INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Project-Team MODBIO

Computational Models in Molecular Biology

Lorraine

THEME BIO

Table of contents

1. Team **1**
2. Overall Objectives **1**
2.1.1. Research themes 1
2.1.2. Scientific and industrial relations 2
3. Scientific Foundations **2**
3.1. Constraint programming 2
3.1.1. Finite domain constraint programming 2
3.1.2. Concurrent constraint programming 3
3.2. Statistical learning 3
3.3. Combinatorial optimization and integer programming 3
4. Application Domains **4**
4.1. Molecular biology 4
4.2. Crystallography 4
4.3. Operations research 5
5. Software **5**
5.1. M-SVM: Multi-class Support Vector Machine 5
5.2. KOALAB: KOupled Algorithmic and Learning Approach for Biological sequences 5
5.3. SCIL – Symbolic Constraints in Integer Linear Programming (ATIPE) 6
6. New Results **6**
6.1. Integer programming and the phase problem in crystallography 6
6.2. Structural risk minimization inductive principle for multi-class discriminant analysis 6
6.3. Probabilistic automata inference 7
6.4. Protein structure prediction 7
6.5. Search for non-coding RNA genes 7
6.6. SELEX data processing 8
6.7. Modeling of alternative splicing regulation 8
6.8. Metabolic pathways analysis 9
6.9. Constraint programming and integer programming 9
6.10. Multiple sequence alignment by cutting planes (ATIPE) 9
6.11. Approximating k-hop minimum spanning trees (ATIPE) 10
6.12. Computing locally coherent discourses (ATIPE) 10
7. Other Grants and Activities **11**
7.1. Regional projects 11
7.2. National projects 11
7.3. International relations 11
8. Dissemination **11**
8.1. Serving the scientific community 11
8.2. Teaching 11
8.3. Miscellaneous 12
9. Bibliography **12**

1. Team

Team Leader
Alexander Bockmayr [Professor, University Henri Poincaré, Nancy 1, until 11/2004]

Team Vice-Leader
Eric Domenjoud [CR CNRS]

Administrative Assistant
Sophie Drouot [INRIA]

Staff member CNRS
Yann Guermeur [CR]

Ph. D. student
Arnaud Courtois [UHP, cofinanced by the Région Lorraine, until 9/2004; ATER INPL since 10/2004]
Yannick Darcy [Allocataire MENRT]
Damien Eveillard [INRIA et UHP, in collaboration with UMR 7567 MAEM, until 8/2004]
Abdelhalim Larhlimi [UHP, cofinanced by the Région Lorraine, since 11/2004]

Post-doctoral Fellow
Sandrine Schermack-Peyrefitte [CNRS (jointly with UMR 7567 MAEM), and INRIA]
Frédéric Sur [CNRS, since 10/2004]

External Collaborator
Emmanuel Gothié [UMR 7567 MAEM, until 5/2004]
François Denis [Professor, Université de la Méditerranée, Marseille; délégation à l'INRIA since 11/2004]

Student intern
Stéphanie Bonne-Billaut [Master Bioinformatique, Bordeaux, from 4/04 to 9/04]
Abdelhalim Larhlimi [DEA Informatique, Nancy, from 2/04 to 7/04]
Pushpraj Shukla [IIT Kanpur, from 5/04 to 7/04]
Yannick Krause [IUT Charlemagne, Nancy, from 4/04 to 6/04]
Illaurie Mignot [Maîtrise MGMC, Nancy, from 1/04 to 2/04]
Myriam Vezain [DESS EGOISt, Rouen, until 11/04]

Independent French-German research group, ATIPE CNRS-MPG
Ernst Althaus [Group Leader, since 4/04]
Stefan Canzar [PhD student, cofinanced by the Région Lorraine, since 11/04]

2. Overall Objectives

The aim of the project MODBIO is to develop computational models for molecular and cell biology. We are focusing on two types of problems:

- Determining the structure of biological macromolecules,
- Discovering and understanding the function of biological systems.

We approach these questions by combining techniques from constraint programming, combinatorial optimization, hybrid systems, and statistical learning theory.

2.1.1. Research themes

- Sequence and structural alignment, phylogeny
- Determination and analysis of macromolecular envelopes
- Protein structure prediction and protein docking
- Modeling alternative splicing regulation
- Metabolic pathway analysis

2.1.2. Scientific and industrial relations

- Participation in the "Génopole Strasbourg Alsace-Lorraine"
- Participation in the Bioinformatics project of the Région Lorraine
- Participation in the ACI project GENOTO3D
- Participation in the ARC INRIA "Process calculi and molecular networks"
- Various national and international collaborations

  - Laboratoire «Maturation des ARN et Enzymologie Moléculaire» (MAEM), UMR 7567, Nancy
  - Laboratoire de Cristallographie, LCM3B, Nancy
  - Institut de Biologie et Chimie des Protéines, IBCP, Lyon
  - Institut Supérieur d'Agriculture, ISA, Beauvais, France
  - Center for Bioinformatics, Saarbrücken, Germany
  - DFG Research Center Matheon, Berlin, Germany
  - Institute of Mathematical Problems in Biology, Russian Academy of Sciences
  - University of California, Irvine, USA

3. Scientific Foundations

3.1. Constraint programming

Constraint programming [50] is a declarative programming language paradigm that appeared in the late 80's, and which has become more and more popular since then. A *constraint* is a logical formula that defines a relation to be satisfied by the values of the variables the formula contains. For instance, the formula $x + y \leq 1$ expresses that the sum of the values of the variables x and y must be less than or equal to 1.

In *constraint programming*, the user programs with constraints, i.e., he or she describes a problem by a set of constraints, which are connected by *combinators* such as conjunction, disjunction, or temporal operators (`always`). Each constraint gives some *partial* information about the state of the system to be studied. Constraint programming systems allow one to deduce new constraints from the given ones and to compute *solutions*, i.e., values for the variables that satisfy all constraints simultaneously.

One of the main goals of constraint programming is to develop programming languages that allow one to express constraint problems in a natural way, and to solve them efficiently.

3.1.1. Finite domain constraint programming

In our work, we are first interested in constraint problems over finite domains. In this case, the domain of each variable (the set of values it may take) is a finite set of integer numbers. Theory tells us that most constraint problems over finite domains are NP-hard, which means that there is little hope to solve them by algorithms polynomial in the size of the input. In practice, these problems are handled by tree search methods which try successively different valuations of the variables until a solution is found. Because of the exponential number of possible combinations, it is crucial to reduce the search space as much as possible, i.e., to eliminate *a priori* as many valuations as possible.

There exist two generic methods to solve such problems. The first one is classical *integer linear programming* (see also Sect. 3.3), which has been studied in mathematical programming and operations research for more than 40 years. Here, constraints are linear equations and inequalities over the integer numbers. In order

to reduce the search space, one typically uses the linear relaxation of the constraint set. Equations and inequalities are first solved over the real numbers, which is much easier; then the information obtained is used to prune the search tree.

The second method is *finite domain constraint programming* which arose in the last 15 years by combining ideas from declarative programming languages and constraint satisfaction techniques in artificial intelligence. In contrast to integer linear optimization one uses, in addition to simple arithmetic constraints, more complex constraints, which are called *symbolic constraints*. For instance, the symbolic constraint `alldifferent`$(x_1, ..., x_n)$ expresses that the values of the variables $x_1, ..., x_n$ must be pairwise distinct. Such a constraint is difficult to express in a compact way using only linear equations and inequalities. Symbolic constraints are handled individually by specific filtering algorithms that reduce the domain of the variables. This information is propagated to other constraints which may further reduce the domains.

A state-of-the-art survey of finite domain constraint programming, with special emphasis on its relation to integer linear programming can be found in [12], see also Sect. 6.9.

3.1.2. Concurrent constraint programming

In *concurrent* constraint programming (cc) [44], different computation processes may run concurrently. Interaction is possible via the *constraint store*. The store contains all the constraints currently known about the system. A process may *tell* the store a new constraint, or *ask* the store whether some constraint is entailed by the information currently available, in which case further action is taken.

Hybrid concurrent constraint programming (`Hybrid cc`) [38] is an extension of concurrent constraint programming which allows one to model and to simulate the temporal evolution of *hybrid systems*, i.e., systems that exhibit both discrete and continuous state changes. Constraints in `Hybrid cc` may be both algebraic and differential equations. State changes can be specified using the combinators of concurrent constraint programming and default logic. `Hybrid cc` is well-suited to model dynamic biological systems, as shown in [4].

3.2. Statistical learning

Statistical learning theory [48] is one of the fields of inferential statistics the bases of which have been established by V.N. Vapnik in the late 1960s. The goal of this theory is to specify the conditions under which it is possible to «learn» from empirical data obtained by random sampling. Learning amounts to solving a problem of model selection. More precisely, given a problem characterized by a joint probability distribution on couples made up of observations and labels, and a set of functions, of cardinality ordinarily infinite, the goal is to find in the set a function with optimal performance. Problems may belong to one of the three following areas: pattern recognition (discriminant analysis), function approximation (regression) and density estimation.

This theory considers more specifically two inductive principles. The first one, named empirical risk minimization (ERM) principle, consists in minimizing the training error. If the sample is small, one substitutes to this the structural risk minimization (SRM) principle. It consists in minimizing an upper bound on the expected risk (generalization error), a bound sometimes called a guaranteed risk. This latter principle is implemented in the training algorithms of the support vector machines (SVMs), which currently constitute the state-of-the-art for numerous problems of pattern recognition.

SVMs are connectionist models conceived to compute indicator functions, to perform regression or to estimate densities. They have been introduced during the last decade by Vapnik and co-workers [32], as nonlinear extensions of the maximal margin hyperplane [47]. Their main advantage is that they can avoid overfitting in the case where the size of the sample is small [48][29].

3.3. Combinatorial optimization and integer programming

Combinatorial optimization is a lively field of applied mathematics, combining techniques from combinatorics, linear programming, and the theory of algorithms, to solve optimization problems over discrete structures [31]. A combinatorial optimization problem can be defined as follows: we are given a ground set N

and consider a finite collection of subsets, say $\{S_1, S_2, \cdots, S_m\}$. For each subset S_k there is an objective function value, $f(S_k)$, typically a linear function over the elements in S_k. The task is to find the subset S_k that minimizes $f(S_k)$. Typically, the feasible subsets are represented by inclusion or exclusion of members such that they satisfy certain conditions. Well known examples of combinatorial optimization problems are assignment, covering, cutting stock, knapsack, matching, packing, partitioning, routing, sequencing, scheduling (jobs), shortest path, spanning tree, and traveling salesman problems.

This then becomes a special class of integer programs (IP) whose decision variables are binary valued: $x_i = 1$ if the i-th element is in the optimal solution; otherwise, $x_i = 0$. In this case, feasible subsets have to be expressed by linear constraints. IP formulations are not always easy, and often there is more than one formulation, some better than others. Many good formulations have exponential size.

4. Application Domains

4.1. Molecular biology

Participants: Ernst Althaus, Alexander Bockmayr, Stefan Canzar, Arnaud Courtois, Yannick Darcy, Eric Domenjoud, Damien Eveillard, Emmanuel Gothié, Yann Guermeur, Abdelhalim Larhlimi, Sandrine Schermack-Peyrefitte, Frédéric Sur, Myriam Vezain.

Molecular biology is concerned with the study of three types of biological macromolecules: DNA, RNA, and proteins. Each of these molecules can initially be viewed as a string on a finite alphabet: DNA and RNA are nucleic acids made up of nucleotides A,C,G,T and A,C,G,U, respectively. Proteins are sequences of amino acids, which may be represented by an alphabet of 20 letters.

Molecular biology studies the information flow from DNA to RNA, and from RNA to proteins. In a first step, called *transcription*, a DNA string ("gene") is transcribed into messenger RNA (mRNA). In the second step, called *translation*, the mRNA is translated into a protein, where each triplet of nucleotides encodes one amino acid ("genetic code"). During transcription, an intermediate maturation step can occur, which happens mainly in eukaryotic cells. In the so-called *splicing* process, introns are removed from the premessenger RNA. The remaining exons are concatenated yielding the mature RNA molecule.

Biological macromolecules are not just sequences of nucleotides or amino acids. Actually, they are complex three-dimensional objects. DNA shows the famous double-helix structure. RNA and proteins fold into complex three-dimensional structures, which depend on the underlying sequence. RNA is a single-stranded chain of nucleotides. However, a nucleotide in one part of the molecule can base-pair with a nucleotide in another part, following the Watson-Crick complementarity rules. This results in a folding of the molecule. The *secondary structure* of RNA indicates the set of base pairings in the three dimensional structure of the molecule. This information can be represented by a graph.

Proteins have several levels of structure. Above the primary sequence is the *secondary structure*, which involves three basic types: α*-helices*, β*-sheets*, and structure elements that are neither helices nor sheets, called *loops*. A *domain* of a protein is a combination of secondary structure elements with some specific function. It contains an *active site* where an interaction with an external molecule may happen. A protein may have one or several domains.

The ultimate goal of molecular biology is to understand the *function* of biological macromolecules in the life of the cell. Function results from the *interaction* between different macromolecules, and depends on their structure. The overall challenge is to make the leap from sequence, through structure, to understand about the function.

4.2. Crystallography

Participants: Alexander Bockmayr, Eric Domenjoud.

X-ray structure analysis is the main tool to establish the three-dimensional atomic structure of biological macromolecules and their complexes. The determination of a structure in X-ray crystallography passes through several stages:

- purification and crystallization of the object under study (a protein, DNA, RNA, virus, or a huge macromolecular complex, such as ribosome or lipoprotein particles);
- X-ray experiment (usually at synchrotron accelerators); data collection (up to a million of independent observations) and their primary processing;
- the solution of the inverse problem of the theory of diffraction to find the electron density distribution in the studied object and to interpret it in terms of atoms.

A key problem of X-ray structure analysis is the so-called *phase problem*. In an X-ray experiment, one can measure only the magnitudes of the complex Fourier coefficients of the electron density distribution under study, but not their phases. Half of the necessary information is therefore lost, and must be restored by other means.

4.3. Operations research

Participants: Ernst Althaus, Alexander Bockmayr, Eric Domenjoud.

While molecular biology has become the main application area of our work, we continue to study selected problems from other domains, in particular operations research. During this year, we have been working on graph and network design problems, and also on problems from computational geometry and linguistics. The corresponding results are presented in Sect. 6.11 and Sect. 6.12.

5. Software

5.1. M-SVM: Multi-class Support Vector Machine

Participant: Yann Guermeur [correspondent].

We have extended the functionalities of the software version devoted to protein sequence processing (http://www.loria.fr/~guermeur/). The new version can now be used to perform different tasks in addition to secondary structure prediction, such as the identification of amphiphilic helices (see also Sect. 6.4). Furthermore, its code has been optimized in terms of `cpu` time and memory requirements.

5.2. KOALAB: KOupled Algorithmic and Learning Approach for Biological sequences

Participants: Damien Eveillard, Abdelhalim Larhlimi [correspondent], Sandrine Schermack-Peyrefitte [correspondent].

KOALAB is a software dedicated to biologists which has been developed to provide a user-friendly interface for the M-SVM (Multi-class Support Vector Machines) technology developed in the team. It is a tool based on web technology easy to use and necessitating only an Apache server to be installed. Its purpose is the search for regulatory motifs of biological interest within nucleic acids (ADN or ARN). The user can provide a collection of motifs on which the program will perform a learning process. There is no need to do any prior alignment of those motifs, a step necessary but limiting for other methods available. KOALAB also integrates the nucleic acid version of `grappe`, the motif finding algorithm developed by the ADAGE project-team. It hence offers the possibility to confront in the same graphical representation motif search results based on statistical learning with those obtained by the algorithmic methodology commonly used to date. A first study for two splicing regulatory protein targets on the HIV-1 virus genome (see Sect. 6.6) shows that this method is powerful compared to others available such as global consensus research or the software `ESEfinder` [30],

which is based on Hidden Markov Models (HMM). KOALAB 1.0 can be downloaded from our website under the GNU GPL licence. It has been presented at JOBIM'2004 conference [19], and an installation on the bioinformatics server of LORIA is being under study.

5.3. SCIL – Symbolic Constraints in Integer Linear Programming (ATIPE)

Participants: Ernst Althaus [correspondent], Alexander Bockmayr.

We designed a new software system SCIL that introduces symbolic constraints into branch-and-cut-and-price algorithms for integer linear programs [26]. Symbolic constraints are known from constraint programming and contribute significantly to the expressive power, ease of use, and efficiency of constraint programming systems. More information can be found on the SCIL-homepage http://www.mpi-sb.mpg.de/SCIL.

6. New Results

6.1. Integer programming and the phase problem in crystallography

Keywords: *Integer programming*, *crystallography*, *phase problem*.

Participants: Alexander Bockmayr, Eric Domenjoud.

The electronic density ρ in a crystal is a non-negative periodic function which may be written as a Fourier series with complex coefficients $\mathcal{F}_n = F_n e^{i\varphi_n}$. X-ray crystallography provides us with the magnitudes F_n but this information is only partial in that the phases φ_n remain unknown. In order to rebuild the ρ function, we must determine these phases by other means. This step constitutes the *phase problem* in crystallography. We have to determine a non-negative function ρ such that the magnitudes of its Fourier coefficients match the measured values. We address a discrete version of the problem where we are interested in the value of ρ only at the vertices of a grid on the unit cell of the crystal. The function ρ itself is taken of the form $\alpha\chi$, where α is a real coefficient and χ takes its values in $\{0, \cdots, K\}$.

In [10], we have shown how this problem can be modeled and solved with binary integer programming (cf. Sect. 3.3). During this year, we started to investigate another approach based on the *Patterson function* [42]. This function links directly the electron density ρ to the magnitudes F_n without involving the phases. We use then local search techniques to minimize an objective function defined as the square of the norm of the difference between the Patterson function computed from the measured values and the one computed from a candidate solution.

The possible benefits are as follows. First, we get rid of the phases which never explicitely appear in the equations. Second most handled values are naturally integers, which allows for efficient computations and updates of the objective function. In addition, at each search step, an optimal value for the coefficient α may be deduced directly. This method is still under investigation, but first results obtained on some real examples like the G-protein are very promising.

6.2. Structural risk minimization inductive principle for multi-class discriminant analysis

Keywords: *Statistical learning theory*, *model selection*, *support vector machine*.

Participants: Yannick Darcy, Yann Guermeur, Frédéric Sur.

We have continued our study of the generalization error of large margin multi-class discriminant models. Two types of contributions have been made, for the general case, and the specific case of M-SVMs. The general case has been tackled through the pathway initially proposed by Vapnik: first connect the capacity measure appearing in the confidence interval of the guaranteed risk to a generalized Vapnik-Chervonenkis (VC) dimension, then bound this VC dimension. This is the subject of [21], where we have extended our previous works on scale-sensitive Ψ-dimensions. These extensions mainly deal with generalized Sauer's

lemmas and the computation of bounds on the margin Natarajan dimension. Two independent studies are currently underway to characterize the generalization performance of M-SVMs. The aim of the first one is to extend to the multi-class case the estimates based on the leave-one-out procedure. Those estimates were initially derived by Chapelle and Vapnik. In the second one, we make use of theorems relating covering problems and the degree of compactness of operators. This work can be seen as a continuation of [8]. First results can be found in [25].

6.3. Probabilistic automata inference

Keywords: *Statistical learning theory*, *grammatical inference*, *probabilistic automata*, *stochastic languages*.

Participant: François Denis.

Multiplicity automata (or rational series) are formal objects which can model *stochastic languages*, i.e., probability distributions over words. They can be represented by a *structure* which is a finite automaton and by *continuous parameters* associated with states and transitions. Given a structure A and a sample S independently distributed according to a probability distribution P, computing parameters for A which maximize the likelihood of the observation is NP-hard, but efficient algorithms can be used in practical cases. On the other hand, inferring both structure and parameters from a sample is a widely open field of research, that is studied with Yann Esposito: he is currently achieving a PhD on this subject. We have proved that the set of stochastic languages generated from $\mathbb{Q}$-rational series is not recursively enumerable and hence, seems not suitable for grammatical inference purpose. However, we showed that the set of stochastic languages generated from $\mathbb{R}_+$ -rational series (PA) can be uniformly identified in the limit with probability one, provided that a structure which fits the sample according to $||\cdot||_\infty$ norm can be found [18]. This problem is likely to be computationally difficult for the general class. We introduce a natural subclass of $\mathbb{R}_+$ -rational series (PRA), which define a class of stochastic languages having an intrinsic characterization by means of their residual languages and for which efficient inference algorithms can be designed [33][34], see also [13].

6.4. Protein structure prediction

Keywords: *Statistical learning*, *disulphide bridges*, *kernel engineering*, *protein secondary and tertiary structure*.

Participants: Yannick Darcy, Yann Guermeur, Frédéric Sur.

Knowing the three-dimensional structure of a protein can greatly help to infer its function. Predicting this *tertiary structure* from the sequence of amino acids (or *primary structure*), remains one of the central open problems in structural biology. This is the subject of the «GENOTO3D» project that we coordinate. This year, our main efforts have been concentrated on the implementation of M-SVMs for secondary structure prediction and disulphide bridge prediction. Those efforts have mainly taken two aspects. On one hand, we have derived uniform convergence results of the empirical risk with the aim to use them as objective function in a procedure of model selection. On the other hand, we have started to study the possibility to derive kernels with good discrimination properties from pair-HMMs.

Another contribution to predictive structural biology, although not directly related to the structure of globular proteins, is a collaboration with the team of Gilbert Deléage, at IBCP Lyon. It deals with the identification of amphiphilic helices. This work provides us with the opportunity to assess the efficiency of the M-SVM dedicated to protein sequence processing [15] in a context different from its initial use, which calls for significant changes in the parameterization.

6.5. Search for non-coding RNA genes

Keywords: *non-coding RNA*, *pattern discovery*, *support vector machine*.

Participants: Emmanuel Gothié, Sandrine Schermack-Peyrefitte.

While traditional genome analysis focuses on protein-encoding sequences, there is a growing demand for tools to analyse non-coding RNA. In the context of a collaboration with the UMR 7567 MAEM, we have been specially interested in small nucleolar RNAs (snoRNA), which are involved in two types of post-transcriptional modification of the ribosomal RNA. A tool to search for snoRNAs has been developed, based on the multi-class support vector machines studied in our group. Preliminary results of this approach have been reported last year [37]. Additional experiments performed this year revealed specificity problems in a number of new testcases. To overcome these problems, we propose to develop problem-specific kernel functions. A possible starting point are marginalized kernels [40], which measure the similarity of two RNA sequences taking into account their secondary structure, which is estimated by using stochastic context-free grammars (SCFG). It is indeed very important for those sequences to consider not only their primary, but also their secondary structure, which plays a crucial role for their function.

6.6. SELEX data processing

Keywords: *SELEX*, *Statistical learning theory*, *pattern discovery*, *support vector machine*.

Participants: Stéphanie Bonne-Billaut, Damien Eveillard, Abdelhalim Larhlimi, Sandrine Schermack-Peyrefitte.

Nucleic acid-protein interactions play an important role in the cell. Recent work shows the importance of nucleic motifs in these interactions. SELEX experiments [46] can automatically characterize the potential ligands for a given target protein, starting from a random oligonucleotidic database. As shown in [11], processing SELEX data is a non-trivial task. The biological motifs generally cannot be directly identified from the experimental database. In particular, this holds for the binding sites of SR proteins, a protein family that is important in the regulation of the alternative splicing process, see Sect. 6.7.

A new method to localise a protein binding motif, based on statistical learning, has been developed in the team. We optimised a kernel method (M-SVM), dedicated to the recognition of SR motifs. Our machine was trained on experimental SELEX data. To analyse the M-SVM results biologically, the graphical interface KOALAB (see Sect. 5.2) was developed. Using data analysis in addition to the graphics interpretation, we can now predict SR binding sites in the HIV-1 genome. A complete analysis of the M-SVM results for two SR proteins, SC35 and 9G8, has been performed [19]. The study concentrated on well documented splicing regulatory sites in the HIV-1 genome, the A2, A3 and A7 acceptor sites, in order to validate the approach with a maximum of experimental data. We also compared our method to the classical global consensus approach using the `grappe` tool, which is also present in KOALAB, as well as the `ESEfinder` software [30], which is based on Hidden Markov Models (HMM). Our results show that the M-SVM gives the best results for SC35, in selectivity and specificity, compared to the other available methods. It suggests some potential sites for 9G8. These have to be tested experimentally, since only poor experimental results for this protein are currently available.

6.7. Modeling of alternative splicing regulation

Keywords: *HIV-1*, *Modeling*, *alternative splicing*, *constraint programming*, *hybrid system*.

Participants: Alexander Bockmayr, Arnaud Courtois, Damien Eveillard, Myriam Vezain.

Alternative splicing is a key process in post-transcriptional regulation, by which several kinds of mature RNA can be obtained from the same premessenger RNA. The resulting combinatorial complexity contributes to biological diversity, especially in the case of the human immunodefficiency virus HIV-1. In collaboration with the UMR 7567 MAEM in Nancy, we have developed different formal models of the alternative splicing regulation in HIV-1 [14].

Despite its importance for the HIV-1 life cycle, the consequences of alternative splicing regulation have not yet been well studied experimentally. In order to overcome this difficulty, we propose an integrative model based on a hybrid automaton with default reasoning [17][11][23]. This hybrid automaton integrates continuous models of the local regulation at the splicing sites A3 and A7 together with assumptions on the sites A4 and

A5 into a generic multi-site model. Using this approach, we may study the impact of the regulation at one activator site on the production of the HIV-1 proteins characteristic for the late phase of the disease. As an example, we analysed the impact of increasing the concentration of the splicing inhibitory protein hnRNPA1. Based on biological queries and a model checking approach, our model can be validated in a qualitative way. In addition, we may generate hypotheses for future experimental work.

The multi-site model of alternative splicing regulation has also been integrated into a model of the HIV-1 life cycle [39], see [11][23]. We are currently performing a theoretical analysis of this new model in order to study the global effect of alternative splicing regulation at the level of the HIV-1 life cycle.

6.8. Metabolic pathways analysis

Keywords: *extreme rays*, *metabolic networks*, *pathways*, *polyhedra*.

Participants: Alexander Bockmayr, Stéphanie Bonne-Billaut, Abdelhalim Larhlimi.

Studying metabolic networks at steady state involves the computation of special pathways that characterize all the possible fluxes in the network. From a mathematical point of view, those pathways correspond to the extreme rays of a polyhedral cone, which is defined by all the fluxes verifying the stoechiometric constraints of the system, together with some non-negativity constraints. Several approaches have been recently proposed to compute such extreme pathways [41]. However, due to the inherent algorithmic complexity of the problem, their number may be very large even for small networks. In [22], we propose an improved formalization of the metabolic network, which reduces the number of variables and constraints. This allows us to determine a minimal set of characteristic pathways, which we call *generic* pathways. The generic pathways form a proper subset of the extreme pathways, and their number is typically much smaller. We use the double description method [36] to compute those generic pathways. We give also an algorithm that allows us to obtain all the extreme pathways from the generic ones. This method is currently being implemented.

In collaboration with ISA Beauvais (A. Chango), we have developed in [20] several models of the one-carbon metabolism. Anomalies in this system, depending on B group vitamins, play an important role in various diseases, such as cardiovascular diseases, Alzheimer etc. Experimental studies being difficult, an in silico analysis seemed promising. We studied the dynamics of the system starting from an ordinary differential equation model of the methionine cycle [43], and by applying the power-law formalism [49]. In a second step, a steady-state analysis of the one-carbon metabolism was performed, based on the method outlined before. The generic pathways that have been found are subject to further biological study.

6.9. Constraint programming and integer programming

Keywords: *constraint programming*, *cooperative solving*, *integer programming*.

Participant: Alexander Bockmayr.

In a joint work with John N. Hooker (CMU), we present in [12] a state-of-the-art survey of constraint programming (CP), with special emphasis on its relationship to mixed integer programming (MIP). CP methods exhibit several parallels with branch-and-cut methods for MIP. Both generate a branching tree. Both use inference methods that take advantage of problem structure: cutting planes in the case of MIP, and filtering algorithms in the case of CP. A major difference, however, is that CP associates each constraint with an algorithm that operates on the solution space so as to remove infeasible solutions. This allows CP to exploit substructure in the problem in a way that MIP cannot, while MIP benefits from strong continuous relaxations that are unavailable in CP. We overview basic concepts of CP, including consistency, global constraints, constraint propagation, filtering, finite domain modeling, and search techniques. We then indicate how CP may be integrated with MIP to combine their complementary strengths.

6.10. Multiple sequence alignment by cutting planes (ATIPE)

Keywords: *cutting planes*, *integer programming*, *multiple sequence alignment*.

Participants: Ernst Althaus, Stefan Canzar.

In [1][24], we propose a cutting plane approach for the alignment of multiple sequences, which is a central problem in computational biology, considering the general case in which (arbitrary) gap costs, besides the customary alignment costs, are specified. An interesting and unusual aspect of our approach is that the three (exponentially large) classes of natural valid inequalities that we considered since the beginning of our study turn out to be both facet defining for the convex hull of the integer solutions and separable in polynomial time. Both the facet defining proofs and the separation algorithms are far from trivial. Experimental results on instances from the BAliBase library of reference alignments [45] show that our method outperforms the best tools developed so far, in that it produces alignments which are better from a biological point of view.

6.11. Approximating k-hop minimum spanning trees (ATIPE)

Keywords: *minimum spanning tree*.

Participant: Ernst Althaus.

In a paper to appear in Operations Research Letters in 2005, we consider the problem of computing minimum-cost spanning trees with depth restrictions. Specifically, we are given an n-node complete graph G, a metric cost-function c on its edges, and an integer $k \geq 1$. The goal in the *minimum-cost k-hop spanning tree* (kHMST) problem is to compute a spanning tree T in G of minimum total cost such that the longest root-leaf-path in the tree has at most k edges.

Our main result is an algorithm that computes a tree of depth at most k and total expected cost $O(\log n)$ times that of a minimum-cost k-hop spanning-tree. The result is based upon earlier work on metric space approximation due to Fakcharoenphol et al. [35], and Bartal [27][28]. In particular, we show that the kHMST problem can be solved exactly in polynomial time when the cost metric c is induced by a so called *hierarchically well-separated tree*.

6.12. Computing locally coherent discourses (ATIPE)

Keywords: *locally coherent discourse*.

Participant: Ernst Althaus.

One central problem in discourse generation and summarisation is to structure the discourse in a way that maximises *coherence*. Coherence is the property of a good human-authored text that makes it easier to read and understand than a randomly-ordered collection of sentences. Several papers in the recent literature have focused on defining *local* coherence, which evaluates the quality of sentence-to-sentence transitions. Measures of local coherence specify which *ordering* of the sentences makes for the most coherent discourse, and can be based e.g. on Centering Theory or on statistical models. While formal models of local coherence have made substantial progress over the past few years, the question of how to efficiently *compute* an ordering of the sentences in a discourse that maximises local coherence is still largely unsolved.

In [16], we present the first algorithm that computes optimal locally coherent discourses, and establishes the complexity of the discourse ordering problem. We first prove that the discourse ordering problem for local coherence measures is equivalent to the Travelling Salesman Problem (TSP). This result implies that the problem is not approximable. Despite this negative result, we show that by applying modern algorithms for TSP, the discourse ordering problem can be solved efficiently enough for practical applications. We define a branch-and-cut algorithm based on linear programming, and evaluate it on discourse ordering problems based on the GNOME corpus and the BLLIP corpus. If the local coherence measure depends only on the adjacent pairs of sentences in the discourse, we can order discourses of up to 50 sentences in under a second. If it is allowed to depend on the left-hand context of the sentence pair, computation is often still efficient, but can become expensive.

7. Other Grants and Activities

7.1. Regional projects

We participate in the «Génopole Strasbourg Alsace-Lorraine» together with the laboratory MAEM («Maturation des ARN et Enzymologie Moléculaire»), UMR 7567, in Nancy and the IGBMC in Strasbourg.

In the framework of the CPER Lorraine 2000-2006, we participate in the project «Bioinformatics and Applications to Genomics» of the PRST «Intelligence Logicielle». Our partners here are the Laboratory of Crystallography LCM3B (UMR 7036) and the MAEM (UMR 7567) at the University Henri Poincaré, Nancy 1.

7.2. National projects

Since February 2002, we have been participating in the cooperative research action ARC CPBIO «Process calculi and Biology of Molecular Networks». Our partners are the project team CONTRAINTES from INRIA Rocquencourt (F. Fages), the Genoscope (V. Schächter) and the laboratory PPS (V. Danos) in Paris.

We have regular contacts with the INRIA project teams HELIX (Rhône-Alpes), SYMBIOSE (Rennes) and COMORE (Sophia-Antipolis). In particular, we have been collaborating with Hidde de Jong (HELIX) in modeling the regulation of alternative splicing.

Since September 2003, we are coordinating a project called GENOTO3D, which is funded by the «Action Concertée Incitative» (ACI) «Masses de Données». The aim of this project is to apply machine learning approaches to the prediction of the tertiary structure of globular proteins. Our partners are the IBCP in Lyon, the LIF in Marseille, the project team SYMBIOSE from IRISA, the LIRMM in Montpellier, and the MIG laboratory of INRA in Jouy-en-Josas.

7.3. International relations

Within the French-Russian Institute Liapunov, we have a joint project with the Institute for Mathematical Problems in Biology (IMPB) of the Russian Academy of Sciences in Pushchino (V. Y. Lunin).

We have been collaborating with researchers from Carnegie-Mellon University (E. Balas, John N. Hooker), the Center of Operations Research CORE in Louvain-la-Neuve (L. Wolsey), the Max Planck Institute for Computer Science in Saarbrücken (working groups of K. Mehlhorn and F. Eisenbrand), SAP AG (T. Kasper), the University of California at Irvine (P. Baldi), IBM at Zurich (A. Elisseeff), and the Wiener laboratories in Rosario (D. Zelus).

8. Dissemination

8.1. Serving the scientific community

Alexander Bockmayr has been leading the research action «Bioinformatics» of LORIA and INRIA Lorraine, and the project «Bioinformatics and Applications to Genomics» of the PRST «Intelligence Logicielle». He has been on the Scientific Board of the ACI IMPBio «Informatics, Mathematics, and Physics in Molecular Biology» of the French Ministry of Research, and a member of the programme committees of CMSB'04, JOBIM'04, and MCO'04. He is an associate editor of INFORMS J. Computing and coordinator of «Optimization Online», http://www.optimization-online.org.

Yann Guermeur has been a member of the program committee of CAP'04.

8.2. Teaching

Ernst Althaus tought a course on large scale optimization, and is teaching datastructures and algorithms at the Universität des Saarlandes, Saarbrücken, Germany.

Alexander Bockmayr is a professor of computer science at the University Henri Poincaré, Nancy 1.

Arnaud Courtois is a teaching assistant («ATER») in computer science at the INPL.
Damien Eveillard has been teaching bioinformatics in the DESS RGTI.

8.3. Miscellaneous

Alexander Bockmayr gave an invited course *"Constraint Problems in Computational Molecular Biology"* at the University of Brasilia in Brazil.

9. Bibliography

Major publications by the team in recent years

[1] E. ALTHAUS, A. CAPRARA, H.-P. LENHOF, K. REINERT. *Multiple sequence alignment with arbitrary gap costs: Computing an optimal solution using polyhedral combinatorics.*, in "Proc. European Conference on Computational Biology", Bioinformatics, vol. 18, n° Supplement 2, October 2002, p. S4–S16.

[2] E. ALTHAUS, K. MEHLHORN. *Traveling Salesman-Based Curve Reconstruction in Polynomial Time*, in "SIAM Journal on Computing", vol. 31, n° 1, 2001, p. 27–66.

[3] E. BALAS, A. BOCKMAYR, N. PISARUK, L. WOLSEY. *On unions and dominants of polytopes*, in "Mathematical Programming, Ser. A", vol. 299, 2004, p. 223-239.

[4] A. BOCKMAYR, A. COURTOIS. *Using hybrid concurrent constraint programming to model dynamic biological systems*, in "18th International Conference on Logic Programming, ICLP'02, Copenhagen", Springer, LNCS 2401, 2002, p. 85-99.

[5] A. BOCKMAYR, V. WEISPFENNING. *Solving numerical constraints*, in "Handbook of Automated Reasoning", A. ROBINSON, A. VORONKOV (editors)., vol. 1, chap. 12, Elsevier, 2001, p. 751-842.

[6] Y. GUERMEUR, A. ELISSEEFF, D. ZELUS. *Bound on the risk for M-SVMs*, in "Statistical Learning, Theory and Applications", 2002, p. 48–52.

[7] Y. GUERMEUR, C. GEOURJON, P. GALLINARI, G. DELÉAGE. *Improved performance in protein secondary structure prediction by inhomogeneous score combination*, in "Bioinformatics", vol. 15, n° 5, 1999, p. 413–421.

[8] Y. GUERMEUR. *Combining discriminant models with new multi-class SVMs*, in "Pattern Analysis and Applications", vol. 5, n° 2, 2002, p. 168–179.

[9] Y. GUERMEUR, H. PAUGAM-MOISY. *Théorie de l'apprentissage de Vapnik et SVM, Support Vector Machines*, in "Apprentissage Automatique", M. SEBBAN, G. VENTURINI (editors)., Hermès, 1999, p. 109–138.

[10] V. Y. LUNIN, A. URZHUMTSEV, A. BOCKMAYR. *Direct phasing by binary integer programming*, in "Acta Crystallographica Section A", vol. 58, 2002, p. 283-291.

Doctoral dissertations and Habilitation theses

[11] D. EVEILLARD. *Modélisation statistique et formelle de la régulation de l'épissage alternatif*, PhD Thesis, Université Henri Poincaré, Nancy 1, May 2004.

Articles in referred journals and book chapters

[12] A. BOCKMAYR, J. N. HOOKER. *Constraint Programming*, in "Handbook of Discrete Optimization", K. AARDAL, G. NEMHAUSER, R. WEISMANTEL (editors)., Handbooks in Operations Research and Management Science, To appear, Elsevier, 2004.

[13] P. DUPONT, F. DENIS, Y. ESPOSITO. *Links between Probabilistic Automata and Hidden Markov Models: probability distributions, learning models and induction algorithms*, in "Pattern Recognition Journal", to appear, 2004.

[14] D. EVEILLARD, D. ROPERS, H. DE JONG, C. BRANLANT, A. BOCKMAYR. *A multi-scale constraint programming model of alternative splicing regulation*, in "Theoretical Computer Science", vol. 325, n° 1, 2004, p. 3-24.

[15] Y. GUERMEUR, A. LIFCHITZ, R. VERT. *A kernel for protein secondary structure prediction*, in "Kernel Methods in Computational Biology", B. SCHÖLKOPF, K. TSUDA, J.-P. VERT (editors)., MIT Press, 2004, p. 193–206.

Publications in Conferences and Workshops

[16] E. ALTHAUS, N. KARAMANIS, A. KOLLER. *Computing Locally Coherent Discourses*, in "Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL 2004), Barcelona, Spain", 2004.

[17] A. BOCKMAYR, A. COURTOIS, D. EVEILLARD, M. VEZAIN. *Building and Analysing an Integrative Model of HIV-1 RNA Alternative Splicing*, in "Computational Methods in Systems Biology, CMSB'04, Paris", To appear, Springer, LNCS, 2004.

[18] F. DENIS, Y. ESPOSITO. *Learning classes of Probabilistic Automata*, in "COLT 2004", LNAI, n° 3120, 2004, p. 124-139.

[19] D. EVEILLARD, A. LARHLIMI, D. ROPERS, S. BILLAUT, S. PEYREFITTE. *KOALAB: A new method for regulatory motif search*, in "5èmes Journées Ouvertes Biologie Informatique Mathématiques, JOBIM 2004, Montreal", 10 pages, 2004.

Internal Reports

[20] S. BONNE. *Modélisation du métabolisme des monocarbones et du cycle de la méthionine*, Master thesis, Univ. Bordeaux, Sep 2004.

[21] Y. GUERMEUR. *Large margin multi-category discriminant models and scale-sensitive Ψ-dimensions*, Research Report RR-5314, INRIA, September 2004, http://www.inria.fr/rrrt/rr-5314.html.

[22] A. LARHLIMI. *Analyse de voies métaboliques en programmation par contraintes*, Stage de DEA, Université Henri Poincaré, Nancy 1, Jun 2004.

[23] M. VEZAIN. *Modélisation de la régulation de l'épissage alternatif de HIV-1*, Rapport de Stage, DESS EGOISt, Rouen, Université Rouen, June 2004.

Miscellaneous

[24] E. ALTHAUS, A. CAPRARA, H.-P. LENHOF, K. REINERT. *Aligning Multiple Sequences by Cutting Planes*, Submitted, 2004.

[25] Y. GUERMEUR, M. MAUMY, F. SUR. *Model Selection for Multi-class SVMs*, Submitted to ASMDA'05, 2004.

Bibliography in notes

[26] E. ALTHAUS, A. BOCKMAYR, M. ELF, T. KASPER, M. JÜNGER, K. MEHLHORN. *SCIL - Symbolic Constraints in Integer Linear Programming*, in "10th European Symposium on Algorithms, ESA'02, Rome", Springer, LNCS 2461, 2002, p. 75-87.

[27] Y. BARTAL. *Probabilistic approximation of metric spaces and its algorithmic applications*, in "Proceedings IEEE Symposium on Foundations of Computer Science", 1996, p. 184–193.

[28] Y. BARTAL. *On approximating arbitrary metrics by tree metrics*, in "Proceedings 30th Annual ACM Symposium on Theory of Computing", 1998, p. 161–168.

[29] C. BURGES. *A tutorial on support vector machines for pattern recognition*, in "Data Mining and Knowledge Discovery", vol. 2, n° 2, June 1998, p. 121–167.

[30] L. CARTEGNI, J. WANG, Z. ZHU, M. Q. ZHANG, A. R. KRAINER. *ESEfinder: a web resource to identify exonic splicing enhancers*, in "Nucleic Acid Research", vol. 31, n° 13, 2003, p. 3568-3571.

[31] W. J. COOK, W. H. CUNNINGHAM, W. R. PULLEYBLANK, A. SCHRIJVER. *Combinatorial Optimization*, Wiley, 1998.

[32] C. CORTES, V. VAPNIK. *Support-Vector Networks*, in "Machine Learning", vol. 20, 1995, p. 273–297.

[33] F. DENIS, Y. ESPOSITO. *Residual languages and probabilistic automata*, in "Automata, Languages and Programming, 30th International Colloquium, ICALP 2003", Springer, LNCS 2719, 2003, p. 452-463.

[34] Y. ESPOSITO, A. LEMAY, F. DENIS, P. DUPONT. *Learning probabilistic residual finite state automata*, in "Grammatical Inference: Algorithms and Applications, 6th International Colloquium, ICGI 2002", Springer, LNCS 2484, 2002.

[35] J. FAKCHAROENPHOL, S. RAO, K. TALWAR. *A tight bound on approximating arbitrary metrics by tree metrics*, in "Proceedings ACM Symposium on Theory of Computing", 2003, p. 448–455.

[36] K. FUKUDA, A. PRODON. *Double Description Method Revisited.*, in "Combinatorics and Computer Science", Springer, LNCS 1120, 1995, p. 91-111.

[37] E. GOTHIÉ, Y. GUERMEUR, S. MULLER, C. BRANLANT, A. BOCKMAYR. *Recherche des gènes de petits ARN non codants*, Research Report RR-5057, INRIA, December 2003.

[38] V. GUPTA, R. JAGADEESAN, V. SARASWAT. *Computing with Continuous Change*, in "Science of computer programming", vol. 30, n° 1-2, 1998, p. 3-49.

[39] B. J. HAMMOND. *Quantitative Study of the Control of HIV-1 Gene Expression*, in "J. Theor. Biol", vol. 163, 1993, p. 199–221.

[40] T. KIN, K. TSUDA, K. ASAI. *Marginalized kernels for RNA sequence data analysis*, in "Genome Informatics 2002", Universal Academic Press, 2002, p. 112-122.

[41] J. A. PAPIN, J. STELLING, N. D. PRICE, S. KLAMT, S. SCHUSTER, B. O. PALSSON. *Comparison of Network-Based Pathway Analysis Methods*, in "Trends in Biotechnology", vol. 22, n° 8, 2004, p. 400-405.

[42] A. PATTERSON. *A Fourier Series Method for the Determination of the Components of Interatomic Distances in Crystals*, in "Phys. Rev.", vol. 46, 1934, p. 372–376.

[43] M. C. REED, H. F. NIJHOUT, R. SPARKS, C. M. ULRICH. *A mathematical model of the methionine cycle*, in "Journal of Theoretical Biology", vol. 226, 2004, p. 33-43.

[44] V. A. SARASWAT. *Concurrent constraint programming*, ACM Doctoral Dissertation Awards, MIT Press, 1993.

[45] J. THOMPSON, F. PLEWNIAK, O. POCH. *BAliBASE: a benchmark alignment database for the evaluation of multiple alignment programs*, in "Bioinformatics", vol. 15, n° 1, 1999, p. 87-88.

[46] C. TUERK, L. GOLD. *Systematic evolution of ligands by exponential enrichment: RNA ligands to bacteriophage T4 DNA polymerase*, in "Science", vol. 249, 1990, p. 505-510.

[47] V. VAPNIK. *Estimation of Dependences Based on Empirical Data.*, Springer-Verlag, N.Y., 1982.

[48] V. VAPNIK. *Statistical learning theory*, John Wiley & Sons, Inc., N.Y., 1998.

[49] E. O. VOIT. *Metabolic modeling: a tool of drug discovery in the post-genomic area*, in "Drug Discovery Today", vol. 7, n° 11, 2002, p. 621-628.

[50] P. VAN HENTENRYCK, V. SARASWAT. *Strategic directions in constraint programming*, in "ACM Computing Surveys", vol. 28, n° 4, 1996, p. 701 – 726.