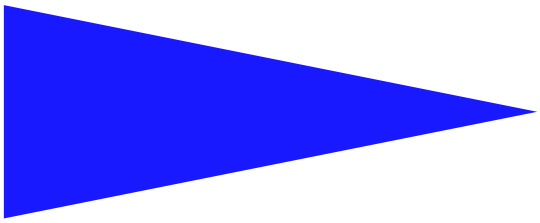
I R I S A
INSTITUT DE RECHERCHE EN INFORMATIQUE ET SYSTÈMES ALÉATOIRES

PUBLICATION
INTERNE
N° 1735



A SIMILAR FRAGMENTS MERGING APPROACH TO LEARN AUTOMATA ON PROTEINS

FRANÇOIS COSTE AND GOULVEN KERBELLEC

ISSN 1166-8687

IRISA
CAMPUS UNIVERSITAIRE DE BEAULIEU - 35042 RENNES CEDEX - FRANCE

IRISA

INSTITUT DE RECHERCHE EN INFORMATIQUE ET SYSTÈMES ALÉATOIRES
Campus de Beaulieu – 35042 Rennes Cedex – France
Tél. : (33) 02 99 84 71 00 – Fax : (33) 02 99 84 71 71
http://www.irisa.fr

A Similar Fragments Merging Approach to Learn Automata on Proteins

François Coste * and Goulven Kerbellec **

Systèmes biologiques
Projet Symbiose

Publication interne n ° 1735 — Juillet 2005 — 18 pages

Abstract: We propose here to learn automata for the characterization of proteins families to overcome the limitations of the position-specific characterizations classically used in Pattern Discovery. We introduce a new heuristic approach learning non-deterministic automata based on selection and ordering of significantly similar fragments to be merged and on physico-chemical properties identification. Quality of the characterization of the major intrinsic protein (MIP) family is assessed by leave-one-out cross-validation for a large range of models specificity.

Key-words: grammatical inference, automata, proteins

(Résumé : tsvp)

Goulven Kerbellec is supported by a PhD research grant from Région Bretagne.

* francois.coste@irisa.fr
** goulven.kerbellec@irisa.fr

Centre National de la Recherche Scientifique (UMR 6074) Université de Rennes 1 – Insa de Rennes
Institut National de Recherche en Informatique et en Automatique – unité de recherche de Rennes

Apprentissage d'automates par une approche de fusion de fragments significativement similaires et expérimentations sur des protéines.

Résumé : Nous proposons d'apprendre des automates caractérisant des familles de protéines pour dépasser les limites des méthodes usuelles de Découverte de Motifs qui sont actuellement restreintes à des caractérisations par positions. Nous introduisons ainsi une nouvelle approche heuristique permettant d'apprendre des automates non déterministes, basée sur la sélection, le tri et la fusion de fragments significativement similaires, ainsi que sur l'identification de propriétés physico-chimiques. La qualité de caractérisation de la famille de protéines MIP (major intrinsic protein) est attestée par validation croisée de type *leave-one-out* pour différents niveaux de spécificité des modèles.

Mots clés : inférence grammaticale, automates, protéines

1 Introduction

Proteins are essential to the structure and function of all living cells and viruses. They are amino acid chains that fold into three-dimensional structures. Most of the times, only the amino acid chain – a sequence over 20 letters each representing one amino acid – is known. Determination of the structure or the function of proteins from their sequences is one of the major challenges in molecular biology. This determination can be done by experiments but needs more and more to be assisted by *in silico* methods to face the rapidly growing amount of available sequences in the databases, produced in particular by DNA sequencing projects. One of the most successful approaches is to define signatures of known *families* of biologically related proteins (typically at the functional or structural level). A representative example of this approach is the well-known Prosite database [8], gathering signatures for a large number of protein families. In Prosite, signatures are sub-regular expressions defined essentially by experts. Automatic discovery of such motifs (Pattern Discovery) is a dynamic research field [19, 2] directed towards detection of conserved sequences. Conservation of sequences is generally important for the function of proteins and/or for the maintenance of their three-dimensional structure. Various types of patterns have been used to describe the set of conserved sequences in learning algorithms [1]. Available methods range from the identification of single short sequences to subclasses of regular expressions, each exact pattern having its probabilistic counterpart. Among the state-of-the-art algorithms learning expressive patterns, Pratt [9] (chosen to be the default pattern discovery tool proposed on the Prosite web site), Teiresias[18] or Splash[3] have been successfully designed to generate Prosite-like exact patterns while, concerning stochastic models, the corresponding state of the art would be training profile hidden Markov models (which are a particular kind of left-right hidden Markov models focusing on so-called "match" positions and allowing to handle deletions or insertions of symbols) as in the commonly used tools such as HMMER [6] and SAM [11]. An important feature of these approaches is that they are limited to *position-specific* characterizations: as a matter of fact, neither relations between positions – for instance, if we consider the disulfide bond between cysteines, the fact that when a cysteine amino acid is present at position i there should be necessarily another cysteine at position j – nor alternative paths (disjunction over more than one position) can be represented, whereas it could be done in truly regular expressions or automata.
We address in this paper the task of learning *automata* for the characterization of proteins families to overcome the current position-specific limitation of Pattern Discovery. Grammatical inference considers the problem of learning grammars from theoretical and algorithmic points of view. Since it is the least expressive class of grammars in the Chomsky hierarchy, inference of regular grammars have been widely studied, notably by using finite state automata representation and state merging techniques. The more representative state merging algorithm is certainly RPNI [17, 12], which has been shown to have identification properties and good performances on artificial data. The number of needed data may be reduced with the help of the EDSM heuristic which has won the Abbadingo competition, still on artificial data.

In contrast, while the application to genomic sequences seems to be a promising field for Grammatical Inference, not much work has been published on this matter (if we restrict ourselves to methods discovering actually a grammar and not just training its weight parameters, which would for instance exclude the work of Sakakibara on stochastic context free grammars for the prediction of RNA structure [20] but would include the application of Sequitur [16] to infer a hierarchical structure on DNA sequences without generalization capabilities). Concerning the application of such methods for the characterization of proteins, we are only aware of the early work of Yokomori [22] on learning locally testable languages, a subclass of automata which may be linked to *n-grams* and to persistent splicing systems, for the identification of protein α-chain regions.
Our main contribution in this article is the proposition of a new heuristic approach in the state-merging framework, following the ideas of [5], allowing a successful inference of automata for the characterization of proteins. The approach, sketched in Algorithm 1, consists of two main stages: first a *characterization* stage, introduced in section 2, detects and orders similar protein fragment pairs, then a *generalization* stage, described in section 3, merges the candidate fragment pairs to identify globally conserved areas and physico-chemical properties. We present a first validation of our approach on a real task of protein characterization in section 4.

Algorithm 1 Significantly Similar Fragment Pairs Merging Approach

procedure SFP_MERGING (S: set of sequences, q: quorum, $\mathcal{G}$: set of amino acid groups, λ_G,λ_Σ: likelihood tests thresholds)

▷ Characterization stage

$L \leftarrow$ LIST_OF_SFP(S) ▷ section 2.1
for each *sfp* $\in L$ **do**
  sfp.score $\leftarrow$ REPRESENTATIVITY_EVALUATION(*sfp*, S) ▷ section 2.2
L.SORT_BY_SCORE()

▷ Generalization stage

$A \leftarrow$ MAXIMUM_CANONICAL_AUTOMATA(S) ▷ section 3.1
for each *sfp* $\in L$ **do**
  A.MERGE_IF_ADMISSIBLE(*sfp*)
A.GAP_GENERALIZATION(q) ▷ section 3.2
A.INFORMATIVE_POSITIONS($\mathcal{G}, \lambda_G, \lambda_\Sigma$) ▷ section 3.3
return A

2 Characterization

2.1 Significantly Similar Fragment Pairs

Each amino acid has multifaceted properties that are responsible for the specificity and diversity of protein structure and function (see for instance the principal properties of amino

acids shown in Fig. 3). According to their shared properties, some amino acids may be more easily substituted each another under the pressure of natural selection. Substitution matrices, like PAM and BLOSUM, provide the probability (or odds) of changing one amino acid into another. Such matrices allow to refine the notion of conservation by introducing similarity measures between protein sequences. We propose in this section to use these matrices to identify similar protein fragment pairs in order to identify interesting conserved areas.

In this study, we use the term *protein fragment* to designate a contiguous subsequence of a protein. We consider protein fragment pairs such that both fragments have the same length. The similarity of such pair is the sum of the individual similarity values of the facing amino acids. Difficulty in comparing the similarity of two different length fragment pairs is a well known problem. To overcome this problem, we use sets of *significantly similar fragment pairs* (SFP) provided by DIALIGN2 [15]. DIALIGN2 is a multiple alignment tool whose first step consists of finding all fragment pairs such that their similarity is significantly larger than expected on random sequences (as measured by a weight function $w(s, l)$ related to the probability of finding any fragment pair of length l with a score at least as large as s taking into account the lengths of the protein sequences). In DIALIGN2, these SFP are then combined to make a multiple alignment optimizing the global sum of weights under consistency constraints. In our approach, this set of SFP is considered as a first selection of fragments such that merging them is potentially interesting. The selection of these fragments is based only on sequence-to-sequence comparison. We will see in the next section how to rank these fragments according to their representativeness of the whole protein family.

2.2 Ordering Similar Fragment Pairs

To order the SFP with respect to their representativeness of the whole family, we may try to estimate their support in other sequences of the family, i.e. to count for each SFP the number of sequences containing a fragment sufficiently similar to it. Several criteria can be chosen to decide if a fragment is similar to two other ones. Let us note that transitivity does not hold for similarity. We use the triangular inequality since it is simple, robust and parameter-free. To simplify the expressions, we use $w(f_1, f_2)$ instead of $w(s, l)$ to designate the DIALIGN2 weight of a fragment pair $p = \langle f_1, f_2 \rangle$ having similarity score s and length l. A SFP (f_1, f_2) is said *supported* by a fragment f if: $w(f, f_1) + w(f, f_2) \geq w(f_1, f_2)$. A SFP is said to be supported by a sequence if it is supported by at least one fragment of the sequence. Let p be a SFP. We define the *support* of p in a set of sequences S as the number of sequences supporting p in S, denoted by $\sigma_S(p)$. Hereafter, we denote $Support_S(p)$ as the set of sequences included in S supporting p.

When S is the family of proteins, $\sigma_S(p)$ may be used to evaluate how each p is representative of the family. In the following, ordering the SFP according to the support σ_S will be refered to as the *support heuristic*. More elaborate indices based on the support may also be constructed. In particular, if a set of proteins known not to belong to the family is available (we will denote this set by N), support may be used to detect discriminative fragments in the family of proteins with respect to the other set of proteins. To achieve this goal, we

compute an *implication index* for each SFP p based on [14], denoted by $\iota(p)$:

$$\iota(p) = \frac{-P(Support_N(p)) + P(Support_S(p)) \times P(N)}{\sqrt{P(Support_S(p)) \times |N|)}}$$

where $|X|$ denotes the cardinality of a set X and $P(X)$ denotes its proportion with respect to S and N: $P(X) = \frac{|X|}{|S|+|N|}$. This formula evaluates how the support of the SFP in the family implies its proportion to be supported in the family and in the other set of sequences. The *implication heuristic* will refer to ordering the SFP according to ι.
The implication index relies on a set of proteins not belonging to the family to give priority to *discriminative characteristic* SFP. This set can be a set of counter-examples, for instance, BLAST hits known to be outside the family. The set can also contain sequences from another family in a classification oriented perspective. Let us notice that since discrimination relies on the implication index instead of the classical compatibility criteria used in grammatical inference (which forbid strictly to recognize one counter-example) the labeling of some counter-examples can be erroneous and some labeling noise may be handled. More generally, the extra set of sequences is not really needed to be labeled. The implication index can be used in a semi-supervised manner without the need to identify close counter-examples.

3 Generalization

We present now the second stage of the approach. The section 3.1 introduces the first generalization step that consists in merging the SFP. It allows to detect conserved fragments and to discard parts of the model without enough evidence in the family as presented in section 3.2. A refinement of the model based on the identification of physico-chemical properties is then proposed in section 3.3.

3.1 Merging Fragment Pairs

The first generalization step applies the classical state-merging scheme popularized by RPNI [17] and EDSM [13] to SFP. We consider the more general case allowing to learn non-deterministic automata. Following the definitions of [4], to which we refer the reader for details, the general sketch of this kind of algorithm is to first construct an automaton, named *maximal canonical automaton* (MCA), representing exactly the training set of sequences and, then, to generalize the recognized language by *merging* (unifying) some of its states. State merging algorithms can be distinguished by their choice of states to merge. We propose here to merge iteratively (see Fig. 1) the states corresponding to the SFP identified in the characterization stage, beginning by SFP with higher representativeness and designating respectively ordering with respect to the dialign weight, the support in the positive training set, and the implication index of the SFP.
To define fragment merging, let us remark that since MCA represents exactly the training set, one can define a one-to-one function from the amino acids positions to the corresponding

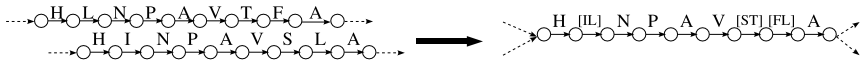


Figure 1: Generalization by merging a pair of fragment in an automaton.

transitions in MCA. The facing positions of a SFP determinate thus a set of pairs of transitions. We define the SFP merging procedure as merging, for each corresponding pair of transition, the two target states together and the two source states together. The SFP ordering is taken into account by introducing a preservation constraint over the previously merged fragments. Namely, after each SFP merging, a constraint stating that the resulting states can not be merged together is set. Further SFP mergings that would violate such constraint are discarded. Let us note that the preservation constraints are used only during the step described in this section and may be forgotten after having considered all the candidate merges.

3.2 Representative Fragments

Merging the SFP allows to identify hot spots: sets of contiguous positions where lots of fragments have been merged. Besides, some positions may be involved in none of SFP merges. These latter localizations are clearly not representative of the family. We propose to treat them as "gaps". We introduce classically a quorum parameter. If a state is used by less sequences than specified by the quorum, it is merged with its neighbors. This step allows to keep only the characteristic regions and is an important generalization step for long proteins. Several variations around this scheme could be implemented. Statistical information like the length or the amino acid composition of the gap could also be considered and added to the model. The version presented here is the simple one used in the experiments.

3.3 Identification of Physico-chemical Properties

The general substitution matrices used so far for the localization of conserved fragments are estimated from large sets of close proteins and thus reflects only average similarity (over various contexts involving different physico-chemical properties of the amino acids). We propose here to use these localizations as contexts to recover the important physico-chemical properties of the amino acids with respect to the function or the structure of the family.
The approach takes as input a set $\mathcal{G}$ of eventually overlapping substitution groups representing important physico-chemical properties (typically the groups proposed by Taylor [21], see Fig. 3). The sketch of a naive identification would be to test for each set of amino acids P aligned by the approach if it is equal to one of the given groups. This approach may be applied only to small groups or else it will require a large amount of training sequences to identify all the important groups (consider for instance the probability of aligning all the 13 hydrophobic amino acids in a limited set of correlated proteins).

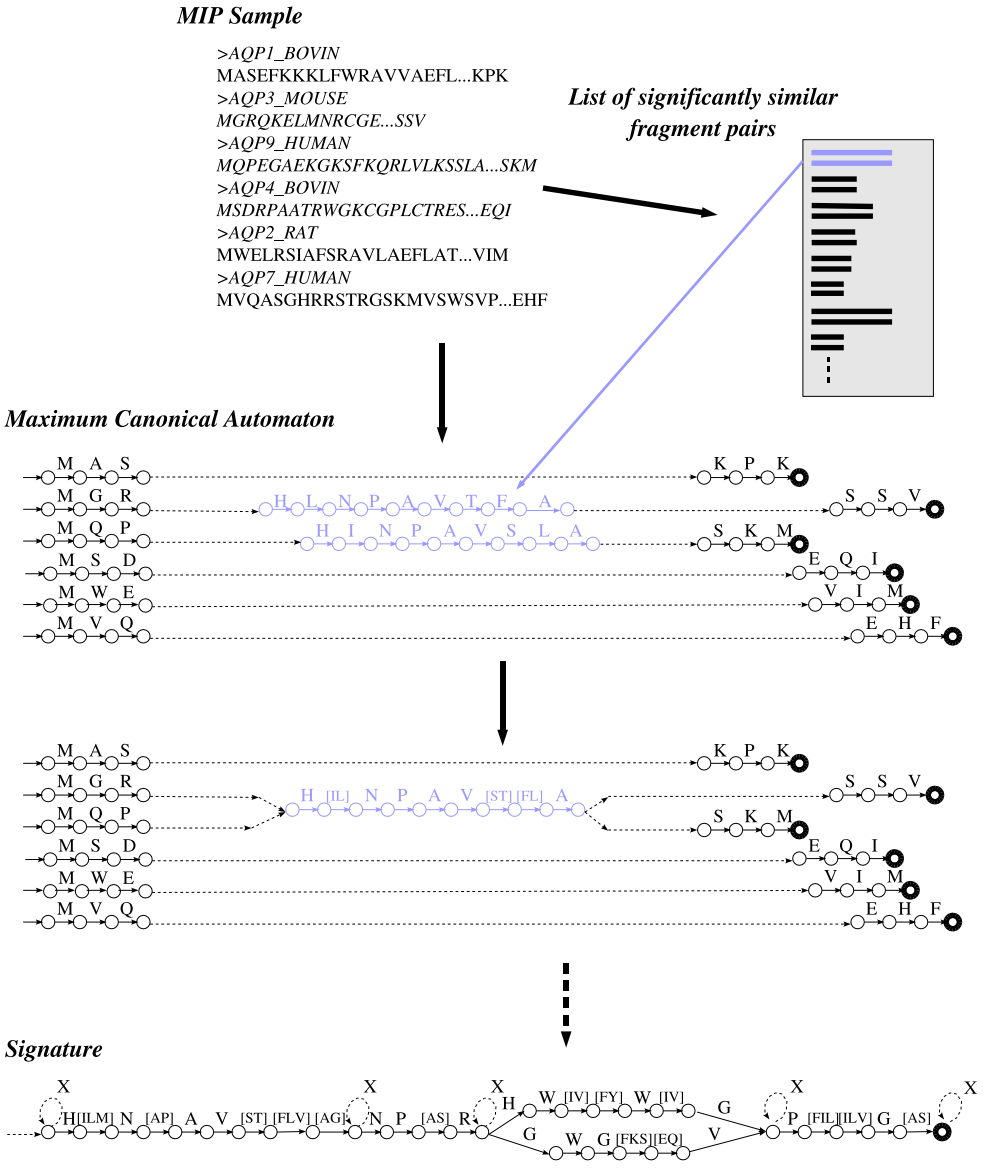


Figure 2: Main scheme of SFP merging approach on 6 MIP.

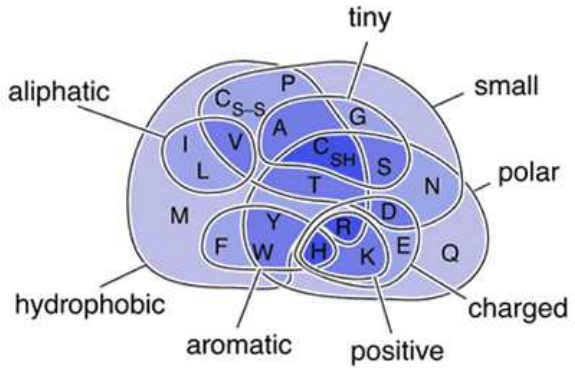


Figure 3: Venn diagram proposed by Taylor in [21] showing the relationship of the 20 amino acids to a selection of physico-chemical properties thought to be important in the determination of protein structure and function.

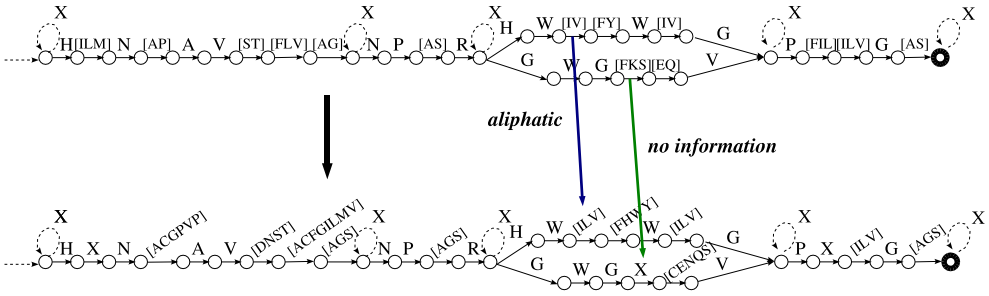


Figure 4: Expansion to the physico-chemical properties.

We propose to use a statistical test to decide if the multi-set P has been generated according to a physico-chemical group G or not (see Fig. 4).
Given two states q_1, q_2 of the automata, let P be the set of all the amino acids allowing to reach q_2 from q_1 and let n be the total number of sequences using these transitions. We decide to replace the current set of amino acids P by the smallest physico-chemical group G including P based on the result of a likelihood ratio test. To compute this ratio, we use the background probability p_a of each amino acid a and we estimate the probability $p_{a|G}$ of this amino acid given that it belongs to G by $p_{a|G} = c_G p_a$ where c_G is a proportional redistribution factor of the missing amino acids: $c_G = \frac{1}{\sum_{a \in G} p_a}$.
In that setting, we can compare the likelihood L_G of G when n amino acids are drawn from G to its likelihood when the amino acids are drawn from P by the ratio:

$$LR_{G/P} = \frac{L_G}{L_P} = \left(\frac{\sum_{a \in P} p_a}{\sum_{a \in G} p_a} \right)^n$$

Given a threshold λ_G, we test the expansion of P to G and reject it when $LR_{G/P} < \lambda_G$.
If the expansion to G is rejected, there is no evidence of a physico-chemical property in the set of amino acids P. In such cases, one may wonder whether the amino acids in P have been generated randomly and then replace the set by the whole alphabet Σ, or whether the composition of the set is important and should be kept as it is. By replacing G by Σ and introducing the threshold λ_Σ, we test in a similar way the expansion of P to Σ by rejecting it when $LR_{\Sigma/P} = \frac{L_G}{L_P} = (\sum_{a \in S} p_a)^n < \lambda_\Sigma$.
These tests introduce two parameters λ_G and λ_Σ allowing to tune the risk when expanding P to G or else to Σ. When both are set to 1, no expansion will be done and the characterization will rely on the quality of the training set. In contrast, setting λ_G to 0 will expand all the sets of amino acids to the closer group including them: the quality of the result will then strongly depend on the choice of the sets of physico-chemical groups. Setting λ_G to 1 and λ_Σ to 0 allows to keep only the sets of amino acids equals to one of the physico-chemical groups of amino acids: this is the kind of behavior that we will favor when choosing the real thresholds, in particular when sequences are closely related and have a large percentage of identity.

4 Experiments

We evaluated our approach on the major intrinsic protein (MIP) family [10]. The MIP family has been constituted according to functional and structural properties. Proteins of this family are transmembrane channels, well-known to be important for water, alcohol and small molecules transport accross cell membranes thanks to P. Agre (Nobel Prize in Chemistry "for the discovery of water channels", 2003). UNIPROT, a biological protein sequence database, contains 911 proteins annotated as being members of the MIP family. Of these 911, 159 protein sequences (denoted hereafter by the set T) are present in SWISSPROT which is the reliable annotated public reference database used by Prosite. Of this set, a

biology expert has identified only 79 sequences with a real biological experiment-based annotation (a lot of proteins being annotated "by similarity"). By filtering out the sequences with more than 90% of identity, this set was then reduced to 44 sequences (set M). Of this set, the expert has identified 24 water-specific sequences (set W+) and 16 glycerol or small molecule facilitator sequences (set W-). Let us notice the difficulty of the discrimination task between these MIP, some sequences of W+ being closer to some sequences of W- than to the other sequences of W+. We have established also a control set composed of sequences close to MIP sequences (first Blast hits) and identified by the expert as being outside the family (set C).
Two kinds of experiments were performed. First, our fragment merging scheme was validated by comparison with Pratt[9] and Teiresias[18] methods and Prosite hand-made pattern[1]. For this purpose, we restricted Protomata-L to return only the first common fragment shared by all sequences. Secondly, we considered the more difficult task of functional discrimination between the MIPs known to be water-specific and the MIPs known to be glycerol or small molecules facilitators. This task is motivated by a better understanding of the transport of these molecules. We used it to study the quality of the characterization, as attested by leave-one-out prediction performances at different specificity levels, on closely related sets of sequences.
All the experiments were performed with an implementation of our approach named Protomata-L using DIALIGN2 with the following options : `-nta -thr 5 -afc`. The group expansion of Protomata-L has been done with the sets of physico-chemical properties proposed in Fig. 5 of [21] except the "unions" group[2], and $\lambda_G = 10^{-7}$, $\lambda_\Sigma = 10^{-19}$. Even with our unoptimized code, the execution never exceeded 10 minutes on a 3GHz desktop station.

4.0.1 First Common Fragment.

For this first set of experiments, in order to compare our fragment merging approach with Pratt[9] and Teiresias[18] methods and Prosite hand-made pattern, we restricted Protomata-L to return only the first common fragment shared by all sequences, using support heuristic. Pratt and Teiresias were used with their default parameters, except the parameter W (maximum length) of Teiresias that was set to 50 to allow longer pattern to be discovered. The patterns were learned from the set M and tested on the set T.

A scan of the Prosite's pattern on SWISSPROT database returns false positive as well as false negative sequences with respect to T (while T was used to define Prosite's pattern). Table 1 summarizes the results of such scans for the three patterns. The recall of our approach is close to Prosite's pattern recall while our precision remains at 100%. Let us remark that in our false positives, one was not a full sequence and 16 were annotated as MIP by similarity. When comparing our approach to Pratt and Teiresias pattern discovery tools, the comparison is clearly in favor of Protomata-L with respect to both the precision and the recall.

[1] Preliminary tests, not reported here, showed that RPNI and EDSM were not able to propose pertinent automata from this kind of data

[2] Identifying two alternative properties is not likely to be interesting here.

Table 1: Comparison of 4 MIP signature patterns.

Method	Precision	Recall	F-mes.	Pattern
Prosite (reference)	95%	91%	0.93	[HNQA]DNP[STA][LIVMF][ST][LIVMF][GSTAFY]
Pratt	90%	78%	0.83	GX(2)[FILMV]NP[AS]X[DST][FIL][AGP]
Teiresias	23%	89%	0.37	[ILMV]X(10)[ST]X(3)[ILMV]NX[AG]X(3)[AG]
Protomata-L	100%	87%	0.93	[ACGSTV]X[ACFGILMV]N[ACGPV][AGS][ACFG ILMV][DNST][ACFGILMV][ACGSTV]X[ACFGHI KLMTVWY]X(12)[FMY]X[ACFGHIKLMTVWY]X Q[ACFGHIKLMTVWY][ACFGILMV][AGS][AGS]

Let us note that the three patterns focus on the same site, the so-called NPA box. Our pattern is much longer than the other patterns. It contains also some common positions linked to amino acids placed on an alpha helix and turned to the channel which are likely to be important for the structure and function of this family. In the next paragraph we focus on a more precise characterization (combining several fragments) of a subclass of the MIP family with the help of counter-examples.

4.0.2 Water-Specific MIP subfamily.

In this second set of experimentations, we focus on the characterization of the water-specific MIP subfamily set W+, using the set W- as counter-example. This discrimination task is motivated by a better understanding of the transport of these molecules. We used it to study the quality of the characterization on closely related sets of sequences at increasing specificity levels. Due to the small number of available sequences, a leave-one-out cross-validation scheme was used to evaluate our approach. For each couple of positive and negative sequences $(w+, w-)$, the training was achieved using the remaining sequences of W+ and W-. For each leave-one-out datasets, several automata – ranging from short automata (like in the previous paragraph) to larger automata characterizing almost all the length of the MIP topology – were obtained by using an increasing number of SFP. Each automaton was then evaluated according to the distance for acceptation of the positive sequence left out $w+$, the negative sequence left out $w-$, and also of the closest sequence c in the control set C. The *distance for acceptation* refers here to the minimal cost of amino acid substitutions needed in the sequence for its acceptation by the automaton (the cost of each amino acid substitution being given by the classical substitution matrix Blosum62 [7]).
Fig. 5 presents the results of all these experiments when using the implication heuristic and a quorum of 100%. On the size axis, we highlighted 4 attraction points which are related to the progressive emergence of common sub-patterns, the first one corresponding to the first

common fragment. The separation of the different sets of sequences is manifest[3] and grows along the automata size axis until an inflexion point near 100 states. Behind this inflexion point, the merged SFP do not contribute anymore to the discrimination but only to a more precise characterization of the family without showing over-generalization evidence.
Fig. 6, illustrates at the inflexion point the importance of the heuristic choice. Compared to the other heuristics, the "implication" heuristic allowed to clearly separate W+ and W-. We can also notice that, the set of control sequences was moved away from the MIPs sequences, even if they were not taken into account during the learning process.
Table 2 sums up the results of the automata at the attraction points for the classification task between W+ and W-, with strict acceptation and with a distance threshold acceptation. In the latter case, the closest counter-example distance from the automata was taken as the threshold distance for acceptation. The approach was then able to raise 100% of precision and 100% of recall for automata sizes of 40 or even 100 states.

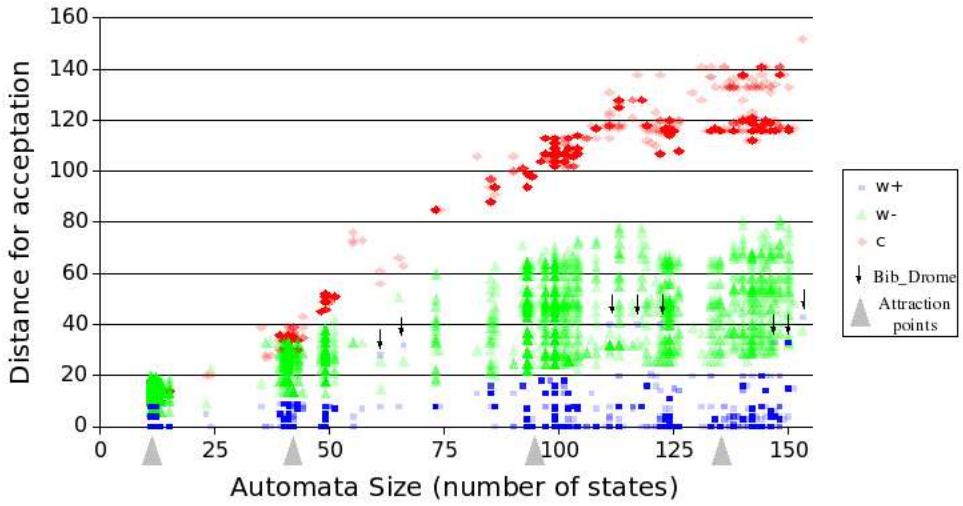


Figure 5: Distance of each test sequence for acceptation by automata when using the implication heuristic. The set to characterize was the Water-Specific W+ set with W- as counter-example set. Set C was a non-MIP control set, only the smallest distance was reported on the plot for C.

[3]Only one sequence from set W+, which is called Bib_Drome is sometimes plotted at the level of the usual distance of W- sequences. Bib_Drome is known to be divergent from the other MIPs and it is not surprising if more substitutions were needed to parse this sequence when no other representative of this family were available in the training set. Nevertheless, this distance needed to parse this sequence was always smaller than the one needed to parse sequences outside the family (from W- or C).

Table 2: Performance on classification task (W+ vs W-).

	Strict			Threshold		
Automata Size	Precision	Recall	F-mes.	Precision	Recall	F-mes.
10	100%	92%	0.96	100%	96%	0.98
40	100%	71%	0.83	100%	100%	1.00
100	100%	54%	0.70	100%	100%	1.00
130	100%	42%	0.59	100%	96%	0.98

5 Conclusion

This study shows – even if it has to be confirmed on other sets of sequences – that good automata can be learned successfully on proteins. The proposed heuristic approach can be applied to the characterization of a family of proteins from positive examples only. It is also able to benefit from available counter-examples to produce more subtle models performing well in the discrimination of closely related family of sequences. Depending on the application, level of the precision in the learned models can be chosen, ranging from short characteristic models (for classification tasks) to more detailed and explanatory models (for modeling the family of sequences).

As proved by performance in leave-one-out cross-validation, the more specific models have still good prediction accuracy when allowing a small distance for acceptation to compensate the limited number of available examples. An alternative way to handle unpredictable family variation would be to use the learned automata as the underlying structure of probabilistic automata, or hidden Markov models, and estimate their stochastic parameters by the classical well-studied training methods. The advantage of our approach is that these variations are treated outside the model by measuring the distance to it, allowing the models to focus only on an explicit characterization of the important properties of the training sequences. We think that we could even improve the prediction accuracy by using distances taking into account the weights of the amino acids at each position with respect to the training sequences, but this has still to be implemented and tested.

Compared to classical protein Pattern discovery algorithms, our approach introduces several new ideas. Globally, we think that, besides the ability to learn more expressive class of model learned, the fundamental difference of Protomata-L with these Pattern Discovery approaches consists in the introduction of the similarity of fragments (which reflects the conservation of the site and probably the conservation of some structural aspects of it) as an important criterion for the characterization. This allows to consider the characterization of positions according to their context. Protomata-L introduces also the possibility to produce discriminative characterization of a set of sequences with respect to another one, which can also be used for learning with counter-examples or with unlabeled examples.

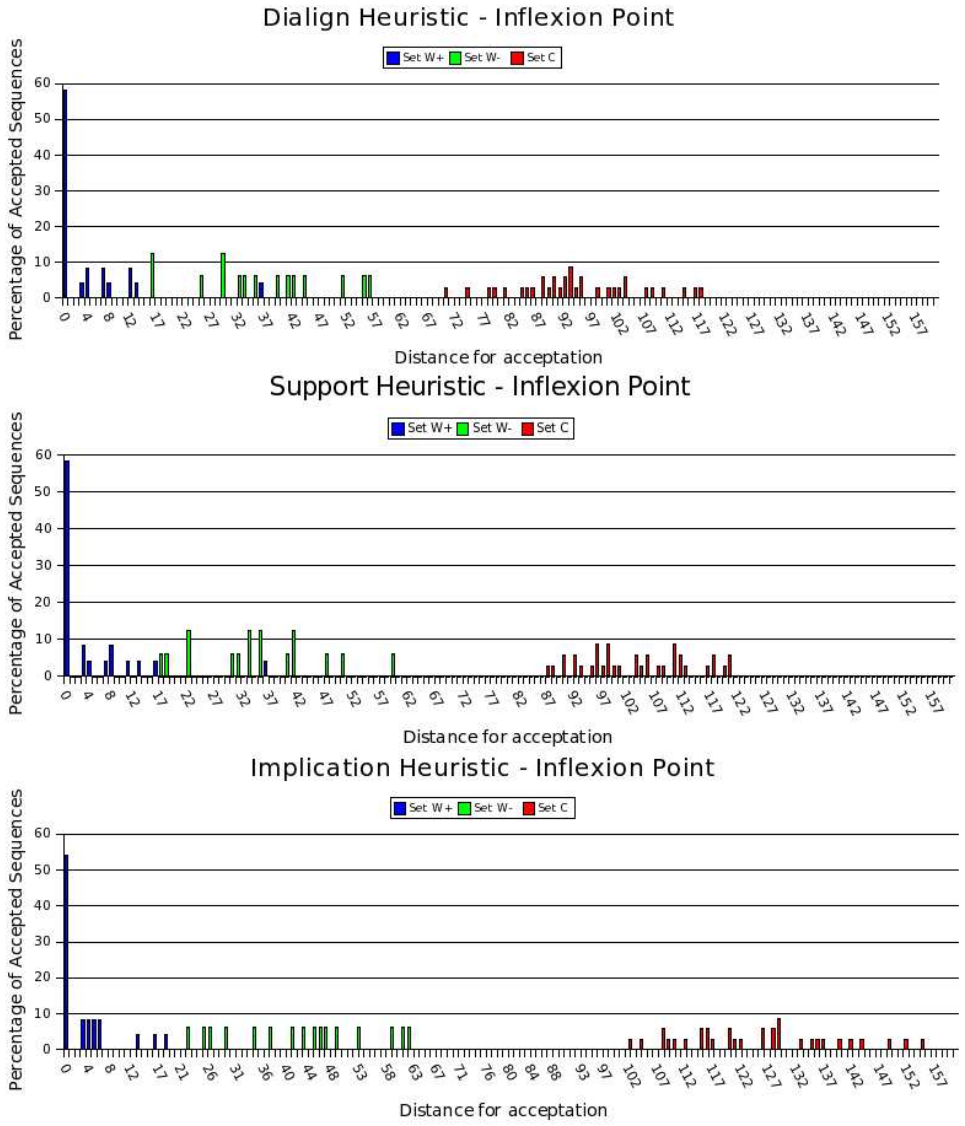


Figure 6: Distribution, for 3 heuristics of percentage of accepted sequences with automata of size Approx. 100. Dialign heuristic refers to ordering the SFP according to their similarity significativity as measured by DIALIGN2 weight function w.

With regard to Grammatical Inference, the confrontation of the classical state-merging techniques with a real application has lead to a new approach based on merging similar fragments. The sole application specific parts are the first and the last step of our approach (the selection of the SFPs step and the physico-chemical properties identification step) and could be replaced by similar modules for other applications. All the remaining of the approach is generic and we expect it to be an inspiration source for new theoretical or algorithmic developments. Among the originalities with respect to the classical approaches, we would like to point out the consideration of the similarity between the symbols of the alphabet, the choice of the non-deterministic representation of automata, the use of fragment-based heuristic to infer this kind of models, the identification of informative positions and the discriminative setting with respect to counter-examples (or unlabeled set of sequences) which replaces the classical compatibility setting and allows to handle some noisy counter-examples.

Acknowledgements:

The authors would like to thank Israël-César Lerman, Basavanneppa Tallur and Anne Siegel for helpfull discussions and ideas about this work.

References

[1] Alvis Brazma, Inge Jonassen, Ingvar Eidhammer, and David Gilbert. Approaches to the automatic discovery of patterns in biosequences. *Journal of Computational Biology*, 5(2):277–304, 1998.

[2] B. Brejova, C. DiMarco, T. Vinar, S. Hidalgo, G. Holguin, and C. Patten. Finding Patterns in Biological Sequences. *Unpublished project report for CS798G*, (CS-2000-22), December 2000.

[3] Andrea Califano. Splash: structural pattern localization analysis by sequential histograms. *Bioinformatics*, 16(4):341–357, 2000.

[4] F. Coste and D. Fredouille. What is the search space for the inference of nondeterministic, unambiguous and deterministic automata ? Technical report, IRISA - INRIA, RR-4907, 2003.

[5] F. Coste and G. Kerbellec. Apprentissage d'automates par fusions de paires de fragments significativement similaires et premières expérimentations sur les protéines mip. In *5èmes Journées Ouvertes de Biologie, Informatique et Mathématiques (JOBIM)*, Montréal, june 2004.

[6] S. Eddy. Hmmer user's guide: biological sequence analysis using prole hidden markov models. http://hmmer.wustl.edu/, 1998.

[7] S. Henikoff and J.G. Henikoff. Amino acid substitution matrices from protein blocks. *Proc. Natl. Acad. Sci. USA*, 89:10915–10919, 1992.

[8] Nicolas Hulo, Christian J. A. Sigrist, Virginie Le Saux, Petra S. Langendijk-Genevaux, Lorenza Bordoli, Alexandre Gattiker, Edouard De Castro, Philipp Bucher, and Amos Bairoch. Recent improvements to the PROSITE database. *Nucl. Acids Res.*, 32(90001):D134–137, 2004.

[9] I. Jonassen, J.F. Collins, and D. Higgins. Finding flexible patterns in unaligned protein sequences. *Protein Science*, 4(8):1587–1595, 1995.

[10] K. El Karkouri, H. Gueune, and C. Delamarche. Mipdb: a relational database dedicated to mip family proteins. *Biol Cell*, 97(7):535–543, July 2005.

[11] K. Karplus. Hidden markov models for detecting remote protein homologies. *Bioinformatics*, 14:846–865, 1998.

[12] K. J. Lang. Random dfa's can be approximately learned from sparse uniform examples. *5th ACM workshop on Computation Learning Theorie*, pages 45 – 52, 1992.

[13] K. J. Lang, B. A. Pearlmutter, and R. A. Price. Results of the abbadingo one DFA learning competition and a new evidence-driven state merging algorithm. *Lecture Notes in Computer Science*, 1433:1–12, 1998.

[14] I.C. Lerman and J. Azé. Indice probabiliste discriminant de vraisemblance du lien pour des données volumineuses. *RNTI-E-1, numéro spécial Mesures de Qualité pour la Fouille des Données, H. Briand, M. Sebag, R. Gras, F. Guillet, CEPADUES*, pages 69–94, 2004.

[15] B. Morgenstern. DIALIGN 2: improvement of the segment-to-segment approach to multiple sequence alignment. *Bioinformatics*, 15:211–218, 1999.

[16] C. Nevill-Manning and I. Witten. Identifying hierarchical structure in sequences: A linear-time algorithm. *Journal of Artificial Intelligence Research*, 7:67–82, 1997.

[17] J. Oncina and P. Garcia. Inferring regular languages in polynomial update time. *Pattern Recognition and Image Analysis*, pages 49 – 61, 1992.

[18] I. Rigoutsos and A. Floratos. Combinatorial pattern discovery in biological sequences: the TEIRESIAS algorithm. *Bioinformatics*, 14(1):55–67, January 1998.

[19] I. Rigoutsos, A. Floratos, L. Parida, Y.Gao, and D. Platt. The emergence of pattern discovery techniques in computational biology. *Metabolic Engineering*, 2:159–177, 2000.

[20] Sakakibara, Brown, Hughey, Mian, Sjolander, Underwood, and Haussler. Recent methods for RNA modeling using stochastic context-free grammars. In *CPM: 5th Symposium on Combinatorial Pattern Matching*, 1994.

[21] William Ramsay Taylor. The classification of amino acid conservation. *Journal of theoretical Biology*, 119:205–218, 1986.

[22] T. Yokomori. Learning non-deterministic finite automata from queries and counterexamples. *Machine Intelligence*, 13:169–189, 1994.