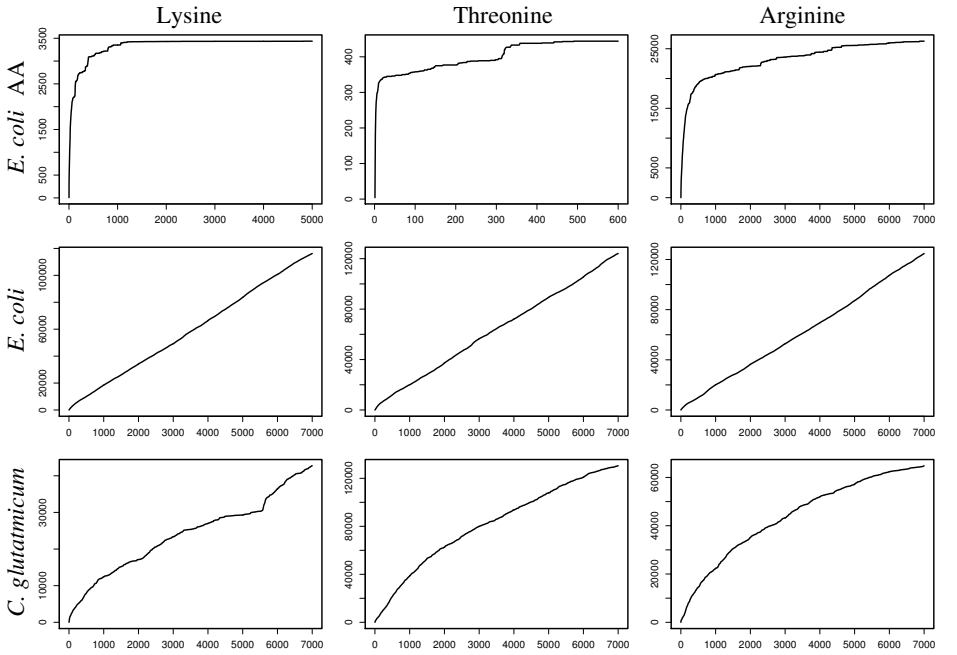


Figure 3: Time-course of the determination of EFMs for the three test-models: *E. coli* AA, [BWvK$^+$08]; *E. coli*, [FHR$^+$07]; *C. glutamicum*, [KN09]. The X-axis gives the running time in seconds and the Y-axis the number of EFMs found.

can be observed for the two other amino acids. In conjunction with the limited size of this model these results indicate that our method has already discovered a significant portion of all EFMs producing the three amino acids. In contrast, in the genome-scale system of *E. coli* we observe an almost linear increase in the number of EFMs without any saturation indicating that the number of EFMs existing in this model is much larger than the number already sampled.

Furthermore we tested the time required for the computation of 2000 EFMs in all models. We found the influence of network size on the running time much smaller than expected. Thus, it took on average 26.3 s to find 2000 EFMs in the model of *C. glutamicum* and 43 s in the genome-scale model of *E. coli* although both models differ more than five-fold in the number of reactions. This behaviour might be attributed to the simplex algorithm used to solve the linear programing problem described in Section 2.1. Since we are iteratively solving very similar problems and the simplex algorithm can start from a previous solution after changing some constraints, new solutions can be found very fast without need to consider the entire problem, but only a specific sub-part for which constraints were changed.

Another interesting aspect of the detected EFMs arises from the part of the network that can be used for the production of particular amino acids. For this analysis we combined

Model	# Rea.	AA	# EFMs	# Min.	CS	2000 EFMs
E. coli	220	Lysine	3436	16	94	95 s
AA metabolism		Threonine*	444	11	67	839 s
[BWvK+08]		Arginine	26276	18	95	8 s
E. coli	3558	Lysine	118598	29	1826	49 s
Genome-scale		Threonine	126491	26	2084	38 s
[FHR+07]		Arginine	127988	37	1895	42 s
C. glutamicum	641	Lysine	43115	23	240	28 s
Genome-scale		Threonine	131346	24	245	22 s
[KN09]		Arginine	65236	35	246	29 s

Table 1: Overview on computed EFMs. For each of the three test-models (number of reactions in the second column) the GA has been used to determine EFMs for the production of lysine, threonine and arginine (third column). The fourth column gives the number of EFMs detected after a running time of 7200 s. The fifth and sixth column indicate the minimal length of a detected EFM for the production of the given amino acid and the total number of different reactions used by all EFMs. The last column indicates the time required for the computation of 2000 EFMs averaged over 10 runs. In the case marked with *, the system only contained 444 EFMs.

all the computed EFMs for each test-case and determined the number of reactions used (Table 1). Furthermore, we determined the minimal number of reactions used by an EFM for the production of a given amino acid (Table 1). Combining all EFMs, the part of the metabolic network that can be used for the production of each amino acid varies in between 31% to 59% of the total network size. In consequence, there seems to be a great versatility in potential pathways. However, this versatility can be mostly attributed to the side-products of amino acid biosynthesis. For instance, in the production of lysine succinyl-CoA is converted to succinate. There are two ways of balancing succinyl-CoA and succinate. Either succinyl-CoA is additionally produced from the input medium and succinate is disposed through some other pathway, or succinate is reconverted into succinyl-CoA. Hence, we see a combinatorial explosion since the basic route producing lysine can be combined, on the one hand, with every pathway producing succinyl-CoA and consuming succinate. On the other hand this route can be combined with every possible pathway converting succinate into succinyl-CoA. This is also apparent from an analysis of the 64699 EFMs producing amino acids in the model of [BWvK+08]. Here we found that 35% of the EFMs do not only produce a single, but several amino acids. These additional amino acids can serve as sinks for side-metabolites.

4 Discussion

In this work we have outlined a new approach based on a genetic algorithm (GA) that allows to determine EFMs using a specific reaction in genome-scale metabolic networks. Previous methods that are based on searching paths in a graph representation of a metabolic network only guarantee to find connected routes while EFMs correspond to routes of actual metabolic conversions [dFSKF09]. Computing EFMs in a network in which they also

can be enumerated using deterministic algorithms we demonstrated that even large sets of EFMs can be recovered almost entirely. Comparing the time-course of the number of EFMs enumerated between a small and two large networks we concluded that we had already found a significant portion of all EFMs in a genome-scale model of *C. glutamicum* but only a small portion in a much larger model of *E. coli*. Analyzing the parts of the metabolic network which can be used by EFMs we found that they corresponded to 31% to 59% of the entire network even though individual pathways are usually much shorter. We attributed this result to the large variability of pathways that can be used to balance side-metabolites of amino acid biosynthetic pathways.

There exist several alternative approaches that allow a similar analysis of pathways in genome-scale networks. They either decompose a large network into smaller subnetworks or consider the entire network. The former approaches bear the problem that they only consider a small network on the local scale and thus they can contain artificial pathways that do not appear on the scale of the entire system [KdFS09]. Among the latter approaches especially constrained based methods are of importance. Methods from this field that allow to perform a similar analysis are flux balance analysis (FBA, [VP94]), flux variability analysis (FVA, [MS03]), and stochastic sampling of the solution space of eqs. 1 - 3 with additional upper bounds on reaction fluxes [WFGP04]. However, FBA only returns a specific pathway optimizing a certain objective function [VP94] and flux variability analysis only determines the set of reactions that can take part in alternative optimal pathways, without allowing to identify these pathways [MS03]. Stochastic sampling in contrast is very similar to our approach, but returns solutions that lie within the solution space of eqs. 1 - 3. Thus, rather than EFMs fluxes that correspond to combinations of EFMs are returned.

Our method represents an important step towards the analysis of EFMs, and thus of pathways, in genome-scale metabolic networks. While we used a fitness function that selects for diversity one can think of other functions that can be used. Thus, it is of interest to analyze suboptimal EFMs for the production of some metabolite which are in a specific range of yield per mole of an input metabolite or fulfill additional criteria like the production of a certain side-metabolite. Furthermore, since EFMs correspond to the concept of minimal transition invariants (MTIs) in petri-nets [SPM$^+$00, KH08], our approach can also be useful to find MTIs in large petri-nets.

5 Acknowledgments

Financial support from the German Ministry for Research and Education (BMBF) to C.K. and J.B. (grant FKZ 0315285E and FKZ 0313078E), and from the Fundação Calouste Gulbenkian, Fundação para a Ciência e a Tecnologia (FCT) and Siemens SA Portugal (PhD grant SFRH/BD/32961/2006) to L.F.F. is gratefully acknowledged.

References

[BK08] T. Blum and O. Kohlbacher. Using atom mapping rules for an improved detection of relevant routes in weighted metabolic networks. *J Comput Biol*, 15(6):565–576, 2008.

[BWvK+08] J. Behre, T. Wilhelm, A. von Kamp, E. Ruppin, and S. Schuster. Structural robustness of metabolic networks with respect to multiple knockouts. *J Theor Biol*, 252(3):433–441, Jun 2008.

[CCWvH06] D. Croes, F. Couche, S. J. Wodak, and J. van Helden. Inferring meaningful pathways in weighted metabolic networks. *J Mol Biol*, 356(1):222–236, Feb 2006.

[CS04] R. Carlson and F. Srienc. Fundamental *Escherichia coli* biochemical pathways for biomass and energy production: Identification of reactions. *Biotechnol Bioeng*, 85(1):1–19, Jan 2004.

[dFSKF09] L. F. de Figueiredo, S. Schuster, C. Kaleta, and D. A. Fell. Can sugars be produced from fatty acids? A test case for pathway analysis tools. *Bioinformatics*, 25(1):152–158, Jan 2009.

[FHR+07] A. M. Feist, C. S. Henry, J. L. Reed, M. Krummenacker, A. R. Joyce, P. D. Karp, L. J. Broadbelt, V. Hatzimanikatis, and B. Ø Palsson. A genome-scale metabolic reconstruction for *Escherichia coli* K-12 MG1655 that accounts for 1260 ORFs and thermodynamic information. *Mol Syst Biol*, 3:121, 2007.

[FP08] A. M. Feist and B. Ø. Palsson. The growing scope of applications of genome-scale metabolic reconstructions using *Escherichia coli*. *Nat Biotechnol*, 26(6):659–667, Jun 2008.

[GK04] J. Gagneur and S. Klamt. Computation of elementary modes: A unifying framework and the new binary approach. *BMC Bioinformatics*, 5:175, 2004.

[KdFS09] C. Kaleta, L. F. de Figueiredo, and S. Schuster. Can the whole be less than the sum of its parts? Pathway analysis in genome-scale metabolic networks using elementary flux patterns. *Genome Res Res*, 2009. Accepted.

[KH08] I. Koch and M. Heiner. *Biological Network Analysis*, chapter Petri Nets, pages 139 – 180. Wiley Book Series in Bioinformatics. Wiley & Sons, 2008.

[KN09] K. R. Kjeldsen and J. Nielsen. In silico genome-scale reconstruction and validation of the *Corynebacterium glutamicum* metabolic network. *Biotechnol Bioeng*, 102(2):583–597, Feb 2009.

[KS02] S. Klamt and J. Stelling. Combinatorial complexity of pathway analysis in metabolic networks. *Mol Biol Rep*, 29(1-2):233–236, 2002.

[LH03] R. Lougee-Heimer. The Common Optimization INterface for Operations Research: Promoting open-source software in the operations research community. *IBM J Res Dev*, 47(1):57–66, 2003.

[MS03] R. Mahadevan and C. H. Schilling. The effects of alternate optimal solutions in constraint-based genome-scale metabolic models. *Metab Eng*, 5(4):264–276, Oct 2003.

[PB08] F. J. Planes and J. E. Beasley. A critical examination of stoichiometric and path-finding approaches to metabolic pathways. *Brief Bioinform*, 9(5):422–436, Sep 2008.

[PRFN05] K. R. Patil, I. Rocha, J. Förster, and J. Nielsen. Evolutionary programming as a platform for in silico metabolic engineering. *BMC Bioinformatics*, 6:308, 2005.

[RAS+05] S. A. Rahman, P. Advani, R. Schunk, R. Schrader, and D. Schomburg. Metabolic pathway analysis web service (Pathway Hunter Tool at CUBIC). *Bioinformatics*, 21(7):1189–1193, Apr 2005.

[Sch98] A. Schrijver. *Theory of Linear and Integer Programming*. John Wiley & Sons, June 1998.

[SDF99] S. Schuster, T. Dandekar, and D. A. Fell. Detection of elementary flux modes in biochemical networks: A promising tool for pathway analysis and metabolic engineering. *Trends Biotechnol*, 17(2):53–60, Feb 1999.

[SKB+02] J. Stelling, S. Klamt, K. Bettenbrock, S. Schuster, and E. D. Gilles. Metabolic network structure determines key aspects of functionality and regulation. *Nature*, 420(6912):190–193, Nov 2002.

[SLP00] C. H. Schilling, D. Letscher, and B. Ø. Palsson. Theory for the systemic definition of metabolic pathways and their use in interpreting metabolic function from a pathway-oriented perspective. *J Theor Biol*, 203(3):229–248, Apr 2000.

[SPM+00] S. Schuster, T. Pfeiffer, F. Moldenhauer, I. Koch, and T. Dandekar. Structural analysis of metabolic networks: Elementary flux modes, analogy to Petri nets, and application to *Mycoplasma Pneumoniae*. In *German Conference on Bioinformatics*, pages 115–120, 2000.

[TS08] Marco Terzer and Jörg Stelling. Large-scale computation of elementary flux modes with bit pattern trees. *Bioinformatics*, 24(19):2229–2235, Oct 2008.

[TUS08] C. T. Trinh, P. Unrean, and F. Srienc. Minimal *Escherichia coli* cell for the most efficient production of ethanol from hexoses and pentoses. *Appl Environ Microbiol*, 74(12):3634–3643, Jun 2008.

[vKS06] A. von Kamp and S. Schuster. Metatool 5.0: Fast and flexible elementary modes analysis. *Bioinformatics*, 22(15):1930–1931, Aug 2006.

[VP94] A. Varma and B. Ø. Palsson. Stoichiometric flux balance models quantitatively predict growth and metabolic by-product secretion in wild-type *Escherichia coli* W3110. *Appl Environ Microbiol*, 60(10):3724–3731, Oct 1994.

[WBE06] V. F. Wendisch, M. Bott, and B. J. Eikmanns. Metabolic engineering of *Escherichia coli* and *Corynebacterium glutamicum* for biotechnological production of organic acids and amino acids. *Curr Opin Microbiol*, 9(3):268–274, Jun 2006.

[WFGP04] S. J. Wiback, I. Famili, H. J. Greenberg, and B. . Palsson. Monte Carlo sampling can be used to determine the size and shape of the steady-state flux space. *J Theor Biol*, 228(4):437–447, Jun 2004.

[YTP07] M. Yeung, I. Thiele, and B. Ø. Palsson. Estimation of the number of extreme pathways for metabolic networks. *BMC Bioinformatics*, 8(1):363, 2007.

On the Benefits of Multimodal Optimization for Metabolic Network Modeling

Marcel Kronfeld, Andreas Dräger, Moritz Aschoff, Andreas Zell

Center for Bioinformatics Tübingen
Eberhard Karls University Tübingen
Sand 14, D-72076 Tübingen, Germany
{marcel.kronfeld, andreas.draeger, moritz.aschoff, andreas.zell}@uni-tuebingen.de

Abstract: The calibration of complex models of biological systems requires numerical simulation and optimization procedures to infer undetermined parameters and fit measured data. The optimization step typically employs heuristic global optimization algorithms, but due to measurement noise and the many degrees of freedom, it is not guaranteed that the identified single optimum is also the most meaningful parameter set. Multimodal optimization allows for identifying multiple optima in parallel. We consider high-dimensional benchmark functions and a realistic metabolic network model from systems biology to compare evolutionary and swarm-based multimodal methods. We show that an extended swarm based niching algorithm is able to find a considerable set of solutions in parallel, which have significantly more explanatory power. As an outline of the information gain, the variations in the set of high-quality solutions are contrasted to a state-of-the-art global sensitivity analysis.

1 Introduction

The parameter estimation for mathematical models of biological systems is a demanding task. For complex systems of differential equations, for example, usually there is hardly any previous knowledge on the required model type and its parameterization. Often, numerical simulation and heuristic optimization of the measurement fit is the only way of infering a parameter set that reproduces the measured data and thus the only way of judging the model's ability to represent the measurements. [Ban08]

This approach brings with it certain ambiguities due to measurement noise and system complexity, which not only means that the target function is non-convex (multimodal) but also entails the existence of distinct parameter sets fitting the data with a very similar quality. This renders the assumption that global optimization methods find the vicinity of the global optimum very quickly [BCPB+08, RFEB06] rather challengeable. Moreover, the local optima may be so similar that they can hardly be discriminated with respect to biological significance—a fact usually ignored in parameter estimation, where mostly artificial data and low-scale noise are used. One way of delivering more evidence on model properties and biological importance lies in model sensitivity [STCR04]. On the one hand, it is usually observed that biological systems are relatively robust towards small changes,

e.g., in concentration of substances involved in a biochemical reaction system. On the other hand, there may be single parameter changes that disturb the system significantly more than others, and if a mathematical model was able to predict these sensitivities, it would increase its biological relevance. This motivates the idea to not only search for one optimum but for a set of different high-quality solutions, to compare them and test for common dependencies and sensitivities. This can be achieved by multimodal optimization (MMO) methods [SD06], which, however, are often developed on simple benchmarks. We thus perform preliminary tests on difficult benchmarks before tackling the computationally much more expensive application. The target system, a metabolic network of the industrially important *Corynebacterium glutamicum*, has been modeled using Generalized Mass Action Kinetics (GMAK) and examined by unimodal optimization [DKZ+09]. In the work at hand, MMO is applied to find sets of high-quality local optima of the biochemical model. We contrast the parameter distribution of identified optima with a global sensitivity analysis to show how, thereby, possibly new biological implications can be drawn.

2 Heuristic Multimodal Optimization

Researchers often face nonlinear, non-convex problems the derivative of which is infeasible to compute. In these cases, modern stochastic metaheuristic optimization methods are an apt choice, because they have a higher chance to locate the global optimum compared to classical local search methods [Ban08]. This is mostly because, instead of only looking at a single possible solution at a time, a whole set ("population") is processed, which converges on the global optimum with higher probability. Two particularly successful optimization techniques are biologically inspired. In Evolutionary Algorithms (EAs), candidate solutions in $\mathbb{R}^n$ are assigned a quality measure, and better ones are selected, recombined and mutated hoping to produce better individuals from good ones. In Particle Swarm Optimization (PSO), a candidate solution $x \in \mathbb{R}^n$ ("particle") is assigned a "velocity" vector. x is accelerated towards (i) the best position the particle itself has come across so far (p^h) and (ii) the best position in a particle neighborhood (p^n). Formally: $v_i(t+1) = \omega v_i(t) + \phi_1 r_1 (p_i^n - x_i) + \phi_2 r_2 (p_i^h - x_i)$ for all vector components i, where ω and $\phi_{1/2}$ are control parameters, while $r_{1/2} \sim U(0,1)$ provide for randomization. A comprehensive introduction to EAs and PSO is given in [Eng02].

For multimodal optimization specifically, the population diversity is boosted to allow multiple optima to be occupied in parallel. Early methods such as *sharing*, *crowding* or *clearing* [SD06, Mah95] accomplish this by punishing similarity within the EA population. Recent approaches emphasize *niching* by explicitely forming sub-populations, e.g., using clustering in EA [SSUZ03] or sub-swarm-formation in PSO [BEvdB03]. The sub-populations are to cumulate around local optima in a self-organizing way, while a diverse main-population may keep exploring the search space. Current works often report swarm methods to be superior to other methods [BEvdB03, ÖY07], which fail especially in higher dimensions [SD06] and in lower dimensions may be outperformed by simple multi-start Hill-Climbing (HC) [SSUZ03]. As swarm-based methods showed to be more promising on the target model than traditional methods [DKZ+09], such as HC or Genetic Algo-

Name	Function	Domain	Parameters
$f_{M6}(\vec{x})$	$= 1 - \sin(30x_1^3)\sin(25\,x_2^2 x_1)$	$[0,1]^2$	
$f_{M10}(\vec{x})$	$= 1 - \frac{1}{n}\sum_{i=1}^{n}[1 - \sin^6(5\pi x_i)]$	$[0,1]^n$	
$f_{SR}(\vec{x})$	$= \sum_{i=1}^{n}(z_i^2 - 10\cos(2\pi z_i) + 10)$	$[-5,5]^n$	$\vec{z} = \vec{x} - \vec{o}$ for shifted optimum $\vec{o}$
$f_{M13}(\vec{x})$	$= cn - \sum_{i=1}^{n} x_i \sin\sqrt{\lvert x_i \rvert}$	$[-512.03, 511.97]^n$	$c = 418.9829$

Table 1: Preliminary benchmark functions.

rithms (GA), we concentrate on swarm-based niching in comparison to clustering EAs.

Typically, an MM approach introduces new parameters to the optimization procedure. The Clustering-Based Niching EA (CBNEA, [SSUZ03]) performs density-based clustering with a strategy parameter σ on the population. Since the selection drives the population towards areas of better fitness, it is expected that clusters form around local optima. Each cluster is decoupled from the main population and evolved with an Evolution Strategy (ES) or a GA to identify a local optimum. We employ CBNES with $\sigma = 0.1$ and a (μ, λ)-ES with $\frac{\mu}{\lambda} = \frac{3}{10}$, simple uniform step-size mutation ($p_m = 1$) and one-point-crossover ($p_c = 0.5$). The real-valued CBNGA only differs in the selection method, which is tournament selection on groups of four instead of elitistic ES-selection, so its selective pressure is lower.

The NichePSO algorithm [EvL07] forms niches by looking for particles having a fitness standard deviation σ below a threshold δ for k iterations. Any such particle forms a sub-swarm with its closest neighbor, and they are again decoupled from the main swarm. To allow for distributed sub-swarm formation, the neighborhood attraction is deactivated in the main swarm ($\phi_1 = 1.2$, $\phi_2 = 0$), whereas sub-swarm particles are fully connected. The inertness factor ω is linearly decreased, and sub-swarms are merged if they overlap. In [ÖY07], a maximum merge distance is introduced to avoid too large sub-swarms, a problem of the original NichePSO. ANPSO is a further extension which adaptively sets the allowed sub-swarm radius to the average of each particle's distance to its closest neighbor [BL06]. ANPSO also reintroduces neighborhood attraction for the main swarm ($\phi_2 = 0.6$) to enforce global search.

We extend the niching PSO variants by a deactivation strategy [SSUZ03]: when a sub-swarm converges, its best position is stored and the particles are reinitialized. Similar to sub-swarm creation, we deactivate a sub-swarm if all its particles have a fitness standard deviation below a threshold ε_{deact} for k iterations. We set $\varepsilon_{deact} = \delta$ ($\delta = 10^{-4}$ in NichePSO [EvL07]). Deactivation enhances exploration and allows the algorithm to identify more optima than the initial swarm size.

Conclusively, we test the following algorithms with a population size of 200: NichePSO with standard parameters, enhanced NichePSO (*NPSO**, [EvL07]), ANPSO with standard parameters [BL06], and an ANPSO variant which employs the SPSO-strategy for the main-swarm using the adaptive swarm-size parameter defined by ANPSO (*ANPSO**). The *ANPSO** strategy parameters are set to $\phi_1 = 1.2$, $\phi_2 = 1.2$, and $\omega = 0.73$.

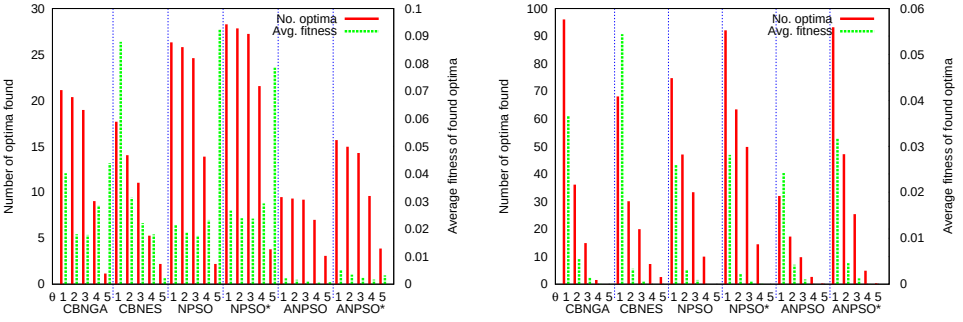


Figure 1: Number of optima identified and their average fitness for f_{M6} (left) and f_{M10} (right).

3 Preliminary Evaluation

We selected 5 diverse, multimodal functions to delineate some characteristics of the MM methods, listed in Tab. 1. f_{M6} is 2-dimensional and has 33 local optima which are unevenly spaced, whereas f_{M10} has 5^n optima of equal fitness distributed evenly;we set $n = 5$. f_{SR} (shifted Rastrigin's) has numerous local optima in a global basin of attraction. f_{M13} (Schwefel's sine root) has numerous local optima and no global basin of attraction. For the latter two we test $n \in \{10, 30\}$. All functions are treated as minimization problems with the solution at $f(\vec{x}^*) = 0$. To measure optimization performance, we look at the number of known optima found with several accuracy thresholds θ_i and the average fitness of the optima. Specifically, we compare $\theta \in \{0.05, 0.01, 0.005, 0.001, 0.0001\}$ and expect that fewer optima are found with decreasing θ corresponding to increasing accuracy.

For f_{SR} and f_{M13} we did not presume knowledge of local optima. The performance criteria for f_{SR}/f_{M13} are based on post processing: the suggested solutions are clustered and the best representative of each cluster C_i is interpreted as candidate solution c_i. Each c_i is refined using a Nelder-Mead-Simplex (NMS) local search started in the close neighborhood of c_i. In case the NMS converges without moving away by more than θ from c_i, the candidate is classified as being locally optimal. The number of solutions found in this way gives a relative measure on how well algorithms converge on a specific benchmark. Additionally, we look at the average fitness of the suggested solutions without regarding convergence state, because for difficult functions, often no close convergence is reached.

Benchmark Results: For each MMO method under consideration, we averaged the results of 25 runs à $5{,}000 \cdot n$ evaluations. Figures 1-2 show the number of optima found (left axis, more is better) and their average fitness (right axis, less is better), for each method and the five different thresholds. As can be seen in Fig. 1, NichePSO tends to find more optima than the CBN and ANPSO methods, but with worse fitness values. For f_{SR} the quality delivered by NichePSO is hardly acceptable.

CBN and ANPSO usually reach better fitness and higher accuracy than NichePSO, which invests equally in both good and bad optima. This, too, can be attributed to the absence of

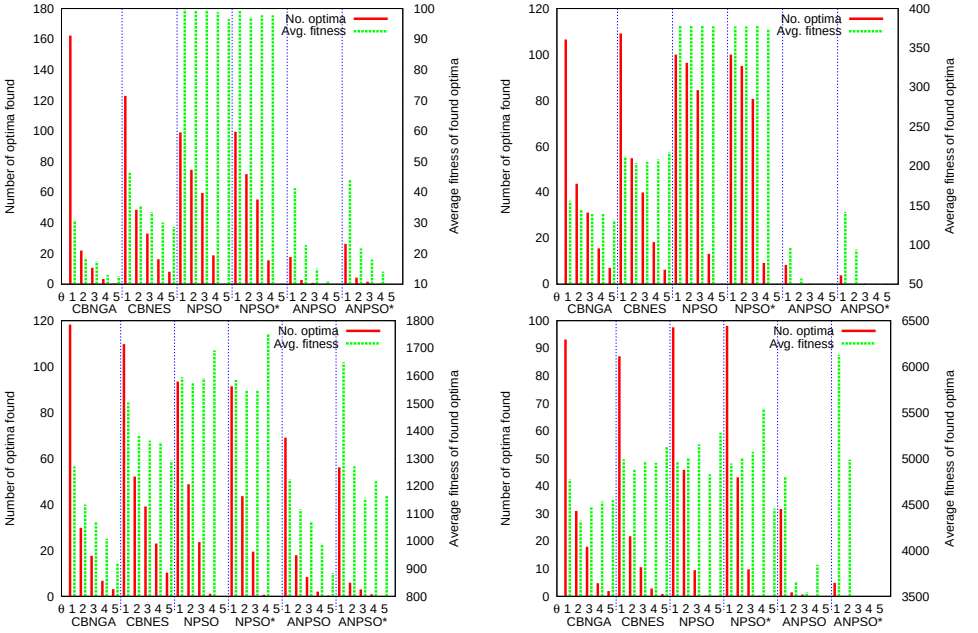


Figure 2: f_{SR} (top) and f_{M13} (bottom) in 10-D (left) and 30-D (right).

a main swarm for global search in NichePSO. To rate the statistical significance, we performed Student's t-tests on all pairs of algorithms testing the null hypotheses that, for each benchmark and threshold, they (i) find the same number of optima and (ii) reach the same fitness quality. For a condensed comparison, we scored $+1(-1)$ for the superior (inferior) algorithm whenever that hypothesis could be significantly rejected at a 5% level. Table 2 shows the summed-up scores for the number of optima (left) and the fitness quality (right). A positive number k in line A_i and column A_j means that algorithm A_i was significantly better than algorithm A_j in k more cases than the other way around. The tests support the conclusions that the CBN variants find slightly fewer optima than NPSO with better fitness values. Also, ANPSO* finds more optima than ANPSO, whereas both find significantly fewer optima than the other algorithms with better fitness values.

For the more difficult benchmarks f_{SR} and f_{M13}, some algorithms do not find any optimum with certain accuracies θ_i, which can be seen from missing fitness bars, e.g., for ANPSO on f_{SR}-30-D and $\theta \in \{0.005, 0.001, 0.0001\}$. Yet this is not equal to bad performance when looking beyond the convergence state. For Tab. 3, the resulting populations were clustered, a cluster's best particle interpreted as local solution, and their average number, mean and minimum fitness values are displayed. Since on $f_{SR/M13}$, CBNES was outperformed by CBNGA, and NPSO* performed very similar to NPSO, those are omitted.

The comparison indicates that, although ANPSO does not converge closely on the local optima resulting in fewer identified optima in Fig. 2, it produces good fitness values across the sub-swarms. As can be expected, NichePSO is competitive on f_{M13}, but not on f_{SR},

	CG	CE	NP	NP*	AP	AP*	CG	CE	NP	NP*	AP	AP*
CBNGA	0	-7	-7	-4	21	15	0	15	12	14	-11	-4
CBNES	7	0	-1	-1	26	17	-15	0	13	11	-14	-7
NPSO	7	1	0	0	18	16	-12	-13	0	-1	-17	-9
NPSO*	4	1	0	0	19	20	-14	-11	1	0	-17	-10
ANPSO	-21	-26	-18	-19	0	-2	11	14	17	17	0	10
ANPSO*	-15	-17	-16	-20	2	0	4	7	9	10	-10	0

Table 2: Significance scores regarding No. optima found (left) and fitness (right).

		Avg. #Opt		Avg. Mean Fit.		Avg. Min. Fit.	
	Algorithm	10D	30D	10D	30D	10D	30D
f_{SR}	HC-100	100.00	100.0	80.25	351.4	42.52	207.8
	CBNGA	185.60	255.7	39.60	219.2	6.12	50.2
	NPSO	100.60	101.4	100.70	380.7	23.90	187.2
	ANPSO	17.76	123.1	92.17	221.6	12.33	57.8
	ANPSO*	15.72	17.0	48.38	269.1	11.69	100.2
f_{M13}	HC-100	100.00	100.0	2082.9	7838.0	1175.7	4110.6
	CBNGA	102.36	120.4	1407.2	5926.4	569.2	3785.1
	NPSO	100.00	100.2	1614.1	4966.6	660.0	3231.8
	ANPSO	76.08	170.3	1265.4	5935.7	612.7	3081.3
	ANPSO*	40.80	51.1	1067.1	5285.4	417.0	2693.4

Table 3: Clustered results on f_{SR} and f_{M13} in 10 / 30 dimensions.

whose global basin of attraction suits the global search components of CBN and ANPSO.

4 The Metabolic Network

Figure 3 shows the reaction pathway of the valine (Val) and leucine (Leu) biosynthesis in *C. glutamicum* according to [DKZ+09]. The metabolic pathway starts with the formation of 2-ketoisovalerate (KIV) from pyruvate (Pyr) in three reaction steps [CFF+08]. At the KIV node the pathway branches: Two parallel reactions produce Val and one forms 2-isopropylmalate (2-IPM), the starting substance for the Leu production. Both Val and Leu can be used for biomass production or secreted into the culture medium—the industrially interesting outcome. Val and Leu inhibit their production rates in four feedback loops. The competition of both products for the secretory protein is modeled by inhibition: Val inhibits the secretion of Leu and vice versa. Additionally, Val inhibits reactions R_{1-3} while Leu inhibits R_7 (Tab. 4). The fast reaction 2-IPM $\rightleftharpoons$ 3-IPM is assumed to process in equilibrium and combined with $\text{3-IPM} + \text{NAD}^+ \longrightarrow \text{2-I}_3\text{OS} + \text{NADH}_2$ and (2S)-2-isopropyl-3-oxosuccinate ($\text{2-I}_3\text{OS}$) $\longrightarrow$ 2-ketoisocaproate (KIC) $+\text{CO}_2$, which only depend on the concentration of 2-IPM, introducing the symbol IPM for both derivates.

Reaction	Parameters (fw/bw/ihb)	Reaction	Parameters (fw/bw/ihb)
R_1 $2\,\mathrm{Pyr} \rightleftharpoons \mathrm{AcLac} + \mathrm{CO_2}$	p_0, p_{10}, p_{18}	R_2 $\mathrm{AcLac} + \mathrm{NADPH_2} \rightleftharpoons \mathrm{DHIV} + \mathrm{NADP^+}$	p_1, p_{11}, p_{19}
R_3 $\mathrm{DHIV} \rightleftharpoons \mathrm{KIV} + \mathrm{H_2O}$	p_2, p_{12}, p_{20}	R_4 $\mathrm{KIV} + \mathrm{Gln} \rightleftharpoons \mathrm{Val} + \alpha\mathrm{KG}$	$p_3, p_{13}, -$
R_5 $\mathrm{KIV} + \mathrm{Ala} \rightleftharpoons \mathrm{Val} + \mathrm{Pyr}$	$p_4, p_{14}, -$	R_6 $\mathrm{Val} \longrightarrow \mathrm{Val_{ext}}$	$p_5, -, p_{21}$
R_7 $\mathrm{KIV} + \mathrm{AcCoA} \rightleftharpoons \mathrm{IPM} + \mathrm{CoA}$	p_6, p_{15}, p_{22}	R_8 $\mathrm{IPM} + \mathrm{NAD^+} \rightleftharpoons \mathrm{KIC} + \mathrm{NADH_2} + \mathrm{CO_2}$	$p_7, p_{16}, -$
R_9 $\mathrm{KIC} + \mathrm{Gln} \rightleftharpoons \mathrm{Leu} + \alpha\mathrm{KG}$	$p_8, p_{17}, -$	R_{10} $\mathrm{Leu} \longrightarrow \mathrm{Leu_{ext}}$	$p_9, -, p_{23}$

Table 4: The reaction system. All except R_6 and R_{10} are modeled reversibly [DKZ+09]. We refer to Dihydroxy-isovalerate as DHIV, Acetyl-CoA as AcCoa, Acetolactate as AcLac, α-Ketoglutaric Acid as αKG; cf. Sec. 4.

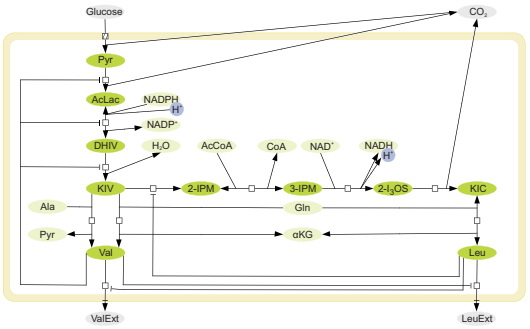


Figure 3: Val/Leu synthesis model [DKZ+09].

Algorithm	#Opt	Avg. #Opt	Best fit.
MSHC	0	0.0	36.13
CBNGA	2	0.4	22.88
NPSO*	3	0.6	22.68
ANPSO*	110	22.0	21.05

Figure 4: No. of interesting optima found (GMAKr model).

In an experiment by Magnus *et al.*, a glucose shock was caused after a starvation period to a *C. glutamicum* culture [MHOT06]. Over a time span of 25 s, 47 samples were taken for 13 metabolites on the pathway, which serve as target system output in the optimization. While Magnus *et al.* used LinLog kinetics, we model the system based on a reversible Generalized Mass Action Kinetics formulation (GMAKr) [DKZ+09]. Table 4 outlines the component reactions, of which all but R_6 and R_{10}—the secretion out of the cell—are considered to be reversible. Conclusively, there are 24 velocity (forward/backward) and inhibition factors to be optimized with respect to how well the measured data can be reproduced by the GMAK model. Due to the strong backwards coupling in the network and necessary numerical integration, the model is computationally expensive and highly nonlinear. Moreover, a noticeable ratio of possible parameters are unstable, which is why they are initialized around velocity values typically observed.

Results on the Metabolic Network: The benchmark evaluation in Sec. 3 shows that ANPSO tends to find fewer optima of higher quality compared to NPSO and CBN methods, especially for the complex 30-D f_{M13} function (Tab. 3). We therefore assume ANPSO to locate multiple high-quality solutions for *C. glutamicum's* Val/Leu synthesis network.

We allow a number of 500 individuals for 500,000 evaluations per run with 5 runs per

algorithm. Global optimization reaches fitness values near $20-23$, so we define a fitness threshold of $\Theta = 25$ below which solutions are said to be interesting. Tab. 4 lists the number of such solutions found in 5 runs for the multi-start hill-climber (MSHC), CBNGA and the swarm-based variants. All population-based approaches clearly outperform the hill-climber, yet only ANPSO* identifies a noticeable number of distinct optima per run. The relatively bad performance of CBNGA compared to the benchmarks is consistent with earlier results on the considered system [DKZ+09]. The performance difference between NichePSO and ANPSO suggests that the local solutions of the target function lie within larger areas of relatively good fitness values which can be exploited by ANPSO.

It should be noted that earlier studies reached better single fitness values using global optimization, e.g., RSE 20.334 [DKZ+09]. However, due to the measurement noise, the single optimal parameter set for a deterministic model will hardly be the most biologically plausible one—it might even be a "phantom optimum" resulting from numerical inaccuracies. A large set of high-quality solutions contains more information and is a basis for analyses of properties hard to handle during optimization. For example, biological systems are known to be stable: they operate within steady-states to which they return after small perturbations [HS96, pp. 40–52]. Thermodynamic validity as well as global or local sensitivity indices can also be regarded for the fitted parameter sets. An exemplary analysis follows in the next section. Compared to [DKZ+09], we demonstrate that a multimodal optimization approach delivers a set of high-quality solutions at a remarkably lower computational cost, since multiple high-quality solutions can be identified within single runs, while global optimizers are designed to converge on a single solution.

Parameter Distribution: Fig. 5 (a) shows the variations within a set of 21 interesting solutions found in one ANPSO* run. They are contrasted with an Extended Fourier Amplitude Sensitivity Test (EFAST) on the target function (Fig. 5 (b)). Several correlations are obvious: some parameters of low sensitivity, such as p_5 and p_{11}, vary over several orders of magnitude in the set of optimized solutions, while others with a very high total effect such as p_3, p_8, and p_{13} receive very similar values. The sensitivity analysis implies that of parameters p_3 and p_4, which correspond to the parallel reactions R_4 and R_5, p_3 shows much higher sensitivity. The variations in the high-quality solutions are very small for p_3 and larger for p_4, leading to the conclusion that R_4 is dominant among the two.

More interesting observations come up from variations in the optimized set that seem unexpected from the global sensitivities: While p_1 and p_{15} have a very similar total effect, p_1 has a considerably larger variance in the optimized set. This indicates that reaction R_7, of which p_{15} is the backwards velocity parameter, is as important as expected from the global sensitivity analysis, while R_2 at the entrance of the production cycle is less sensitive for biologically relevant parameters. Yet comparing p_4 and p_5, both of which exhibit very low global sensitivity, depicts that p_5 varies over a much larger scale than p_4 in the optimized set. This indicates that R_5, of which p_4 is the forward velocity, is of more relative importance than R_6. Looking back at Tab. 4 and Fig. 3 this turns out to be plausible, as R_5 consumes the central KIV which is a key substance at the crossing of the network, while R_6—and with it p_5—"only" affects the transport of Val out of the cell, having no recurrent effects on the system.

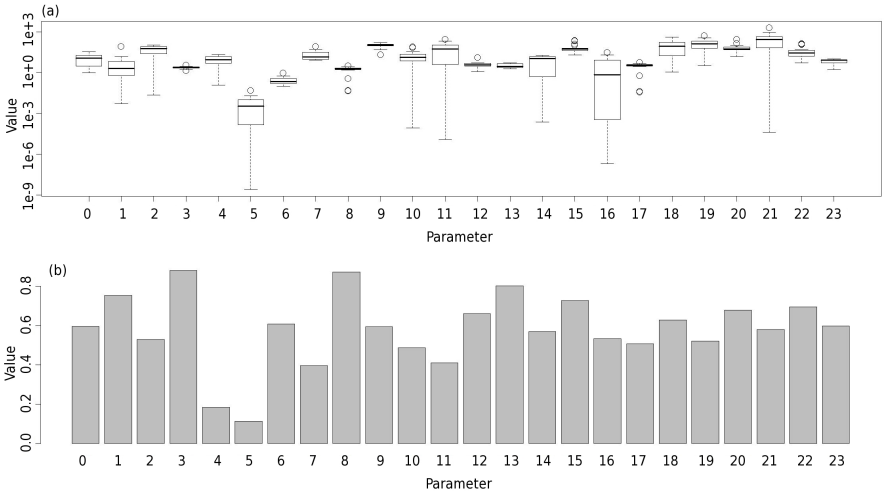


Figure 5: Parameter value distribution (a) and sensitivity total effect per parameter (b).

5 Conclusion

Multimodal optimization techniques aim at finding several local optima of an unknown target function in parallel. As they are usually developed on low-dimensional benchmarks, we looked at a set of current methods and benchmarked them on high-dimensional functions, finding that clustering EA approaches as well as the adaptive swarm-based approach ANPSO are able to find multiple solutions with sensible fitness values, where an adapted version, ANPSO*, was especially successful on the most complex benchmark function. The standard NichePSO approach lacks a globally searching main swarm and is mostly unable to compete on functions with many local optima on large-scale basins of attraction. The subsequent application on a GMAK model of a metabolic network representing the Leu/Val-synthesis of *C. glutamicum* showed that ANPSO* finds a considerable number of distinct high-quality solutions in parallel, while CBN and NichePSO widely fail. This can be attributed to the rather exploitative nature of ANPSO. Since default GA and ES strategies showed to be inferior to swarm-methods on the GMAK model earlier [DKZ+09], the success of ANPSO over the CBNEA variants is consistent. The analysis of parameter variations of a set of local optima compared to a global sensitivity analysis allowed several interesting interpretations which indicate that multimodal optimization can be a useful tool for assessing the results of heuristic parameter estimation in systems biology.

References

[Ban08] J. R. Banga. Optimization in computational systems biology. *BMC Systems Biology*, 2:47, May 2008.

[BCPB+08] E. Balsa-Canto, M. Peifer, J. R. Banga, J. Timmer, and C. Fleck. Hybrid optimization method with general switching strategy for parameter estimation. *BMC Systems Biology*, 2:26, Mar. 2008.

[BEvdB03] R. Brits, A.P. Engelbrecht, and F. van den Bergh. Scalability of Niche PSO. In *The IEEE Swarm Intelligence Symp. SIS '03. Proc. of*, pp. 228 – 234, 2003.

[BL06] S. Bird and X. Li. Adaptively choosing niching parameters in a PSO. In *GECCO '06: Proc. of the 8th annual conf. on Genetic and evolutionary computation*, pp. 3–10, New York, NY, USA, 2006. ACM.

[CFF+08] R. Caspi, H. Foerster, C. A. Fulcher, P. Kaipa, M. Krummenacker, M. Latendresse, S. Paley, S. Y. Rhee, A. G. Shearer, C. Tissier, T. C. Walk, P. Zhang, and P. D. Karp. The MetaCyc Database of metabolic pathways and enzymes and the BioCyc collection of Pathway/Genome Databases. *Nucleic Acids Research*, 36(Database issue):D623–D631, Jan. 2008.

[DKZ+09] A. Dräger, M. Kronfeld, M. J. Ziller, J. Supper, H. Planatscher, J. B. Magnus, M. Oldiges, O. Kohlbacher, and A. Zell. Modeling metabolic networks in *C. glutamicum*: a comparison of rate laws in combination with various parameter optimization strategies. *BMC Systems Biology*, 3:5, Jan. 2009.

[Eng02] A. Engelbrecht. *Computational Intelligence: An Introduction*. Halsted, New York, NY, USA, 2002.

[EvL07] A. Engelbrecht and N. H. van Loggerenberg. Enhancing the NichePSO. In D. Srinivasan and L. Wang, eds., *2007 IEEE Cong. on Evolutionary Comp.*, pp. 2297–2302, Singapore, 2007. IEEE Computational Intelligence Society, IEEE Press.

[HS96] R. Heinrich and S. Schuster. *The Regulation of Cellular Systems*. Chapman and Hall, New York, NY, 1996.

[Mah95] S. W. Mahfoud. *Niching Methods for Genetic Algorithms*. PhD thesis, University of Illinois at Urbana-Champaign, Champaign, IL, USA, 1995.

[MHOT06] J. B. Magnus, D. Hollwedel, M. Oldiges, and R. Takors. Monitoring and Modeling of the Reaction Dynamics in the Valine/Leucine Synthesis Pathway in Corynebacterium glutamicum. *Biotech. Progress*, 22:1071–1083, 2006.

[ÖY07] E. Özcan and M. Yilmaz. Particle Swarms for Multimodal Optimization. In *ICANNGA: Int. Conf. on Adaptive and Natural Computing Algorithms*, pp. 366–375, 2007.

[RFEB06] M. Rodriguez-Fernandez, J. A. Egea, J. R. Banga. Novel metaheuristic for parameter estimation in nonlinear dynamic biological systems. *BMC Bioinf.*, 7:483, Nov. 2006.

[SD06] G. Singh and K. Deb. Comparison of multi-modal optimization algorithms based on evolutionary algorithms. In *GECCO '06: Proc. of the 8th annual conf. on Genetic and evolutionary computation*, pp. 1305–1312, New York, NY, USA, 2006. ACM.

[SSUZ03] F. Streichert, G. Stein, H. Ulmer, and A. Zell. A Clustering Based Niching EA for Multimodal Search Spaces. In *Proc. of Evolution Artificielle (LNCS 2935*, pp. 293–304. Springer, 2003.

[STCR04] A. Saltelli, S. Tarantola, F. Campolongo, and M. Ratto. *Sensitivity Analysis in Practice: A Guide to Assessing Scientific Models*. Halsted, New York, NY, USA, 2004.

Automated Bond Order Assignment as an Optimization Problem

Anna Katharina Dehof, Alexander Rurainski, Hans-Peter Lenhof, Andreas Hildebrandt

Center for Bioinformatics, Saarland University, 66041 Saarbrücken, Germany

anne@bioinf.uni-sb.de

Abstract: Numerous applications in Computational Biology process molecular structures and hence require not only reliable atomic cordinates, but also correct bond order information. Regrettably, this information is not always provided in molecular databases like the Cambridge Structural Database or the Protein Data Bank. Very different strategies have been applied to derive bond order information, most of them relying on the correctness of the atom coordinates. We extended a different ansatz proposed by Wang et al. that assigns heuristic molecular penalty scores solely based on connectivity information and tries to heuristically approximate its optimum. In this work, we present two efficient and exact solvers for the problem replacing the heuristic approximation scheme of the original approach: an ILP formulation and an A* approach. Both are integrated into the upcoming version of the Biochemical Algorithms Library BALL and have been successfully validated on the MMFF94 validation suite.

1 Introduction

Correct bond order information is essential for many algorithms in Computational Structural Biology and Chemistry, since bonds do not only define the connectivity of atoms in a molecule but also define structural aspects like rotatability of individual groups. However, bond order information can often not be directly infered from the available experimental data. Even important molecular databases, like the Protein Data Bank (PDB) [BHN03] and the Cambridge Structural Database [All02], are known to contain erroneous data for connectivity and bond order information [Lab05] or to even omit them entirely. For proteins and nucleic acids, bond orders can be easily deduced due to their building block nature, but this does not hold for other kinds of molecules like ligands. The problem is made much worse by the fact that quite often, the bond order assignment for a given molecule is not unique, even when neglecting symmetries in the molecule. The chemical reasons for this effect are complex and out of scope of this work; here we just want to state that the concept of integer bond orders is only an approximation to a full quantum chemical treatment, and cannot explain all effects occurring in molecules. Important examples are aromatic or delocalized bonds, leading to different resonance structures (c.f. Fig. 1). In addition, formal charges are often not contained in the input files, but atoms carrying a formal charge will obviously show a different bonding pattern. One body of opinion tries to overcome these

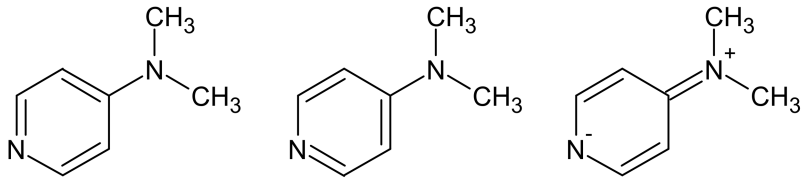


Figure 1: Different resonance structures of 4-(N,N-dimethylamino)pyridine. A bond order assignment program should optimally be able to compute all of these configurations.

obstacles by hand curation which clearly provides the highest reliability. On the other hand, manual data curation does not scale well to large numbers of molecules, and it does not help in conditions where modifications are systematically applied to molecules, e.g. in computational combinatorial chemistry.

In the past decades, the problem of assigning bond orders automatically has been addressed by a number of different approaches. Early methods in the field strongly rely on the correctness of atomic coordinates and focus on reference bond lengths and valence angles [BH92], or additionally consider functional group detection [HRB97] and further molecular features like hybridization states and charges [vABF$^+$96, ZCW07]. The main drawbacks of those approaches are the dependence on correct atomic coordinates and their heuristic nature.

In constrast, exact solvers proposed previously represent the bond order assignment problem as a Maximum Weighted Matching for nonbipartite graphs [Lab05] or as an integer linear programming problem that generates valid Lewis structures (electron dot structures) with minimal formal charge on each atom [FH05].

Recently, Wang et al. [WWKC06] have presented an elegant novel approach to the problem which is implemented in the established Antechamber package, a suite of tools used for the preparation of input structures for molecular mechanics studies. In this approach, a chemically motivated, expert generated penalty function is used to score bond order assignments. This function is then heuristically optimized. However, this procedure has two drawbacks: the score of resulting assignment is not guaranteed to be optimal and the algorithm provides only one solution while there can be more than one assignment with optimal score. In this work, we propose an approach that solves the problem to provable global optimality by discrete optimization techniques. We give an integer linear program formulation for very efficient computation of one optimal assignment and an A* approach for enumerating all optimal or, if desired, all feasible solutions.

2 Methods

The idea behind the bond order assignment algorithm proposed in the work of [WWKC06] is to cast it into a discrete optimization problem. Finding the most probable consistent bond order assignment for a given molecule is addressed by minimizing a total penalty score tps, where each atom is assigned an atomic valence av that is defined as the sum over all bond orders bo of all bonds connected to the atom under consideration:

$$av = \sum_{i=1}^{con} bo_i$$

Here, con denotes the number of bonded atoms. The distance of the calculated av to the atom's most desirable valence value is measured by the atomic penalty score aps: the possible valences of an atom and the corresponding distance penalty scores are stored in a penalty table that uses a rule-based atom type classification derived by Wang et al. The sum over all atomic penalty scores of a molecule now yields the total penalty score

$$tps = \sum_{i=1}^{n} aps_i$$

where n denotes the number of atoms. The smaller the tps of a given bond order assignment, the more reasonable it is. In [WWKC06], minimization now proceeds in a heuristic and greedy manner.

2.1 Integer Linear Program (ILP)

To compute a bond order assignment with guaranteed globally minimal tps, we formulated the aforementioned problem as an integer linear program [PS98] as described below.

Let P be the penalty table. We use the following notations:

- A is the set of all atoms of the molecule under consideration.
- $B(a)$ is the set of bonds of atom $a \in A$ and B denotes the set of all bonds of the molecule.
- $V(a) \subset \mathbb{N}$ contains the possible valences of atom $a \in A$ according to the penalty table P.
- $P(a, v)$ is the entry of P for atom $a \in A$ and valence $v \in V(a)$.

Our approach uses two different classes of variables. For each bond $b \in B$, we introduce a variable $x_b \in \{1, \ldots, \mu\}$, where μ is the maximum bond order considered (in the following, we will set μ to 3, allowing single, double, and triple bonds). For all atoms a and corresponding possible valences v according to the penalty table P we introduce choice

variables $y_{a,v} \in \{0,1\}$. Each $y_{a,v}$ symbolizes whether the corresponding penalty $P(a,v)$ is chosen or not, i.e., penalty $P(a,v)$ contributes to the score iff $y_{a,v} = 1$. Thus, the objective function of our score minimization problem can be formulated as a linear function in $\mathbf{y}$ with penalty prefactors:

$$\min_{\mathbf{y}} \quad \sum_{a \in A} \sum_{v \in V(a)} P(a,v) \cdot y_{a,v}.$$

To ensure that each atom is assigned exactly one valence state, we add the additional linear constraints

$$\sum_{v \in V(a)} y_{a,v} = 1$$

for all $a \in A$. In addition, we have to ensure that the sum of its bond orders equals its chosen valence. The constraints can be formulated as

$$\sum_{v \in V(a)} y_{a,v} \cdot v = \sum_{b \in B(a)} x_b$$

for all $a \in A$, because the left hand side evaluates to valence v iff $y_{a,v} = 1$.

In summary, the score minimization problem can be formulated as the following integer linear program

$$\begin{aligned}
\min_{\mathbf{x},\mathbf{y}} \quad & \sum_{a \in A} \sum_{v \in V(a)} P(a,v) \cdot y_{a,v} \\
\text{s.t.} \quad & \sum_{v \in V(a)} y_{a,v} \cdot v = \sum_{b \in B(a)} x_b && \forall a \in A, \\
& \sum_{v \in V(a)} y_{a,v} = 1 && \forall a \in A, \\
& y_{a,v} \in \{0,1\} && \forall a \in A, \forall v \in V(a), \\
& x_b \in \{1,2,3\} && \forall a \in A, \forall b \in B(a).
\end{aligned}$$

For the solution of ILPs to provable global optimality, several strategies can be chosen, like the popular pure branch & bound approaches or branch & cut methods [PS98]. We employed the open source solver lp_solve [BEN] which uses a simplex-algorithm-based branch & bound approach [PS98]. It is interesting to note that the penalties in [WWKC06] can all be expressed as powers of two and as such led to short computation times. Still, the problem itself is NP complete [PS98]. Empirically, however, in many test cases the solution of the relaxed linear program, i.e., the above program without the integrality constraints, has been integral and, hence, a solution of the original problem (obtained without any branching). In other cases, the solution of the linear program has been almost integral, leading to only few branch steps. In principle, ILP solvers can also enumerate all optimal solutions. However, in our experiments we have seen a drastic increase in runtime if more than one solution is computed. Thus, the ILP approach is particularly well suited for obtaining *one* optimal bond order assignment.

2.2 The A* approach

In order to be able to efficiently enumerate *all* feasible solutions – optimal and non-optimal ones – we formulated the bond order total penalty minimization problem as an A* search algorithm. This allows enumeration of all assignments in the order of increasing penalty and hence, for instance, to compare the assignments of all solutions for a given molecule up to a user defined penalty threshold. In addition, such an A* algorithm is simpler to implement, and often easier to extend, than an ILP approach; for instance, it is easily possible to influence the order in which solutions with equal score are computed.

As a combinatorial optimization problem, the bond order assignment problem can be represented by a tree, where each layer stands for one of the decisions that have to be made. In our case, the tree has k layers, where k is the number of bonds that have to be assigned. A node at layer i has μ children, where μ is the number of possible bond orders, typically 3, and each edge is labeled with its corresponding order. Hence, by tracing the path from the root to a node w at layer i, we can determine the values of the first i bonds in this particular partial assignment represented by the node w. Thus, the root node corresponds to a completely unassigned molecule with only unknown bond orders, while the leave nodes correspond to complete bond order assignments. If we only add child nodes if the resulting valence state is valid the leaf nodes correspond to the feasible bond order combinations. In order to discriminate between the different combinations, each leaf is assigned its atomic penalty score.

Visiting all nodes in the tree, the optimal bond order assignment can be found in a brute–force manner with exponential runtime. If, additionally, all intermediate nodes are assigned the atomic penalty score of the partial bond order assignment they represent, a greedy search will yield an assignment with heuristically good (but not necessary optimal) atomic penalty score in linear runtime. It can be shown that, if at each intermediate node more information is provided, finding an optimal solution can be guaranteed with greatly improved expected runtime. This leads to the popular A*-search-algorithm [HNR68], which employs a search heuristic to guide the algorithm in descending the tree. More formally, the algorithm associates with each node w a function $f(w) = g^*(w) + h^*(w)$, where $g^*(w)$ describes the score corresponding to the decisions already made and $h^*(w)$ is the so-called search heuristic. For the purposes of the A*-search algorithm, the search heuristic must be an admissible estimate of the score of the best leaf that can be reached starting from node w and descending further down the tree. Here, admissible means that it needs to be 'optimistic': for all nodes w, the estimated cost $h^*(w)$ may never be greater than the lowest real cost to reach a goal node. Given the additional information provided by h^*, the A*-search algorithm always expands one of the nodes with the most promising score, ensuring that the first leaf reached is optimal (roughly speaking, if the algorithm would visit a leaf with worse score first, the search-heuristic would have overestimated the penalty of the real optimal solution, which an admissible heuristic never does).

In addition to the notations introduced in the previous section, we need notations that are adapted to the partial bond order assignments corresponding to each node w in the search tree. We denote the set of all assigned bonds in the node w by $W(B)$, the assigned bonds connected to atom a in node w by $W(a)$, and the set of atoms for which all bonds are

already assigned with a bond order by K. The bond order of an assigned bond is denoted by $bo(b)$. A partial bond order assignment induces a simple lower bound

$$v_w(a) := \sum_{b \in W(a)} bo(b)$$

for the valence of atom a. Assuming a single bond for each unassigned bond of atom a, a tighter lower bound for the valence is given by

$$lo(a) := v_w(a) + \sum_{b \in B(a) \setminus W(a)} 1 = v_w(a) + |B(a) \setminus W(a)|.$$

Thus, the maximum order of an unassigned bond with respect to atom a is given by

$$t(a) := \max\{V(a)\} - lo(a) + 1.$$

Denoting by a_1, a_2 the atoms connected by an unassigned bond b, its maximum bond order equals

$$bo_{max}(b) := \min\{t(a_1), t(a_2)\},$$

yielding an upper bound of the atomic valence of an atom a

$$up(a) := \min \left\{ \max\{V(a)\}, \quad v_w(a) + \sum_{b \in B(a) \setminus W(a)} bo_{max}(b) \right\}.$$

The functions g^* and h^* can then be defined as follows:

$$g^* = \sum_{a \in K} P\left(a, v_w(a)\right) \tag{1}$$

$$h^* = \sum_{a \in A \setminus K} \min_{lo(a) \le i \le up(a)} \left\{ P\left(a, i\right) \right\}. \tag{2}$$

The function g^* sums the atomic penalties of all completely assigned atoms in the partial bond order assignment represented by node w, whereas h^* considers all atoms with bonds of unassigned bond order. For the atoms in this set, we compute the minimal atomic penalty possible under the current partial assignment independently of the other atoms in the set: each atom can choose its preferred value for each unassigned bond without considering overall consistency. Obviously, h^* is optimistic.

3 Results

We have implemented and integrated both approaches in the Biochemical Algorithms Library BALL (http://www.ball-project.org, [KL00]). For validating our algorithms, we

molecule	score		number of
	Antechamber	BALL	optimal solutions
DAKCEX.mol2	1	0	2
GETFIU.mol2	1	0	1
GIDMEL.mol2	2	1	7
KEWJIF.mol2	4	0	1
SAFKAL.mol2	1	0	1
JECYIZ.mol2	4	0	1

Table 1: Comparison of the penalties for molecules of the MMFF94 validation suite, where BALL found bond order assignments with smaller penalty score than the assignment heuristically computed by Antechamber.

method	reference is		no solution
	1st solution	optimal	
Antechamber	282 (37.05%)	282 (37.05%)	18 (2.36%)
ILP	401 (52.69%)	401 (52.69%)	4 (0.53%)
A*	473 (62.15%)	599 (78.71%)	4 (0.53%)

Table 2: Performance of the original Antechamber implementation, our ILP formulation and our A*-search algorithm on the MMFF94 validation suite. The second column denotes the number of molecules for which the algorithms return the original bond order assignment as first solution. The third column denotes the number of cases, where the reference bond order assignment was within the solutions with minimal tps (if this is not the case, we need to change the objective function rather than the optimization method to correctly address this molecule). Finally, the fourth column denotes the number of molecules for which no solution was found.

chose to compare the computed results on the MMFF94 validation suite [Hal96]. The MMFF94 Suite contains 761 thoroughly prepared drug like molecules that were originally used for the validation of the Merck Molecular Force Field. We used the penalty table as defined in Wang et al. [WWKC06]. On this data set, A* and ILP had comparable runtimes if generating single solutions only (≈220 seconds for the whole set on a standard PC, where the majority of the time is spent in SMARTS matching).

As can be seen in Tab. 2, both of our methods are able to correctly reproduce significantly more molecules of the MMFF94 validation suite than the original Antechamber approach by Wang et al. In cases where the reference molecule is the only possible assignment with minimal tps, ILP and A* both find the optimal bond order assignment, whereas Antechamber returns non-optimal solutions in 6 cases as shown in Tab. 1.

The difference between the performance of ILP and A*-search are due to fact that the MMFF94 validation suite contains 348 molecules with more than one optimal bond order assignment (with respect to the penalty table of Wang et al.) and that the ILP solver systematically prefers assignments different to the A*-search algorithm. The A*-search always prefers lower bond orders which seems to be the more natural behaviour.

As can also be seen in Tab. 2, the enumeration of all optimal solutions leads to a success rate of 78.71% in reproducing the bond order assignments of the MMFF94 validation suite.

However, it should be kept in mind that in reality, bond order assignment for a single molecule need not have a unique solution; for instance, molecules like benzene show several resonance structures, differing only in their bond order configuration (if aromatic bonds are 'kekulized', i. e. replaced by a compatible pattern of single and double bonds, as needed for most force fields).

Obviously, the quality of the penalty table, e.g., the definition of the atom classes, their allowed valence states, and the choice of the valence state's penalties have a significant influence on the performance of our algorithms. As can be seen in column four of Tab. 2, the current penalty table does not cover all molecules in the MMFF94 validation suite – for four molecules, the required atom classes are missing. Please note that the difference to the Antechamber bailing out rate is a result of the heuristic nature of the optimization proposed in [WWKC06].

4 Conclusion

In this work, we have presented two exact solvers for the connectivity based bond order assignment problem posed by Wang et al. [WWKC06]. Both methods improve considerably upon earlier approximate solution schemes by guaranteeing optimality while retaining highly efficient runtimes.

Our ILP-formulation allows for very rapid computation of an optimal bond order assignment with respect to the underlying penalty tables. In our implementation, the ILP is solved directly by the open source solver lp_solve [BEN]. This approach scales well with increasing number of atoms and bonds and should be preferred if only one optimal assignment is sought. However, when computing more than one solution with the ILP solver, runtimes greatly deteriorated.

In these cases, our A*-approach usually has much better runtime, in particular when enumerating all solutions – optimal and non-optimal ones sorted by their score. In addition, the order in which solutions are returned can be easily influenced. Thus, it has the potential to create ensembles of putative bond order assignments, opening new avenues for probabilistic structure analysis. Furthermore, the A*-search algorithm is simple to implement and independent of external solvers.

So far, only connectivity based information is scored in the search heuristic. The inclusion of structural properties like bond lengths and angles might help to further distinguish between assignments if atomic coordinates are reliable. For large molecules, the employment of more sophisticated optimization techniques as presented in [BBST09] might help to speed up computation times.

Both approachs are fully integrated into the upcoming version of the Biochemical Algorithms Library BALL (http://www.ball-project.org, [KL00]) that can be downloaded from our homepage.

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):380–388, Jun 2002.

[BBST09] S. Böcker, Q. B. A. Bui, P. Seeber, and A. Truss. Computing Bond Types in Molecule Graphs. In *Proc. of Computing and Combinatorics Conference (COCOON 2009)*, 2009. To be presented.

[BEN] M. Berkelaar, K. Eikland, and P. Notebaert. lp_solve 5.5. http://lpsolve.sourceforge.net/.

[BH92] J. C. Baber and E. E. Hodgkin. Automatic assignment of chemical connectivity to organic molecules in the Cambridge structural databa. *J Chem Inform Comput Sci*, 32:401–406, 1992.

[BHN03] H. Berman, K. Henrick, and H. Nakamura. Announcing the worldwide Protein Data Bank. *Nat Struct Biol*, 10(12):980, Dec 2003.

[FH05] M. Froeyen and P. Herdewijn. Correct bond order assignment in a molecular framework using integer linear programming with application to molecules where only non-hydrogen atom coordinates are available. *J Chem Inf Model*, 45(5):1267–1274, 2005.

[Hal96] T.A. Halgren. MMFF VI. MMFF94s option for energy minimization studies. *J Comp Chem*, 17:490–519, 1996.

[HNR68] P. E. Hart, N. J. Nilsson, and B. Raphael. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Transactions on Systems Science and Cybernetics SSC4*, 4:100–107, 1968.

[HRB97] M. Hendlich, F. Rippmann, and G. Barnickel. BALI: Automatic assignment of bond and atom types for protein ligands in the Brookhaven Protein Databank. *J Chem Inform Comput Sci*, 37:774–778, 1997.

[KL00] O. Kohlbacher and H. P. Lenhof. BALL–rapid software prototyping in computational molecular biology. Biochemicals Algorithms Library. *Bioinformatics*, 16(9):815–824, Sep 2000.

[Lab05] P. Labute. On the perception of molecules from 3D atomic coordinates. *J Chem Inf Model*, 45(2):215–221, 2005.

[PS98] C. H. Papadimitriou and K. Steiglitz. *Combinatorial Optimization: Algorithms and Complexity*. Courier Dover Publications, 1998.

[vABF+96] D. M. van Aalten, R. Bywater, J. B. Findlay, M. Hendlich, R. W. Hooft, and G. Vriend. PRODRG, a program for generating molecular topologies and unique molecular descriptors from coordinates of small molecules. *J Comput Aided Mol Des*, 10(3):255–262, Jun 1996.

[WWKC06] J. Wang, W. Wang, P. A. Kollman, and D. A. Case. Automatic atom type and bond type perception in molecular mechanical calculations. *J Mol Graph Model*, 25(2):247–260, Oct 2006.

[ZCW07] Y. Zhao, T. Cheng, and R. Wang. Automatic perception of organic molecules based on essential structural information. *J Chem Inf Model*, 47(4):1379–1385, 2007.

Maximum Likelihood Estimation of Weight Matrices for Targeted Homology Search

Peter Menzel[1,2,*], Jan Gorodkin[1], and Peter F. Stadler[2−6]
[1]Division of Genetics and Bioinformatics, IBHV, University of Copenhagen,
Grønnegårdsvej 3, DK-1870 Frederiksberg, Denmark
[2]Bioinformatics Group, Department of Computer Science,
and Interdisciplinary Center for Bioinformatics,
University of Leipzig, Härtelstrasse 16-18, D-04107 Leipzig, Germany.
[3]Max Planck Institute for Mathematics in the Sciences,
Inselstrasse 22, D-04103 Leipzig, Germany
[4]Fraunhofer Institut für Zelltherapie und Immunologie,
Perlickstraße 1, D-04103 Leipzig, Germany
[5]Institute for Theoretical Chemistry, University of Vienna,
Währingerstrasse 17, A-1090 Vienna, Austria
[6]The Santa Fe Institute, 1399 Hyde Park Rd., Santa Fe, New Mexico
*Corresponding Author

Abstract: Genome annotation relies to a large extent on the recognition of homologs to already known genes. The starting point for such protocols is a collection of known sequences from one or more species, from which a model is constructed – either automatically or manually – that encodes the defining features of a single gene or a gene family. The quality of these models eventually determines the success rate of the homology search. We propose here a novel approach to model construction that not only captures the characteristic motifs of a gene, but are also adjusts the search pattern by including phylogenetic information. Computational tests demonstrate that this can lead to a substantial improvement of homology search models.

Introduction

Homology search is one of the generic important tasks in bioinformatics. It is indispensable, e.g., for the assessment of the phylogenetic distribution of genes and gene families and it forms the basis for detailed phylogenetic analyses in general. Homology search also comprises the first step in gene annotation pipelines. The ever increasing influx of genomic sequence data makes reliable and automated homology search a crucial bottleneck in many projects.

Typically, the starting point for homology search is a collection of known sequences, usually in the form of a multiple sequence alignment. Then, one or all of these "seed sequences" are fed into a pairwise alignment algorithm – such as `blast` [MM04] – and compared to the sequence database of the target species. In many cases, e.g. for distant

homologs or short query sequences, the sensitivity of this approach is too low. In such cases one can determine from the alignment the sites that share the same residues in all or most of the seed sequences. These highly conserved sequence blocks typically comprise the specific biological function of the gene – like binding site motifs, catalytically active sites, or structural elements. Once identified, these blocks can be used to build a more sophisticated search pattern that contains the intrinsic properties of this particular gene. The `fragrep` approach, for instance, represents the query as a collection of short consensus patterns and distance constraints between them [MSS06]. Again, restricting oneself to the consensus sequence information of the blocks may lead to a rather low sensitivity or specificity of the search pattern. This is the case e.g. for DNA binding sites [Sto00], which not necessarily share a common consensus sequence.

More expressive sequence models can be build with position specific scoring matrices (PSSM), which record the relative frequencies of residues at each site. The application of PSSMs for homology search requires more elaborate profile alignment algorithms. An example for proteins is `psi-blast` [AMS+97]. For short, ungapped, PSSMs arising e.g. as models of transcription factor binding sites, a relative scoring scheme is used [KGR+03], which can be extended to the gapped case by means of fractional programming [MCS07]. Hidden Markov Models are a viable alternative. In many cases, the highly variable gap sizes and the small set of seed sequences are problematic for the training procedures. PSSM-based approaches therefore were instrumental in several recent studies on highly variable ncRNA families such as Y RNAs [MGSS07], vault RNAs [SCH+09], and telomerase RNAs [XMQ+08].

While theoretically straightforward, the construction of reliable PSSMs from sequence alignments turns out to be a quite non-trivial task. In principle, one just has to count the frequency of the residues in the alignment columns, decide on a scheme to treat gap characters, and possibly add pseudo-counts. In practice, however, one has to deal with biases in the phylogenetic distribution of the seed sequences, which are often dominated by a set of closely related model organisms. The small size of the seed set, on the other hand, makes it undesirable to exclude a large fraction of the available data. A commonly used remedy is to use one of several weighting schemes [VS93]. For amino acid sequences more sophisticated methods for creating unbiased PSSMs are available, e.g. via the `EasyPred` web server [Nie]. Such unbiased "centroid" PSSMs, however, still do not include all the available phylogenetic information, in particular, they do not take into account any knowledge on the relative phylogenetic position of the target genome among the aligned seed sequences.

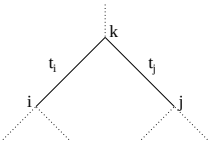
In this contribution we therefore explore the possibility to employ a maximum likelihood (ML) approach to optimize search patterns for usage on a particular target. Our approach is similar in spirit to the reconstruction of ancestral sequences from their extant offsprings. Given a phylogenetic tree T, ancestral genes are "resurrected" by inferring the states for internal nodes of T given the known sequences at the leafs. The earliest approaches were based on the parsimony principle [Fit71]. Alternatively, maximum likelihood methods, introduced by Felsenstein [Fel81], are in use. The latter require an explicit model of sequence evolution. On the other hand, they naturally provide probability distributions over the amino acid or nucleotide alphabet for every sequence position and every internal node

of the tree. In other words, ML provides us with PSSMs for ancestral states. Compared to parsimony approaches, maximum likelihood methods are more accurate because branch lengths, more detailed residue substitution models, and back-mutations are taken into account [ZJ97]. Ancestral sequence reconstruction has been proven to be a powerful tool for testing hypotheses regarding the function of genes from extinct species, see, e.g., [Tho04].

Here, we modify this approach. Instead of focusing on the internal nodes of the tree T, we use the same mathematical machinery to infer the most likely nucleotide sequence at an additional leaf node in the tree — the target species for homology search.

Construction of Search Patterns

We start from a given multiple sequence alignment M with m sequences and a phylogenetic tree T with $m+1$ leaves, representing the phylogenetic relationships and branch lengths among the m species included in the alignment, and a single additional target species 0. Our approach combines two ML computations. First we use M and $T \setminus 0$, the phylogenetic tree restricted to the aligned species, to estimate for each alignment column i a relative substitution rate $\hat{\mu}_i$ The calculation of the likelihood follows Felsenstein's pruning algorithm [Fel81]. The likelihood of a residue s_k at an interior node k is obtained from the corresponding likelihoods at the two child nodes i and j, which are separated from k by branches of length t_i and t_j, respectively:



$$L_{s_k}(\mu) = \Big(\sum_{s_i} P_{s_k s_i}(t_i,\mu) L_{s_i}(\mu)\Big) \times \Big(\sum_{s_j} P_{s_k s_j}(t_j,\mu) L_{s_j}(\mu)\Big) \qquad (1)$$

For each alignment column i, we numerically optimize $\hat{\mu}_i = \operatorname{argmax}_\mu L_T(\mu)$ using Golden Section Search [Kie53]. The likelihood of the tree T is given by the sum over all possible states s_r at the root node r: $L_T(\mu) = \sum_{s_r} \pi_s L_{s_r}(\mu)$ where the π_s are the prior probabilities of observing letter s. The transition matrix $\mathbf{P}$ contains probabilities $P_{xy}(t,\mu) = [e^{t\mu\mathbf{Q}}]_{xy}$ for changing from state y to state x over time t and a rate μ. The instantaneous rate matrix Q represents a standard substitution model, such as the HKY85 [HKY85] or General Time Reversible (GTR) [Tav86] model for DNA sequences. Parameters for these models can be estimated from the alignment by using standard maximum likelihood analysis software like `PAML` [Yan07]. We advocate that this should be done ideally on larger data sets than the usually short query alignments themselves.

In the second step, we use the estimated values $\hat{\mu}_i$ to compute the probabilities for each residue at the i-th position of the target sequence. To this end, we re-root the original tree T to the target species 0 and then calculate the likelihoods $L_{s_0}(\hat{\mu}_i)$ for T^0. From these likelihoods at the root node of T^0, we directly obtain the residue probabilities in each alignment column i. Finally, these are transformed into a PSSM.

Figure 1 exemplifies the difference of a PSSM inferred by the ML approach and a PSSM obtained by counting the nucleotide frequencies in the seed alignment. In this particular

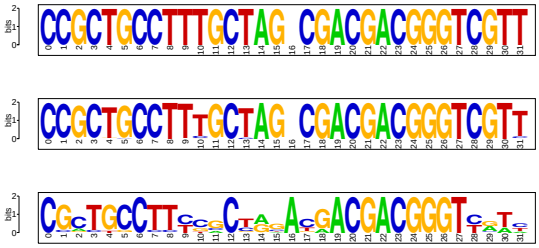


Figure 1: Example for estimating a PSSM. **top:** Target sequence in the 5' region of the 7SK RNA of *Drosophila persimilis*. **middle:** Maximum likelihood estimated nucleotide probabilities for *D. persimilis*. **bottom:** PSSM derived from nucleotide frequencies of the 11 other Drosophila sequences.

case, the ML estimate is significantly more informative and much closer to the motif in the target sequence.

The ML-PSSM pattern depends explicitly on the relative position of the target species in T. If the target is in close proximity to one or more other species, then high probabilities will be assigned to the residues that are present in those neighboring species. With increasing distance to the target species, on the other hand, the probabilities will converge to an uninformative equilibrium distribution. A column equilibrates faster, the larger the substitution rate $\hat{\mu}_i$. The algorithm thus tells us, which alignment columns or regions can be expected to be informative for a particular target sequence. To this end, we compute the Shannon information of each alignment position as

$$H(i) = -\sum_s f_i(s) \cdot \log_2 f_i(s) \tag{2}$$

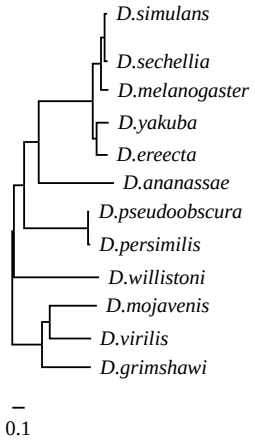
where $f_i(s)$ is the estimated frequency of residue s at position i. The corresponding information content is $I(i) = \bar{H} - H(i)$, where $\bar{H} = -\sum_s \bar{f}(s) \log_2 \bar{f}(s)$ and $\bar{f}(s)$ is the background distribution of the residues. In the simplest case, $\bar{H} = 2$ for an uniform distribution of the four nucleotides.

Significant patterns can now be extracted by finding windows of a user-defined minimum length that have an average information content above a certain threshold. Alignment columns with high estimates of $\hat{\mu}$, on the other hand, can be excluded from the search pattern to compensate for highly variable sites. Thus, the maximum likelihood algorithm not only provides residue probabilities for each alignment position, but also gives information about the conserved sites and the variation of mutation rates within one sequence. We remark that our approach of optimizing the $\hat{\mu}_i$ is similar to the method used in the `Rate4Site` program [PBM+02], which aims at identifying functional important regions in protein surfaces.

Performance Evaluation

As test data we used a collection of genomic `multiz` alignments of Drosophila species [Con07] downloaded from the UCSC Genome Browser[1]. Only segments covering all 12

[1] `http://hgdownload.cse.ucsc.edu/goldenPath/dm3/multiz15way/`



Species	Data set 1			Data set 2		
	ML	Freq	Δ	ML	Freq	Δ
D. sim.	1.000	0.981	0.019	1.000	0.980	0.020
D. sec.	1.000	0.981	0.019	1.000	0.975	0.025
D. mel.	0.986	0.979	0.007	0.970	0.972	-0.002
D. yak.	0.970	0.971	-0.001	0.963	0.959	0.003
D. ere.	0.971	0.972	-0.001	0.959	0.959	0.000
D. ana.	0.896	0.885	0.011	0.841	0.842	-0.001
D. pse.	1.000	0.933	0.067	1.000	0.867	0.133
D. per.	1.000	0.928	0.072	1.000	0.865	0.135
D. wil.	0.912	0.890	0.022	0.774	0.765	0.009
D. moj.	0.912	0.882	0.030	0.838	0.772	0.066
D. vir.	0.913	0.891	0.022	0.858	0.787	0.071
D. gri.	0.877	0.864	0.013	0.824	0.759	0.065

Figure 2: **left:** Phylogenetic tree of the 12 Drosophila species [Con07]. **right:** Median match scores of the maximum likelihood PSSMs (ML) and the frequency PSSMs (Freq) for 10 randomly selected 30nt windows from each alignment in both data sets.

drosophilid species were retained and gapped columns excluded. *Set1* consists of the 56 alignment segments of *D. melanogaster* chromosome 4 with minimum length 500 and a `multiz` score of at least 10000. The average pairwise sequence identity is 76.1%. *Set2* contains 45 alignments with `multiz` scores between 100 and 10000 and minimum length of 200. This set has 67.1% average sequence identity.

We removed one sequence at a time from the alignment and computed the residue probabilities for this sequence with our ML approach from the 11 remaining sequences using the phylogenetic tree in figure 2 and the HKY85 substitution model. The transition bias parameter κ was estimated using `PAML`. For comparison, we computed the position frequency matrix from the same 11 species. Both results were converted to a PSSM. From each alignment we randomly selected 10 windows of different lengths. The `MATCH` scores [KGR$^+$03] of the corresponding interval of the two PSSMs against the 12th aligned sequence that was excluded from training were computed using `pwmatch`[2] [TBF$^+$07]. Then we compared the match scores of each pair of PSSMs and used the Wilcoxon rank-sum test to see if the maximum likelihood (ML) scores are significantly larger than the scores from the frequency method (Freq).

Figures 3 and 4 show the `MATCH` scores of each pair of PSSMs for windows of length $L = 30$ for *Set1* and *Set2* for a representative subset of the 12 drosophilid species. Overall, we observe that the ML matrices have significantly higher `MATCH` scores than the frequency matrices for most of the target species. The difference is especially apparent for those drosophilids that have a closely related neighbor in the phylogenetic tree, such as *D. simulans* and *D. sechellia* or *D. pseudoobscura* and *D. persimilis*. Here the median `MATCH` score improvement is up to 0.076 for *D. persimilis* in *Set1* and 0.135 in *Set2*. Only for *D. ananassae* and *D. willistoni* there is no significant difference of the scores in *Set2* where both the ML and Freq PSSMs perform equally and only a slight average improve-

[2] `http://www.bioinf.uni-leipzig.de/Software/pwmatch/`

ment of the ML PSSMs is visible in *Set1*. Due to the relatively large distance from all other species, and the relatively even distribution of the species in the tree, the frequency-based matrix scores are very similar to the ML estimate in these two cases. Generally, the improvement of the MATCH scores is higher in *Set2*, which has lower sequence identity For instance, the average score difference of both methods in *D. pseudoobscura* is 0.067 in *Set1* and 0.133 in *Set2*, where the median score of the frequency method is much lower than in *Set1*.

For homology search, short blocks with high information content are of particular importance, since such queries can be searched most efficiently. Thus, we extracted from both data sets those sub-patterns containing columns with high information content at most positions. Figure 5 summarizes the MATCH scores of the ML and the frequency PSSMs for all (non-overlapping) windows of length 20nt which have an average information content of at least 1.8 bits in the ML matrices. For these patterns, we observe again that the ML approach performs significantly better for most target species. For some species, only few windows fulfilling these constraints can be found, e.g. *D. ananassae* (n=23) or *D. willistoni* (n=27). Due to the relatively large distance to the other drosophilids, the ML algorithm assigns high residue probabilities only to highly conserved alignment columns. Eventually, these probabilities are very similar to the nucleotide frequencies in the seed alignment and the performance of ML and frequency approach becomes indistinguishable.

Due to the close phylogenetic relationship of *D. simulans* and *D. sechellia*, and *D. pseudoobscura* and *D. persimilis*, resp., the ML approach estimates very high nucleotide probabilities for these target species Thus many windows with high average information content can be found. Compared to the frequency PSSMs, the ML PSSMs provide a big performance improvement in these species.

Discussion

In this contribution we presented a novel approach for constructing PSSM-like sequence models for homology search. Unlike standard methods, our maximum likelihood method aims at building models that are specifically adapted to a particular target species. This is achieved by utilizing the phylogenetic information of the seed sequences and the relative position of the target species therein.

Evaluation on genomic sequence alignments of the 12 sequenced drosophilid species shows that the maximum likelihood method indeed provides the expected improvements. We are able to find highly conserved sites in the alignment and make use of the sequence information from neighboring species in the phylogenetic tree. The more proximal a known sequence is related to the target species, the more specific the search pattern from the maximum likelihood computation becomes, even for randomly drawn samples. If the target species is evolutionary distant in the tree from the known taxa, the alignment sites with high information content can be used for building the search pattern and the specificity is better or the same compared to normal search patterns based on residue frequencies.

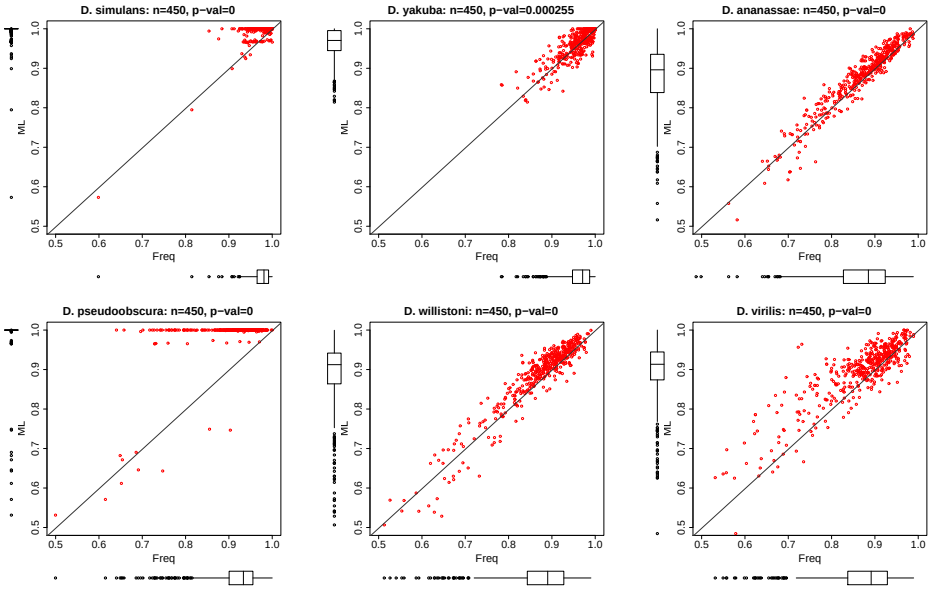


Figure 3: *Set 1*: `MATCH` scores of maximum likelihood (ML) and frequency (Freq) PSSMs for random windows of length 30nt ($n = 450$). P-values of "0" are smaller than machine precision.

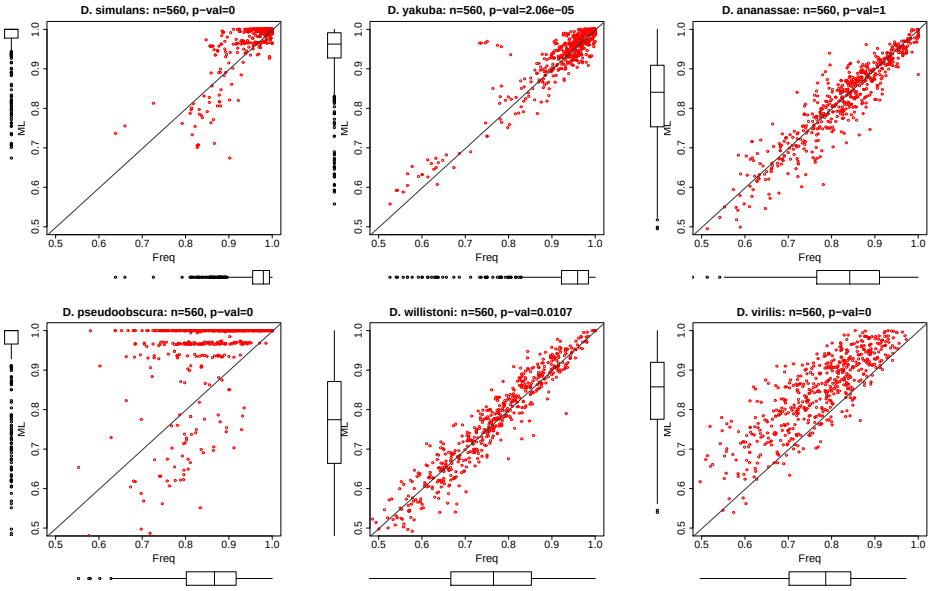


Figure 4: *Set2*: `MATCH` scores for maximum likelihood (ML) and frequency (Freq) PSSMs for random windows of length 30nt ($n = 450$). P-values of "0" are smaller than machine precision.

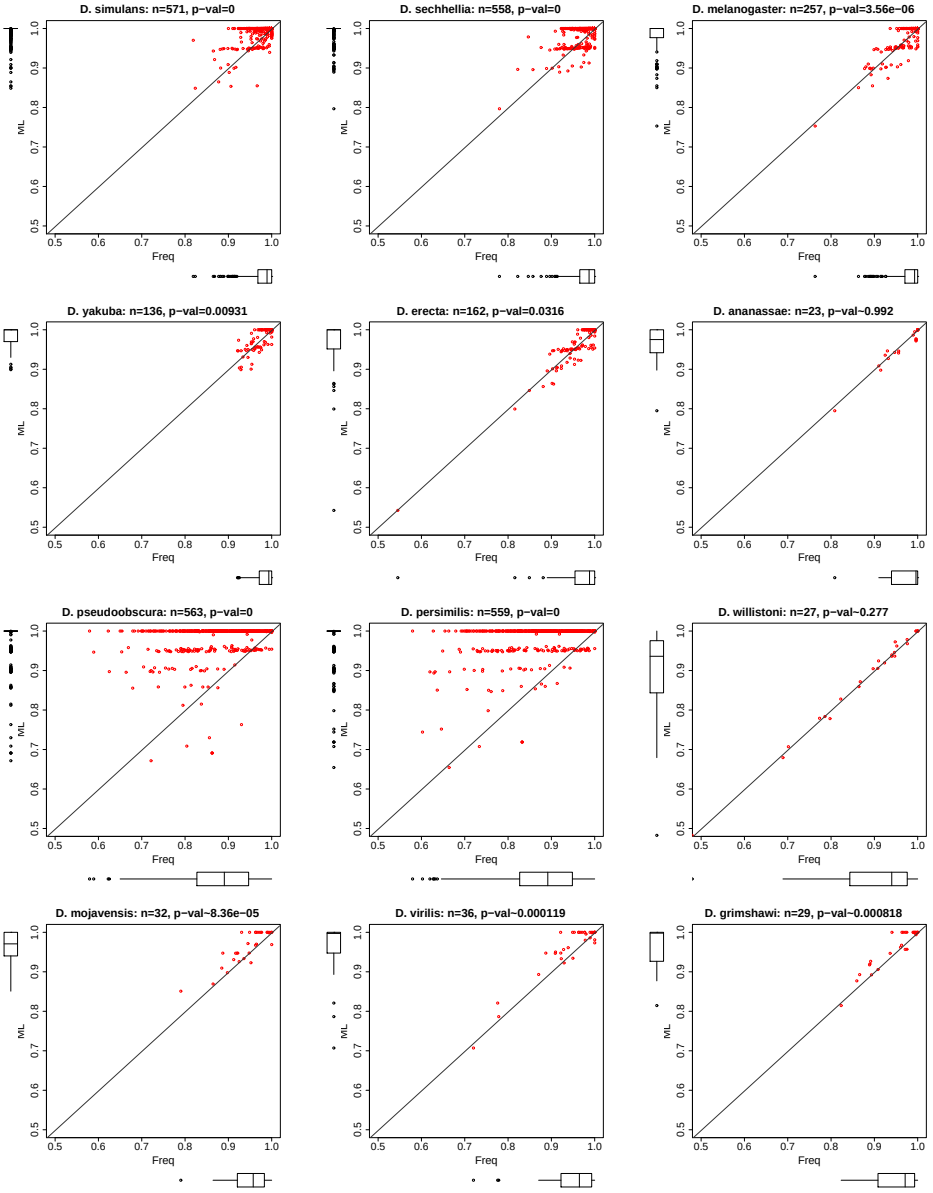


Figure 5: *Set2*: MATCH scores for the ML and frequency-based PSSMs for all non-overlapping windows of length 20 with average information content $H \geq 1.8$. In a few cases (indicated by $p \sim$ instead $p =$) the p-value estimates are approximations due to the small sample size.

The approach proposed here is potentially useful not only for the purely sequence-based homology search. In particular for structured RNAs it seems natural to incorporate phylogenetic information also into covariance models such as those utilized by SCFG-based tools. To this end, base pair substitution models for paired alignment columns need to be incorporated. We expect that this will be helpful in the detection of conserved structural elements in ncRNA families as well as aiding in automatic estimation of highly probable structure motifs in a target species. A second issue that needs to be addressed in future work is the handling of gaps, which we excluded here for the sake of clarity. In the simplest case, the approach of `fragrep` [MCS07] provides a remedy.

Acknowledgement. PM is supported by the Danish research council for Technology and Production through and the Danish research school in biotechnology. This work was supported by the Danish Center for Scientific Computation.

References

[AMS+97] S. F. Altschul, T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang, W. Miller und D. J. Lipman. Gapped `BLAST` and `PSI-BLAST`: a new generation of protein database search programs. *Nucleic Acids Res.*, 25:3389–3402, 1997.

[Con07] Drosophila 12 Genomes Consortium. Evolution of genes and genomes on the Drosophila phylogeny. *Nature*, 450:203–208, 2007.

[Fel81] Joseph Felsenstein. Evolutionary trees from DNA sequences: A maximum likelihood approach. *Journal of Molecular Evolution*, 17(6):368–376, November 1981.

[Fit71] W.M. Fitch. Toward defining the course of evolution: minimum change for a specific tree topology. *Systematic Zoology*, 20:406–416, 1971.

[HKY85] M. Hasegawa, H. Kishino und T. Yano. Dating the human-ape split by a molecular clock of mitochondrial DNA. *Journal of Molecular Evolution*, 22:160–174, 1985.

[KGR+03] A.E. Kel, E. Gossling, I. Reuter, E. Cheremushkin, O.V. Kel-Margoulis und E. Wingender. MATCH: a tool for searching transcription factor binding sites in DNA sequences. *Nucl. Acids Res.*, 31(13):3576–3579, 2003.

[Kie53] J. Kiefer. Sequential Minimax Search for a Maximum. *Proceedings of the American Mathematical Society*, 4(3):502–506, 1953.

[MCS07] Axel Mosig, Julian L. Chen und Peter F. Stadler. Homology Search with Fragmented Nucleic Acid Sequence Patterns. In *WABI 2007 (R. Giancarlo & S. Hannehalli, eds.)*, Seiten 335–345, 2007.

[MGSS07] Axel Mosig, Meng Guofeng, Brbel M. R. Stadler und Peter F. Stadler. Evolution of the Vertebrate Y RNA Cluster. *Th Biosci.*, 126:9–14, 2007.

[MM04] Scott McGinnis und Thomas L. Madden. BLAST: at the core of a powerful and diverse set of sequence analysis tools. *Nucl. Acids Res.*, 32(suppl2):W20–25, 2004.

[MSS06] Axel Mosig, Katrin Sameith und Peter F. Stadler. `fragrep`: Efficient Search for Fragmented Patterns in Genomic Sequences. *Geno. Prot. Bioinfo.*, 4:56–60, 2006.

[Nie] Morten Nielsen. EasyPred web server: `http://www.cbs.dtu.dk/biotools/EasyPred/`. website.

[PBM+02] Tal Pupko, Rachel E Bell, Itay Mayrose, Fabian Glaser und Nir Ben-Tal. Rate4Site: an algorithmic tool for the identification of functional regions in proteins by surface mapping of evolutionary determinants within their homologues. *Bioinformatics*, 18 Suppl 1:S71–S77, 2002.

[SCH+09] Peter F. Stadler, Julian J.-L. Chen, Jörg Hackermüller, Steve Hoffmann, Friedemann Horn, Phillip Khaitovich, Antje K. Kretzschmar, Axel Mosig, Sonja J. Prohaska, Xiaodong Qi, Katharina Schutt und Kerstin Ullmann. Evolution of Vault RNAs. *Mol. Biol. Evol.*, 2009. accepted.

[Sto00] Gary D. Stormo. DNA binding sites: representation and discovery. *Bioinformatics*, 16(1):16–23, 2000.

[Tav86] S Tavare. Some probabilistic and statisical problems on the analysis of DNA sequences. *Lect. Math. Life Sci.*, 17:57–86, 1986.

[TBF+07] The Athanasius F. Bompfünewerer RNA Consortium:, Rolf Backofen, Christoph Flamm, Claudia Fried, Guido Fritzsch, Jörg Hackermüller, Jana Hertel, Ivo L. Hofacker, Kristin Missal, Sonja J. Mosig, Axel Prohaska, Domininc Rose, Peter F. Stadler, Andrea Tanzer, Stefan Washietl und Will Sebastian. RNAs Everywhere: Genome-Wide Annotation of Structured RNAs. *J. Exp. Zool. B: Mol. Dev. Evol.*, 308B:1–25, 2007.

[Tho04] Joseph W. Thornton. Resurrecting ancient genes: experimental analysis of extinct molecules. *Nat Rev Genet*, 5(5):366–375, Mai 2004.

[VS93] Martin Vingron und Peter R. Sibbald. Weighting in sequence space: A comparison of methods in terms of generlized sequences. *Proc. Natl. Acad. Sci. USA*, 90:8777–8781, 1993.

[XMQ+08] Mingyi Xie, Axel Mosig, Xiaodong Qi, Yang Li, Peter F. Stadler und Julian J.-L. Chen. Size Variation and Structural Conservation of Vertebrate Telomerase RNA. *J. Biol. Chem.*, 283:2049–2059, 2008.

[Yan07] Ziheng Yang. PAML 4: Phylogenetic Analysis by Maximum Likelihood. *Mol Biol Evol*, 24(8):1586–1591, 2007.

[ZJ97] Nei M. Zhang J. Accuracies of ancestral amino acid sequences inferred by the parsimony, likelihood, and distance methods. *J Mol Evol*, 44:S139–46, 1997.

Index of Authors

Gesellschaft für Informatik e.V. (GI)

publishes this series in order to make available to a broad public recent findings in informatics (i.e. computer science and informa-tion systems), to document conferences that are organized in co-operation with GI and to publish the annual GI Award dissertation.

Broken down into the fields of
• Seminar,
• Proceedings
• Dissertations
• Thematics
current topics are dealt with from the fields of research and development, teaching and further training in theory and practice. The Editorial Committee uses an intensive review process in order to ensure the high level of the contributions.

The volumes are published in German or English

Information: http://www.gi-ev.de/service/publikationen/lni/

ISSN 1617-5468
ISBN 978-3-88579-221-5

This volume contains the papers presented at the German Conference on Bio-informatics, GCB 2009, held in Halle (Saale), September 28-30, 2009. GCB is an annual international conference providing a forum for the presentation of research in Bioinformatics and Computational Biology. The conference is orga-nized on behalf of the FG “Informatik in den Biowissenschaften (BIOINF)” of the German Society of Computer Science (GI), the AG “Computereinsatz in den Biowissenschaften” of the German Society of Chemical Technique and Biotechnology (DECHEMA) and the Studiengruppe “Bioinformatik” of the German Society for Biological Chemistry and Molecular Biology (GBM).