ICOT Technical Memorandom: TM-1236

TM-1236

# Stochastic Motif Extraction using a Genetic Algorithm with the MDL Principle

by
A. Konagaya & H. Kondou (NEC)

November, 1992



ICOT
Mita Kokusai Bldg. 21F
4-28 Mita 1-Chome
Minato-ku Tokyo 108 Japan

(03)3456-3191~5
Telex ICOT J32964

Institute for New Generation Computer Technology

# Stochastic Motif Extraction using a Genetic Algorithm with the MDL Principle

Akihiko Konagaya Hiroyasu Kondou

C&C Systems Research Laboratories., NEC Corporation
1-1, Miyazaki 4-chome, Miyamae-ku, Kawasaki, Kanagawa 216, Japan

## Abstract

*This paper proposes a new methodology to extract "stochastic motifs" from protein sequences. Extracting motifs is not trivial because (1) almost all motifs have exceptions, (2) no quantitative criterion has been available so far for good motifs, and (3) combinatorial explosion may occur when searching for all motif candidates.*

*Instead of pursuing precise motifs, we are trying to extract stochastic motifs that inherently include exceptions, are more stable and suitable for representing important regions. As for the quantitative criterion, we adopt Rissanen's Minimum Description Length (MDL) principle to avoid overfitting to sample sequences. To avoid combinatorial explosion in motif extraction, we adopt a "genetic algorithm", a kind of probabilistic search algorithm based on the biological evolution process. Our experimental results demonstrate that the MDL principle greatly increases the convergence speed of a genetic algorithm when extracting stochastic motifs.*

## 1 Introduction

Recently, some biologists have focused on searching for common patterns in protein sequences which have been preserved in the evolution process. Such patterns are called "motifs" and are considered to represent special biological functions (e.g. Serine proteases and Cysteine proteases) and/or special structures (e.g. Zinc fingers and Leucine zipper consensus)[1]. However, extracting motifs is not trivial because (1) almost all motifs have exceptions, (2) no quantitative criterion has been available so far for good motifs, and (3) combinatorial explosion may occur when searching for all motif candidates.

Common patterns in protein sequences are good approximations of protein functions. A good example is the well-known motif for the heme c binding site in a cytochrome c which plays an important role in the respiratory chain. Figure 1 shows some portions of known cytochrome c sequences for various species. Each character in the sequence corresponds to an amino acid. In most cytochrome c sequences, we can find the common pattern *"CXXCH"* which represents a cysteine,

| Species | Sequence of Cytochrome c |
|---|---|
| Human | ..FIMKCSQCHTVEK.. |
| Mouse | ..FVQKCAQCHTVEK.. |
| Chicken | ..FVQKCSQCHTVEK.. |
| Snake | ..FSMKCGTCHTVEE.. |
| Prawn | ..FVQRCAQCHSAQA.. |
| Yeast | ..FKTHCLQCHTVEK.. |
| Hemp | ..FKTKCAECHTVGR.. |
| Tetrahymena | ..FDSQCSACHAIEG.. |
| Rhodopila | ..FHTICILCHTDIK.. |
| Microbium | ..VFKQCKICHQVGP.. |
| Pseudomonas | ..VFKQCMTCHRADK.. |

Figure 1: Some portions of cytochrome c sequences

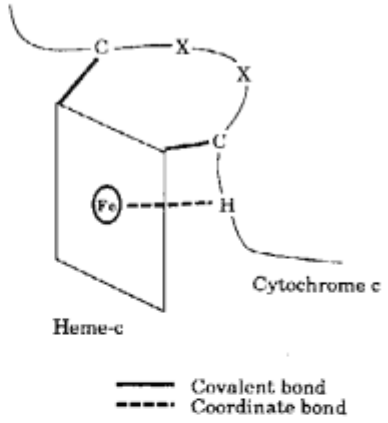


Figure 2: A heme c binding in a cytochrome c

followed by two arbitrary amino acids, followed by another cysteine, followed by a histidine. In this pattern the second "$X$" does not necessarily coincide with the first "$X$". The pattern "$CXXCH$" can be considered as a motif for a cytochrome c, because it corresponds to a protein function; two cysteines and one histidine bind to a heme which cytochrome c holds in the center (Figure 2).

As with other motifs, the pattern "$CXXCH$" also has exceptions. It does not exist in the cytochrome c of Euglena, and the pattern "$CXXCH$" exists in an adrenodoxin of a pig which is a different category from the cytochrome c. In this case, it would be possible to eliminate such exceptions by introducing more complex patterns. However, one should not expect that more complex patterns always represent protein functions more precisely. This is because more complex rules may cause overfitting to sample data and would not necessarily work better for the discrimination of unknown data, especially in the case of learning stochastic rules from noisy data[3]. This suggests that, instead of pursuing precise motifs, we should try to extract more stable motifs which may contain exceptions but work better for the prediction of unknown data. We call such motifs "stochastic motifs" in this paper. The following example gives the flavor of a stochastic motif. "If the pattern $\cdots$ "$CXXCH$" $\cdots$ is included in the sequence, then the sequence is cytochrome c with probability 130/227 and otherwise it belongs to other protein categories with probability 8072/8076." For the representation of a stochastic motif, we also propose a stochastic decision predicate, which consists of Horn clauses and their probability parameters.

To establish a quantitative criterion for stochastic motifs, Rissanen's MDL principle[2] is adopted. This is because overfitting may occur if we try to extract the stochastic motif that best fits the sample protein sequences. We can easily show that the best fitting stochastic motif is unstable in the sense that it varies according to the sampling of sequences. The MDL principle solves this problem by balancing between the complexity of a motif and its classification errors. It gives a strategy of selecting a "good" stochastic motif on the basis of the sum of the bit lengths required to encode a stochastic motif and its logarithmic likelihood to the sample protein sequences. That is, the principle enables us to compare a simple stochastic motif with classification errors and a complex stochastic motif without classification errors, quantitatively.

To avoid the combinatorial explosion in the motif extraction, we use "genetic algorithms", which are a kind of probabilistic search algorithm based on the biological evolution process. The virtue of genetic algorithms is that they offer an efficient generate-and-test search by means of simple genetic operators that simulate "crossover", "mutation" and "selection". Our experimental results demonstrate that the MDL principle plays an essential role for extracting stable stochastic motifs in terms of convergence speed of genetic algorithms. In fact, a genetic algorithm cannot find stable stochastic motifs without the bias to the complexity of stochastic motifs, that is, with a maximum likelihood method, as far as we have seen in our tests.

The organization of the rest of this paper is as follows. Section 2 gives a representation for stochastic motifs, which we call *Stochastic Decision Predicates*. Section 3 gives a strategy for selecting a good stochastic motif using the MDL principle. Section 4 gives an algorithm for finding optimal stochastic motifs. Section 5 presents the experimental results on extracting stochastic motifs based on our proposed methodology. Finally, in section 6 we discuss current difficulties and future work. This work has been done as a part of the fifth generation computer systems project for the evaluation of the parallel inference machines.

## 2 Stochastic Decision Predicates

There are many ways to represent stochastic motifs. As a first step for a stochastic representation of motifs, we devised the stochastic decision predicate, a natural extension of a decision list with probabilities. The stochastic decision predicate consists of Horn clauses with probability parameters as follows.

```
motif(S,cytochrome_c) with 137/244.
     :- contain(S,''CXXCH'').
motif(S,others) with 9386/9389.
```

The general form is the following.

$$
\begin{aligned}
&motif(S, C_1) \quad (\text{with } p_1) \quad :- Q_1^{(1)} \wedge \cdots \wedge Q_{k_1}^{(1)}.\\
&motif(S, C_2) \quad (\text{with } p_2) \quad :- Q_1^{(2)} \wedge \cdots \wedge Q_{k_2}^{(2)}.\\
&\qquad \cdots\cdots\cdots\cdots\\
&\qquad \cdots\cdots\cdots\cdots\\
&motif(S, C_{m-1}) \ (\text{with } p_{m-1}) : \ Q_1^{(m-1)} \wedge \cdots \wedge Q_{k_{m-1}}^{(m-1)}.\\
&motif(S, C_m) \quad (\text{with } p_m) \ : \ Q_1^{(m)} \wedge \cdots \wedge Q_{k_m}^{(m)}.
\end{aligned}
$$

Here we call each "$motif(S, C_i)$ (with $p_i$) :- $Q_1^{(i)} \wedge \cdots \wedge Q_{k_i}^{(i)}$." a *stochastic clause*. The stochastic clause can be read as $S$ is categorized into $C_i$ with probability $p_i$ if $Q_1^{(i)}, \cdots, Q_{k_i}^{(i)}$ are all **true**. We assume sequential interpretation of the stochastic clauses in this paper. That is, $motif(S, C_i)$ is selected after $motif(S, C_1), \cdots, motif(S, C_{i-1})$ are examined. The body goals $Q_1^{(i)} \wedge \cdots \wedge Q_{k_i}^{(i)}$ $(i = 1, \cdots, m)$ represent a condition to discriminate a category $C_i$ when $S$ is given. Each goal $Q_j^{(i)}$ consists of the disjunction of goals $R_{1_j}^{(i)}; \cdots; R_{h_j}^{(i)}$ where $R_{h_j}^{(i)}$ represents some predicate that discriminates a category $C_i$, such as $contain(S, \sigma)$ which is *true* when $S$ contains a pattern $\sigma$.

### 2.1 Semantics of Stochastic Decision Predicate

The semantics of stochastic decision predicates are given from the viewpoint of computational learning theory of stochastic rules[3]. A stochastic decision predicate represents a probabilistic mapping from protein sequences to categories. The probabilistic mapping can be regarded as a conditional probability distribution over the categories when a sequence is given, by introducing a probability structure on the

sequence–category pairs. See the paper [4] for the formal approach to learning stochastic motifs.

## 3 The MDL Principle in Motif Extraction

In our methodology, the MDL principle gives a new quantitative criterion for "good" stochastic motifs. The most important point is that it enables us to avoid overfitting when extracting stochastic motifs.

For example, as we have shown in the previous section, the pattern "$CXXCH$" has exceptions in the cytochrome c. It is possible to avoid these exceptions by adding more conjunctions and disjunctions of patterns such as "$AAQCH$" and "$PGTKM$". However, care must be taken so that the obtained result does not become too complex and overfit to the sample sequences. Therefore, we adopt the MDL principle to extract simple but stable stochastic motifs which may contain exceptions rather than precise motifs without exceptions.

The MDL principle originally comes from coding theory in communication. The basic idea is to optimize the number of bits when sending a piece of information, by means of encoding a rule and its exceptions in the piece of information. The MDL principle selects a rule such that minimizes the total bit length of the rule and the exceptions.

The flavor of the MDL principle is the following. Suppose there is a binary string "101101100". Sending the string requires 9 bits if we do not use any rule. Less bits are sufficient if we compress the string using a rule and its exception. In this case, we can represent the string as three repeats of "10*" and exceptions "110" for the third bit of each repeat instead of * in the rule. The rule requires $log3^3 = 4.75$[1] bits since we have to choose on of $3^3$ varieties that represent 3-character rules using three kinds of characters. The exception requires $log2^3 = 3.0$ bits. The total bits becomes 7.75 bits. We may find a more complex rule to reduce the number of exceptions, but such a rule might require a longer bit length. Therefore, it is important to balance the complexity of the rule and the number of exceptions to reduce the total bit length.

In our methodology, we apply the MDL principle for extracting stochastic motifs in the way proposed by Yamanishi for learning stochastic rules: Yamanishi's MDL learning algorithm[3]. In his algorithm, the MDL principle selects a stochastic rule that balances the complexity of the stochastic rule and its likelihood of matching the sample data. The rest of this section follows his algorithm with slight modification which mainly comes from the difference of stochastic rule representation, that is, stochastic decision lists and stochastic decision predicates, and some practical reasons for applying the MDL learning algorithm to the motif extraction.

Our methodology selects a stochastic motif that balances the complexity of representation and likelihood of matching the sample sequences. The complexity of a stochastic motif representation is measured by the description lengths to encode the probability parameters and the Horn clauses of a stochastic decision predicate. The likelihood of a stochastic motif is measured by the description length of likelihood, that is, by the logarithmic likelihood of categories when the sequences are given to the stochastic motif. The description lengths are calculated as follows.

### 3.1 Description Length of Likelihood

Let $\ell(L)$ be the description length of likelihood given by logarithmic likelihood of categories when sequences are given to a stochastic motif. The likelihood of the categories can be calculated using probabilities associated for categories on each Horn clause in the stochastic motif.

Let $(S_1, C_1), \cdots, (S_N, C_N)$ be given N sample sequence and category pairs. Let $E_j$ be the set of sequences which are **false** for the $1, \cdots, j-1$th clauses and are **true** for the $j$th clause. Let $N_j$ be the number of sequences in $E_j$ and let $N_j^+$ be the number of sequences which are in $E_j$ and belong to the category of the $j$–th clause. Then the likelihood of the categories $(C_1, \cdots, C_N)$ when given sample sequences $(S_1, \cdots, S_N)$ with respect to a stochastic predicate with probabilities $(p_1, \cdots, p_m)$, which we denote $L$, is calculated as follows.

$$L = \prod_{j=1}^{m} p_j^{N_j^+} (1-p_j)^{N_j - N_j^+}.$$

The description length $\ell(L)$ is given by $-\log L$ which can be calculated, as follows:

$$\ell(L) = \sum_{i=1}^{m} N_i \{H(\tilde{p}_i) + D_{KL}(\tilde{p}_i \parallel \hat{p}_i)\} \qquad (1)$$

where $\tilde{p}_i = N_i^+/N_i$ and $\hat{p}_i$ is an estimate of the true parameter $p_i^*$, which is set to be $\frac{N_i^+ + 1}{N_i + 2}$(the Bayes estimator) to avoid the difficulties of calculating the description length when $N_i^+ = 0$ *or* $N_i$. In addition, $H(\tilde{p}_i)$ and $D_{KL}(\tilde{p}_i \parallel \hat{p}_i)$ are the entropy function and Kullback-Leibler divergence defined as follows.

$$H(\tilde{p}_i) = -\tilde{p}_i \log \tilde{p}_i - (1-\tilde{p}_i)\log(1-\tilde{p}_i)$$

$$D_{KL}(\tilde{p}_i \parallel \hat{p}_i) = \tilde{p}_i \log \frac{\tilde{p}_i}{\hat{p}_i} + (1-\tilde{p}_i) \log \frac{1-\tilde{p}_i}{1-\hat{p}_i}$$

The description lengh $\ell(L)$ indicates the number of bits required to encode the distribution of positive examples and negative examples relative to the stochastic decision predicate. The length varies from near 0 bit[2], when $p_i = 0$ *or* $1.0$ $(i = 1, \cdots, m)$, to N bits, when $p_i = 0.5 (i = 1, \cdots, m)$. The former occurs when the stochastic decision predicate completely discriminates the target categories in the given sequences. The latter occurs when the stochastic decision predicate does not contribute to any discrimination of the given sequences.

[1] "log" denotes logarithm with base 2.

[2] It is not appropriate to neglect the value of Kullback-Leibler divergence when the value of entropy function is small.

### 3.2 Description Length of Probabilities

Let $\ell(P)$ be the description length of the probabilities $\hat{P} = (\hat{p}_1, \cdots \hat{p}_m)$ for a stochastic decision predicate. Since the accuracy (variance) of the maximum likelihood estimator is $O(1/\sqrt{N})$, the description length $\ell(P)$ is given by:

$$\ell(P) = \sum_{i=1}^{m} \frac{\log N_i}{2} \tag{2}$$

### 3.3 Description Length of Horn Clauses

Let $\ell(M)$ be the description length of the Horn clauses $M$. $\ell(M)$ significantly depends on the encoding scheme from Horn clauses to binary strings. The scheme ought to be designed so that the description length can reflect the complexity of the Horn clauses.

In the motif extraction system, $\ell(M)$ is given by:

$$\begin{aligned}\ell(M) \quad = \quad & \sum_{i=1}^{m} [\ \log^*(\sum_{j=1}^{k_i} h_j) + (\sum_{j=1}^{k_i} h_j - 1) \\ & + \sum_{j=1}^{k_i} \sum_{l=1}^{h_j} \{\log \begin{pmatrix} L_l^j(i) \\ X_l^j(i) \end{pmatrix} \\ & + (L_l^j(i) - X_l^j(i)) * \log(\mid \mathcal{A} \mid -1)\} + \log r \ \}\end{aligned} \tag{3}$$

where $L_l^j(i)$ and $X_l^j(i)$ are the number of amino acids and of variables, respectively, in the pattern in the $l-$th predicate in the $j-$th disjunction region of the $i-$th clause. On the righthand of (3), the first term denotes the description length of the number of *contain* predicates in the $i-$th clause. For any $d > 0$, $\log^* d$ denotes $\log d + \log\log d + \cdots$ where the sum is taken over all positive terms (Rissanen's integer coding scheme [5]). The second term of (3) denotes the description length to encode the disjunctions and conjunctions occurring in the $i-$th clause. The third term denotes the description length of the positions of variables in the pattern $\sigma$ appearing in the predicate '$contain(S, \sigma)$.' The fourth term denotes the description length required to describe amino acids (not variables) included in the pattern $\sigma$ appearing in the predicate '$contain(S, \sigma)$'. $\mid \mathcal{A} \mid$ is 20 for amino acids. The last term $\log r$ denotes the description length of the category $C$ appearing in the predicate '$motif(S, C)$'.

### 3.4 Description Length of Stochastic Motif

By summing (1), (2), and (3), we have the following description length $\ell(T)$ of a stochastic motif represented by a decision predicate:

$$\begin{aligned}& \ell(T) \\ & \stackrel{\text{def}}{=} \ell(L) + \lambda\{\ell(P) + \ell(M)\}\end{aligned} \tag{4}$$

where $\lambda$ is the adjustment parameter. The MDL principle asserts that one should select the stochastic motif which minimizes the description length $\ell(T)$ . Notice here that it is still computationally intractable to find the stochastic motif that minimizes the description length $\ell(T)$ when all possible combinations of Horn clauses are large. Next, we will discuss algorithms to avoid this combinatorial explosion of the search space.

## 4 Genetic Algorithms

Genetic algorithms are stochastic search algorithms based on the biological evolution process[6]. As in figure 3, genetic algorithms simulate the survival of the fittest in a population of individuals which represent points in a search space. The individuals are represented by binary strings. A function, often called a fitness function, gives values to the binary strings. The aim of a genetic algorithm is to find a global optimum of the fitness function when given an initial population of individuals by applying genetic operators in each generation. The genetic operators consist of the following: crossover, mutation and selection.

### Crossover

The crossover operator produces two descendants by exchanging part of two individuals. This operator aims to make a better individual by replacing a part of an individual with a better part of another individual. For example, crossover of the strings "000110" and "110111" at the third position produces the strings "000111" and "110110". The candidates of the crossover operation and the crossover position are randomly chosen.

### Mutation

The mutation operator changes certain bit(s) in an individual. For example, the string "000110" becomes "001110" if mutation occurs at the third bit. This operation aims to escape from search spaces from which individuals cannot escape by means of only the crossover operator.

### Selection

The selection operator chooses good individuals in a population according to their fitness values and the given selection strategy. This operator aims to increase better individuals in the population while maintaining certain diversity. It simulates the survival of the fittest principle. The operator first calculates the relative fitness of all individuals. Then, several lesser individuals are discarded and the same number of better individuals are duplicated according to their relative fitness values. In case of roulette wheel selection strategy, it selects the next individuals with the probabilities in proportion to their retative fitness values. So, better individuals have a higher chance of remaining or being duplicated but this is not guarranteed.

One of interesting characteristics of our genetic algorithm is in its use of the MDL principle to calculate the fitness value of an individual motif. The MDL length gives the natural relative fitness values in the population, although the smaller the better in this case.

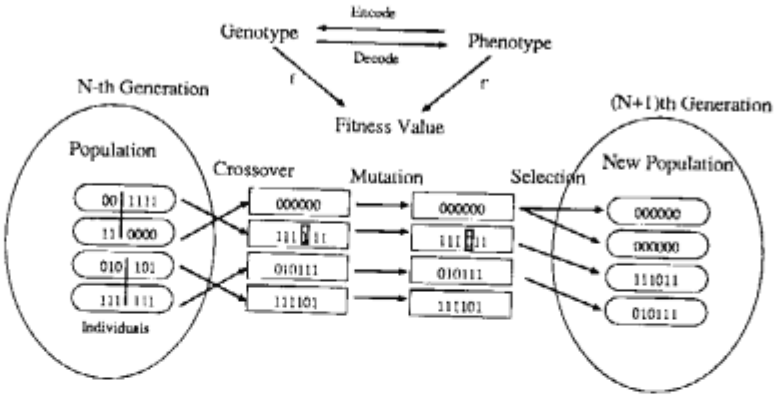


Figure 3: Mechanism of Simple Genetic Algortithms

## 5 Evaluation

### 5.1 The Experimental Motif Extraction System

The overview of our experimental motif extraction system is the following. The target hypothesis space is the domain of stochastic decision predicates. The search strategy is based on the MDL principle. The search algorithm is an asynchronous parallel genetic algorithm which consists of the set of subpopulations in which individuals migrate asynchronously. In each subpopulation, individuals represent stochastic decision predicates in the target hypothesis space, and fitness function calculates the corresponding description lengths of the stochastic decision predicates.

The search time depends considerably on the size of the hypothesis space. A large hypothesis space makes it difficult for us to find the optimal stochastic decision predicate in a reasonable time. Therefore, as the first step of motif extraction, we restricted the stochastic predicates to the following forms.

```
motif(S,proteinClass) with p1
   :- contain(S,pattern1) and
      contain(S,pattern2) ...
motif(S,others) with p2.
```

That is, we use a predicate *motif* which discriminates the target protein category *proteinClass* from other proteins (*others*) in the database. The discrimination conditions are represented by the conjunction of a predicate *contain*. As the pattern candidates in the *contain* predicate, we adopt 128 patterns that occur frequently in the target proteins.

The mapping from a stochastic decision predicate to a binary string is the following. Each bit corresponds to one of the 128 patterns. A bit 1 represents the occurrence of the pattern in a discrimination condition, and a bit 0 represents the pattern does not occur in the discrimination condition. For example, suppose we use 3-bit length binary strings whose first, second, third bits correspond to the pattern "$CXXCH$", "$PXLXG$", "$GXKM$", respectively. Then, the binary string "100" represents the following stochastic decision predicate.

```
motif(S,proteinClass) with p1
     :- contain(S,"CXXCH").
motif(S,others) with p2.
```

The binary string "011" represents the following stochastic decision predicate.

```
motif(S,proteinClass) with p1
 :- contain(S,"PXLXG") & contain(S,"GXKM").
motif(S,others) with p2.
```

According to this mapping, 128 bits binary strings can express $2^{128}$ kinds of stochastic decision predicates. As for the genetic operators, we adopt one-point crossover, one-point mutation and roulette wheel selection as described in section 4. The values of other runtime parameters are: the adjustment parameter is 1.0, the number of subpopulations is 63, the subpopulation size is 16, the crossover rate is 1.0, the mutation rate is 0.01 and the migration rate is 0.5, that is, one individual per two generations in average.

### 5.2 Experimental Results

Table 1 contains some of the stochastic motifs extracted by our experimental system when applied to the protein categories that have more than 10 entries in the Protein Identification Resources (PIR32.0) which currently has 9633 entries[3]. The rest of results are presented in the appendix.

In table 1, the column $PC$ is the super family number of the protein category in PIR32.0. The column $StochasticMotif$ is the conjunctions of patterns extracted by our system. The columns $\ell(T)$,$\ell(M)$, $\ell(P)$ and $\ell(L)$ are description lengths of a stochastic motif,

[3]Annotated and classifed entries by homology in pirl.dat.

Table 1: Results of Stochastic Motif Extraction

| PC | StochasticMotif | $\ell(T)$( | $\ell(M)$, | $\ell(P)$, | $\ell(L)$), | $E$, | $N_1^+$, | $N_1$, | $N_2^+$, | $N_2$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | CXXCH | 309.844( | 18.288, | 10.564, | 280.693), | 140, | 137, | 244, | 9386, | 9389 |
| 11 | CXXCH& GXXCXXC | 94.611( | 38.383, | 9.114, | 47.114), | 17, | 16, | 32, | 9600, | 9601 |
| 14 | IXXXWY& WGXT | 47.312( | 36.255, | 8.346, | 2.713), | 11, | 11, | 11, | 9622, | 9622 |
| 21 | FXXGXXXC&CXGXXXA | 95.705( | 40.868, | 9.179, | 45.658), | 33, | 31, | 35, | 9596, | 9598 |
| 23 | GXCXXC& CXXGXC& FXSC | 447.695( | 55.201, | 9.435, | 383.059), | 91, | 50, | 50, | 9542, | 9583 |
| 29 | HNXV& PXXXXXMXG | 180.516( | 38.324, | 9.068, | 135.124), | 35, | 25, | 30, | 9593, | 9603 |
| 30 | GXRD& RDXXXXL&LRXXXY | 102.494( | 36.008, | 8.877, | 37.609), | 23, | 21, | 23, | 9608, | 9610 |
| 33 | CXXCXXC&GHE | 74.458( | 35.061, | 8.877, | 30.520), | 20, | 19, | 23, | 9609, | 9610 |
| 38 | GXHXD& HGD& RPR | 72.659( | 47.879, | 8.399, | 16.381), | 13, | 12, | 12, | 9494, | 9495 |
| 63 | AXCXXN& WXXNE | 63.554( | 37.575, | 9.275, | 16.503), | 41, | 40, | 40, | 9592, | 9593 |
| 80 | LXXRXN& PXPXXXN&PXXXRXN | 116.070( | 59.137, | 8.616, | 48.316), | 17, | 14, | 16, | 9614, | 9617 |
| 105 | AYXS& MXXYG& YSS | 74.438( | 49.879, | 8.201, | 16.358), | 10, | 9, | 9, | 9628, | 9624 |
| 117 | HXXXMXP&IPF | 46.195( | 35.061, | 8.408, | 2.725), | 12, | 12, | 12, | 9621, | 9621 |

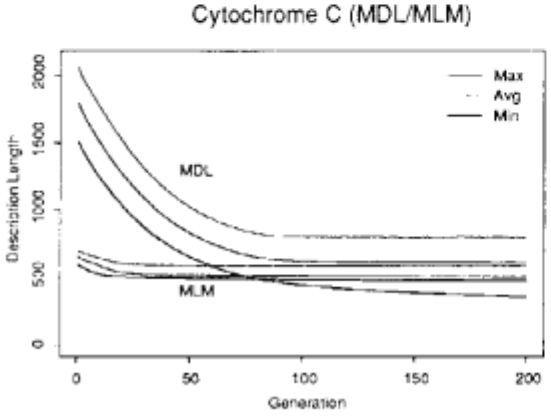


Figure 4: Average description lengths of the best stochastic motif encountered in each generation

Horn clauses, probabilities, and a logarithmic likelihood to the sample sequences.

The column $E$ is the number of target protein sequences in the protein sequence database (PIR). The column $N_1$,$N_2$ is the number of protein sequences that become *true* in the first, second clause of a stochastic decision predicate. The column $N_1^+$, $N_2^+$ is the number of protein sequences which belong to the target protein category in $N_1$,$N_2$, respectively.

The correspondence between the obtained stochastic motifs and biologically meaningful regions remains as future research issues.

### 5.3 Comparison of the MDL principle and the Maximum likelihood method

To demonstrate the effectiveness of the MDL principle, various indexes including prediction errors, convergence speed are compared to the maximum likelihood method (MLM). In MLM, good individuals are selected using only the description length of likelihood ($\ell(L)$) without consideration for the complexity of a stochastic decision predicate ($\ell(M) + \ell(P)$).

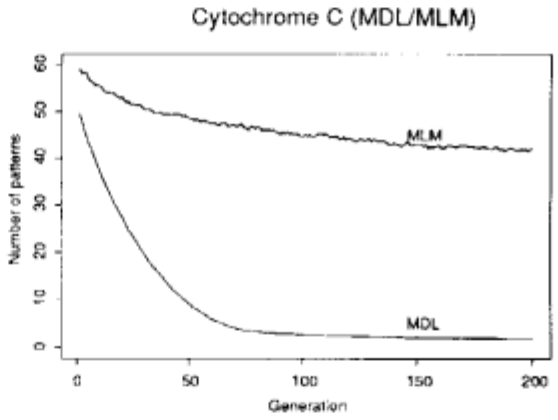


Figure 5: Average number of patterns of the best stochastic motif encountered in each generation

Table 2: Prediction errors for Cytochrome C by Cross Validation Method

| | MDL-GA | MLM-GA |
|---|---|---|
| $\sum_{i=1}^{10} E_i^+$ | 3 | 57 |
| $\sum_{i=1}^{10} E_i^-$ | 96 | 0 |
| $Total$ | 99 | 57 |

Using cross validation technique ([7] p.75-76), the prediction errors can be counted as follows. Let $S_i$ be a disjoint subgroup of protein sequences $S$ for certain $N$ where $S = \cup_{i=1}^{N} S_i$. Let $S_i'$ be a sample set which removes the $i$ th subgroup from the original protein sequences ($S_i' = S - S_i$). Then, let $M_i$ be a stochastic motif extracted from the sample set $S_i'$, and count the number of prediction errors $E_i^+$ and $E_i^-$ using the subgroup $S_i$ as a test set, where $E_i^+$ shows the number of protein sequences that belong to the target protein category but are not *true* for the first clause of the stochastic motif $M_i$. $E_i^-$ shows the number of protein sequences that do not belong to the target protein category but are *true* for the first clause of the stochastic motif $M_i$.

Table 2 shows the prediction errors for cytochrome c by the cross validation method when divided into 10 subgroups. The best scored stochastic motif is selected from 50 trials for each subgroups. Each trial requires 200 genetic algorithm generations.

The results show that the stochastic motifs obtained using a genetic algorithm with the MDL principle (MDL-GA) are more stable than the ones obtained using a genetic algorithm with the ML method (MLM-GA). As seen in table 2, the stochastic motifs obtained by the genetic algorithm with the ML method is apparently overfitted to the sample protein sequences. It shows strong discrimination performance for the sample protein sequences ($\sum_{i=1}^{10} E_i^-$), but shows weak predictive performance for the test sequences ($\sum_{i=1}^{10} E_i^+$).

Contrary to our expectations, the result does not comes from the intrinsic difference between MDL and MLM, but comes from the difference of convergence speed between MDL-GA and MLM-GA. As in figure 4, MDL-GA shows good covergence speed compared to MLM-GA. That is, MLM-GA is too slow to give us the global optimum in the search space within reasonable time. The difference of the convergence speed comes from the bias caused by the MDL principle. As shown in figure 5, MDL-GA rapidly decreases the number of patterns in the best stochastic motif encountered, while MLM-GA gradually decreases. This is natural since the description length of Horn clauses basically corresponds to the number of patterns. In other words, the MDL principle gives a bias for GA to select individuals with fewer patterns.

Figure 6 shows the effectiveness of the bias for the convergence speed of a genetic algorithm with the MDL principle by changing the adustment parameter ($\lambda$) from 0.5 to 2.0. The histogram in the figure 6 shows the distribution of generations in which the optimal solution ($CXXCH$) is found. In case of $\lambda = 0.0$, that is, the genetic algorithm with the maximumly likelihood method, no optimum stochastic motif is found so far as 10000 generations. In addition, the same stochastic motif has not been extracted even in 10000 trials.

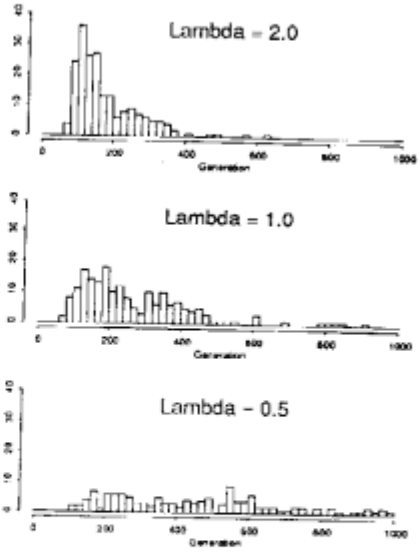


Figure 6: Comparison of convergenece speed by the distributions of generations in which the optimal solution is found

## 6 Discussion

The following work remains to deal with actual protein sequences on the basis of our methodology.

- The extension of stochastic decision predicate form: In our experience, the number of categories for discrimination is limited to two, that is, the target category and the others. A stochastic decision predicate over two categories can be constructed by concatenating the obtained stochastic clauses for each protein category and recalculating the probabilistic parameter, although it causes another combinatorial problem; in the order of protein categories. Another interesting extension is providing other predicates, such as a distance between patterns.

- Disjunction of patterns: In the current implementation, no form is provided for the disjunction of patterns on the mapping from stochastic decision predicates to binary strings on the genetic algorithm. For example, the pattern "$CXXCH \vee AXXCH$" may be more appropriate since it eliminates three exceptions caused by Euglinae.

Finding the pattern "$AXXCH$" is possible if we use (multiple) alignment information of homologous protein sequences.

- More complex patterns: The patterns we used in our experiments are too simple to reflect protein functions. For example, it is a well known fact that in the heme-c binding motif "$CXXCH$", neither histidine, cysteine, proline nor tryptophan occur in "XX" and small amino acids tend to occur there. To represent such information, more complex patterns are required. Our early experience shows that hidden markov models (HMM) seems to be appropriate for this purpose.

- The handling of category hierarchy: The current MDL principle might select too simple stochastic motifs which have nothing to do with the protein categories. For example, the MDL principle might select only "$PGTKM$" instead of "$CXXCH \wedge PGTKM$" for a mitochondria cytochrome c, a subcategory of a cytochrome c. Such selection is tolerable for the purpose of database search, but less effective in the sense that it might lose biological meaning. Such over-simplification can be avoided by adding constraint that reflects category hierarchy.

- Reducing hypothesis space: Since the MDL principle has a bias against selecting complex patterns, it is possible to eliminate complex patterns, for example, more than five patterns from the hypothesis space. One may think it would be faster to search all candidates less than four patterns than to use a genetic algorithm. However, genetic algorithms are also faster if we change their mapping so that it only represents combinations of less than four patterns. In addition, we might bias to the description length of Horn clauses. If this is true, we have to change the adjustment parameter, and also have to search a larger hypothesis space which may include complex patterns more than five patterns. In that case, genetic algorithms would be more powerful tools than conventional search algorithms.

- The handling of point mutations and experimental ambiguity: For example, actual amino acid sequences contain mutation information and special characters that represent ambiguous elements, such as B for asparagine or asparatic acid, and Z for glutamine and glutamic acid. The disjuctive form of stochastic decision predicates may help to some extent. However, such information should be counted for the calculation of description lengths of the stochastic motifs.

## 7 Conclusion

We have proposed a new methodology for extracting stochastic motifs from protein sequences. Our proposed methodology is characterized by the stochastic representation of motifs using stochastic decision predicates, quantitative criterion using the MDL principle and fast search algorithms using genetic algorithms. Our experimental results show that the methodology actually produces stable motifs from real protein sequences. The effectiveness of the MDL principle has been statistically proven and compared to the maximumly likelihood method, although data are limitted to cytochrome c in this paper. We believe the methodology can also be applied to the various kind of discrimination problems on genetic information such as protein sequences. This work has been done as a part of fifth generation computer systems project for the evaluation of parallel inference machines.


## Acknowledgements

The authors wish to express their sincere gratitude to Dr. K. Nitta of ICOT, and to Dr. N. Koike of NEC Corporation for their encouragement and support. The authors thank Dr. K. Yamanishi for his technical advices especially for the MDL principle. The authors also thank Mr. K. Yamagishi, Mr. S. Oyanagi, Ms A. Ikeda of NSIS, and Ms Y. Kobayashi and Ms K. Hikita of C&C Systems Research Laboratories, NEC Corporation for their great contribution to our analysis on real protein sequences.


## References


[1] Aitken, Alastair, (1990). *Identification of Protein Consensus Sequences*, Ellis Horwood Series in Biochemistry and Biotechnology.

[2] Rissanen, J.(1978). Modeling by shortest data description. *Automatica, 14*, 465-471.

[3] Yamanishi, K.(1990). A learning criterion for stochastic rules. *Proceedings of the 3-rd Annual Workshop on Computational Learning Theory*, (pp. 67-81), Rochester, NY: Morgan Kaufmann. Its full version is to appear in Jr. on Machine Learning.

[4] Yamanishi, K. & Konagaya, A.(1991). Leaning Stochastic Motifs from Genetic Sequences. *in Proc. of the Eighth International Workshop of Machine Learning.*

[5] Rissanen, J.(1983). A universal prior for integers and estimation by minimum description length. *Annals of Statistics, 11*, 416-431.

[6] Goldberg,D.E., (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley Publishing Company, Inc.

[7] Breiman, L., Friedman, J.H., Olshen, R.A., & Stone, C.J.(1984). Classification and regression trees. *Wadsworth Statistics/Probability Series.*

## APPENDIX: Continued from Table 1

| PC | StochasticMotif | t(T)( | t(M), | t(P), | t(L)), | E, | $N_1^+$, | $N_1$, | $N_2^+$, | $N_2$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 118 | CSE& QWY | 41.255( | 29.932, | 8.569, | 2.734), | 15, | 15, | 15, | 9616, | 9616 |
| 119 | HHXG& WLF | 42.906( | 31.932, | 8.277, | 2.698), | 10, | 10, | 10, | 9623, | 9623 |
| 154 | GXHXXPE HXGD | 59.337( | 36.253, | 8.616, | 14.468), | 12, | 12, | 16, | 9617, | 9617 |
| 162 | CDP& CXXPXXP | 46.036( | 33.061, | 8.277, | 2.698), | 10, | 10, | 10, | 9623, | 9623 |
| 168 | HXYXXH& HSXD | 51.720( | 36.253, | 8.320, | 6.947), | 13, | 13, | 14, | 9619, | 9619 |
| 177 | BNV& PCXXXS& QXRXR | 67.404( | 52.201, | 8.466, | 6.827), | 12, | 12, | 13, | 9620, | 9620 |
| 201 | FCT& RXMM | 43.451( | 31.932, | 8.739, | 2.780), | 19, | 19, | 19, | 9614, | 9614 |
| 251 | EXXPF& GPM | 51.347( | 35.253, | 8.466, | 9.628), | 11, | 11, | 13, | 9620, | 9620 |
| 252 | EXXXYW& PXXYW | 49.214( | 37.575, | 8.845, | 2.793), | 22, | 22, | 22, | 9611, | 9611 |
| 255 | DFG& HXDAXXXN | 246.366( | 35.739, | 9.986, | 203.641), | 107, | 95, | 108, | 9512, | 9325 |
| 281 | DXNV& WNXP | 54.242( | 33.932, | 8.877, | 11.433), | 21, | 21, | 25, | 9610, | 9610 |
| 304 | FXQF& FYH | | | | | | | | | |
| | AFY& FXQF | 52.559( | 31.932, | 8.937, | 11.691), | 23, | 23, | 25, | 9608, | 9608 |
| 306 | YXGXXV& FXSXY& LYXXI | 66.846( | 55.525, | 8.569, | 2.754), | 15, | 15, | 15, | 9618, | 9618 |
| 348 | YGC& YGXXC | 78.021( | 33.253, | 9.516, | 35.251), | 57, | 55, | 56, | 9575, | 9577 |
| 360 | NYC& RAH | 40.950( | 29.932, | 8.346, | 2.713), | 11, | 11, | 11, | 9622, | 9622 |
| 404 | CKXXXT& TFXH | 54.163( | 34.253, | 9.293, | 8.617), | 40, | 40, | 41, | 9592, | 9592 |
| 408 | LXXWXW& NDD | | | | | | | | | |
| | KNW& LXAWXW | | | | | | | | | |
| | EFK& LXXWXW | 45.387( | 34.253, | 8.408, | 2.725), | 12, | 12, | 12, | 9621, | 9621 |
| 407 | DXXRXD& PXXNH& NHD | 63.259( | 52.201, | 8.346, | 2.713), | 11, | 11, | 11, | 9622, | 9622 |
| 415 | GXXXWXA&IXXXXWC | 78.106( | 40.190, | 8.937, | 28.979), | 27, | 25, | 25, | 9606, | 9608 |
| 420 | KSC& YCXXI | 44.994( | 33.253, | 8.937, | 2.804), | 25, | 25, | 25, | 9608, | 9608 |
| 421 | MXPN& NQK | 61.233( | 31.932, | 8.569, | 20.732), | 15, | 14, | 15, | 9617, | 9618 |
| 456 | GWO& CXXDXG | 124.490( | 34.253, | 9.435, | 80.802), | 40, | 37, | 50, | 9580, | 9583 |
| 473 | NSW& YWXXXN | 45.815( | 34.253, | 8.776, | 2.785), | 20, | 20, | 20, | 9613, | 9613 |
| 476 | PXXXFD& VPXXXC | 80.892( | 38.575, | 8.700, | 33.326), | 18, | 17, | 18, | 9613, | 9613 |
| 485 | DXGA& GAD& GXDXXXQ | 98.757( | 51.686, | 8.408, | 38.663), | 10, | 8, | 12, | 9619, | 9621 |
| 513 | ELXXXD& DXIA& YXPT | 96.543( | 52.879, | 9.136, | 34.506), | 35, | 33, | 34, | 9597, | 9599 |
| 518 | GTA& HAPE VLXXXA | 110.064( | 48.879, | 8.408, | 58.776), | 14, | 10, | 12, | 9617, | 9621 |
| 520 | ARXP& IGXG& AXXGXG | 122.084( | 52.879, | 9.137, | 60.069), | 16, | 14, | 33, | 9596, | 9600 |
| 521 | RLXXN& FIXXXL&LLXI | 122.422( | 55.008, | 8.477, | 58.537), | 20, | 17, | 23, | 9607, | 9610 |
| 546 | WXXWXXP&WXLXXP | 80.911( | 39.385, | 9.676, | 40.852), | 73, | 70, | 70, | 9560, | 9563 |
| 547 | KDD& PKL& YGR | | | | | | | | | |
| | NVF& NYG& YGR | | | | | | | | | |
| | GPP& KDD& NVF | | | | | | | | | |
| | FEE& KNY& NVF | | | | | | | | | |
| 552 | KXNM& KWR | 56.076( | 44.557, | 8.739, | 2.780), | 19, | 19, | 19, | 9614, | 9614 |
| 561 | GDXS& GXRP& YPG | 63.134( | 51.932, | 8.466, | 2.736), | 13, | 13, | 13, | 9620, | 9620 |
| 577 | HXXWG& HWN | 69.635( | 48.587, | 8.346, | 32.732), | 12, | 10, | 11, | 9620, | 9622 |
| 616 | FHR& PXXXXNF | 64.228( | 33.253, | 8.277, | 2.698), | 10, | 10, | 10, | 9623, | 9623 |
| 617 | HSE& YEXXW | 71.920( | 35.061, | 8.811, | 27.648), | 19, | 16, | 21, | 9611, | 9612 |
| 661 | GXRXR& GXHXXXS&NXXGXH | 44.519( | 33.253, | 8.320, | 2.746), | 14, | 14, | 14, | 9619, | 9619 |
| 694 | CXXFXY& FXYXXC | 97.367( | 57.330, | 9.293, | 30.744), | 39, | 38, | 41, | 9591, | 9593 |
| 696 | PXCG& TYXXXC | 88.033( | 38.575, | 8.877, | 40.581), | 22, | 20, | 23, | 9608, | 9610 |
| 697 | WAXXFXXXXT&VXMM | 103.289( | 36.253, | 8.569, | 56.466), | 18, | 14, | 18, | 9614, | 9618 |
| 704 | TXCXK& YXXPW | 136.720( | 38.658, | 8.739, | 89.145), | 25, | 18, | 19, | 9607, | 9614 |
| 708 | CCXXXXC&PXXCXC | 131.305( | 36.575, | 8.659, | 86.071), | 19, | 13, | 17, | 9610, | 9616 |
| 719 | CXCXXL& CXXXQV | 64.464( | 30.384, | 8.659, | 16.442), | 18, | 17, | 17, | 9613, | 9616 |
| 731 | PNXDXW& WXKF | 109.800( | 38.575, | 8.116, | 63.108), | 14, | 8, | 8, | 9620, | 9625 |
| 733 | FXXXYXP&PTXXXXY&YXPT | 97.962( | 36.253, | 8.904, | 2.800), | 24, | 24, | 24, | 9609, | 9609 |
| 790 | CQS& RCF | 120.311( | 50.816, | 9.238, | 63.258), | 43, | 38, | 38, | 9590, | 9595 |
| 791 | PRW& HFXW | 63.134( | 29.932, | 8.466, | 2.736), | 13, | 13, | 13, | 9620, | 9620 |
| 796 | CXGY& CXXYC | 69.537( | 31.932, | 8.659, | 28.946), | 19, | 17, | 17, | 9614, | 9616 |
| 797 | SWXXP& YXXLXC | 60.306( | 25.253, | 8.616, | 26.436), | 17, | 16, | 16, | 9616, | 9617 |
| 802 | CXXXXTC&LST& STC | 86.923( | 37.575, | 8.569, | 40.776), | 18, | 15, | 15, | 9615, | 9618 |
| 809 | HHXXXP& HSXXXXXXXY | 98.844( | 49.686, | 8.408, | 40.749), | 15, | 12, | 12, | 9618, | 9621 |
| 809 | PXXPG& NXXTR& YXNXXXR | 174.417( | 43.180, | 8.992, | 124.244), | 30, | 21, | 27, | 9597, | 9606 |
| 812 | CCXXXC& LCG | 81.207( | 56.330, | 8.466, | 16.411), | 11, | 13, | 13, | 9619, | 9620 |
| 813 | GWXD& WMXF | 100.660( | 34.253, | 9.686, | 56.960), | 64, | 62, | 66, | 9563, | 9565 |
| 840 | CXXXXCN&CCXYXXXN | 71.215( | 29.932, | 8.408, | 32.876), | 13, | 11, | 12, | 9619, | 9621 |
| 833 | CXXXCXK&CXCXXL | 188.469( | 40.866, | 9.951, | 137.670), | 109, | 98, | 100, | 9522, | 9533 |
| 851 | KSC& VXXCR | 152.972( | 39.385, | 8.659, | 104.930), | 36, | 17, | 17, | 9607, | 9619 |
| 872 | AWXXV& CAW | 57.903( | 33.253, | 8.277, | 16.373), | 11, | 10, | 10, | 9622, | 9623 |
| 886 | DXXXVXC | 80.175( | 33.253, | 9.293, | 46.629), | 39, | 37, | 41, | 9590, | 9592 |
| 892 | CNYXXFXP&FXPXXXXXXW&CUXXXXXP | 692.147( | 30.095, | 10.871, | 661.181), | 266, | 237, | 279, | 9223, | 9254 |
| 893 | CXVXH& VXHXXXXP | 289.786( | 64.079, | 9.477, | 217.203), | 74, | 53, | 53, | 9559, | 9580 |
| 902 | PXTXXXF&HGXXV& PXTXXXXXXF | 167.948( | 30.646, | 9.924, | 117.478), | 69, | 67, | 99, | 9532, | 9534 |
| 905 | MQXF& MQI | 703.061( | 59.918, | 10.678, | 632.268), | 456, | 367, | 343, | 9176, | 9250 |
| | MXIF& MQI | | | | | | | | | |
| 908 | AXKK& KKA& KXXSXXK&RXGXS | 68.698( | 31.932, | 8.992, | 27.774), | 16, | 16, | 27, | 9606, | 9606 |
| 909 | AGXXF& FXVG& LXFP | 104.297( | 70.314, | 8.877, | 25.106), | 23, | 21, | 23, | 9609, | 9610 |
| 910 | FXND& NXFXXD& VTK | 69.238( | 51.879, | 9.114, | 8.241), | 31, | 31, | 33, | 9601, | 9601 |
| 911 | AIH& AQD& PFQ | 62.681( | 50.879, | 8.992, | 2.809), | 27, | 27, | 27, | 9606, | 9606 |
| | DFK& FQR& PFQ | | | | | | | | | |
| | DFK& DTN& FQR | | | | | | | | | |
| | AQD& DFK& IAQ | | | | | | | | | |
| | AIH& AQD& FQR | | | | | | | | | |
| | AIH& FQR& QDF | | | | | | | | | |
| 912 | IRR& KRH& VKR | 56.118( | 44.557, | 8.776, | 2.785), | 20, | 20, | 20, | 9613, | 9613 |
| 917 | PXRR& RPXXXR& SRXXR | 69.032( | 44.557, | 8.700, | 2.774), | 18, | 18, | 18, | 9615, | 9615 |
| 927 | GFXXF& HNP& RXPXT | 116.476( | 54.201, | 9.238, | 53.037), | 21, | 20, | 36, | 9594, | 9595 |
| 948 | GKK& KKXXA& DAXXXG | 86.517( | 51.201, | 8.408, | 24.908), | 10, | 9, | 12, | 9620, | 9621 |
| 952 | DXXPXP& FNXT& VTP | 135.949( | 53.008, | 8.992, | 53.949), | 13, | 11, | 27, | 9604, | 9606 |
| 954 | GXRXX& PXXAXR& PNS | 65.984( | 50.879, | 8.408, | 6.696), | 11, | 11, | 12, | 9621, | 9621 |
| 957 | CXXXGR& RCXXXG | 63.466( | 52.201, | 8.520, | 2.746), | 14, | 14, | 14, | 9619, | 9619 |
| 963 | EFXXT& MXGH | 91.684( | 38.575, | 9.463, | 43.646), | 13, | 13, | 32, | 9581, | 9581 |
| 976 | EQXXM& MNP | 72.881( | 35.253, | 8.700, | 28.927), | 15, | 14, | 18, | 9614, | 9615 |
| 979 | EXXXFP& FDXXG& IGXXV | 74.898( | 33.253, | 8.520, | 33.125), | 15, | 13, | 14, | 9617, | 9619 |
| 984 | EFD& GLK& VIK | 92.478( | 55.523, | 8.116, | 28.839), | 10, | 8, | 8, | 9633, | 9635 |
| | | 66.637( | 44.557, | 8.520, | 13.560), | 10, | 10, | 14, | 9619, | 9619 |

| PC | StochasticMotif | $\ell(T)$( | $\ell(M)$, | $\ell(P)$, | $\ell(L)$), | $E$, | $N_1^+$, | $N_1$, | $N_2^+$, | $N_2$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 987 | GAA& FXLP& MGP | 106.943( | 46.557, | 8.277, | 52.109), | 14, | 10, | 10, | 9619, | 9623 |
| 999 | ADN& NAXV& NXGA | 99.062( | 48.557, | 8.739, | 41.766), | 17, | 13, | 19, | 9612, | 9614 |
| 1015 | FXVXV& KXXRXG | 117.105( | 37.575, | 9.375, | 70.154), | 11, | 8, | 46, | 9584, | 9587 |
| 1049 | GPXXXR& GXMG& MGP | 75.877( | 50.879, | 8.569, | 16.429), | 16, | 13, | 15, | 9617, | 9618 |
| 1057 | GXXXXXFXR&LXVK&RXYXXP | 161.615( | 57.271, | 9.516, | 94.828), | 64, | 56, | 56, | 9569, | 9577 |
| 1058 | YPXXXXW&YRXXQ | 49.902( | 36.383, | 8.739, | 2.780), | 19, | 19, | 19, | 9614, | 9614 |
| 1060 | EIXXY& LEXE& IXXYR | 119.473( | 53.201, | 9.137, | 57.135), | 32, | 29, | 33, | 9597, | 9600 |
| 1072 | FXXWXP& PXPXXH | 81.457( | 38.575, | 8.908, | 33.974), | 25, | 23, | 24, | 9607, | 9609 |
| 1075 | MPXT& RDXT | 45.691( | 33.932, | 9.137, | 2.623), | 33, | 33, | 33, | 9600, | 9600 |
| 1079 | PXXXDXXXG&DXXXGXI&DXXXDXX1 | 355.502( | 63.079, | 9.666, | 282.757), | 66, | 63, | 69, | 9541, | 9564 |
| 1087 | EQL& QLXP | 140.231( | 31.932, | 9.959, | 98.340), | 14, | 10, | 104, | 9525, | 9529 |
| 1110 | NPXXY& WXPY | 103.798( | 35.253, | 8.346, | 60.198), | 12, | 8, | 11, | 9618, | 9622 |
| 1116 | WND& WXDXXC | 119.043( | 34.253, | 8.739, | 76.050), | 20, | 15, | 19, | 9609, | 9614 |
| 1120 | FXXQXG& WXXXAI | 76.313( | 38.575, | 8.776, | 28.961), | 22, | 20, | 20, | 9611, | 9613 |
| 1148 | DXXXXGXW&TDY | 168.462( | 36.324, | 9.068, | 123.070), | 21, | 13, | 30, | 9593, | 9603 |
| 1155 | CXXCK& GXHY | 50.720( | 35.253, | 8.520, | 6.947), | 13, | 13, | 14, | 9619, | 9619 |
| 1170 | CXSXC& GCXC | 89.548( | 35.253, | 8.908, | 45.387), | 21, | 19, | 24, | 9607, | 9609 |
| 1191 | RRM& WFQ | | | | | | | | | |
| | KEP& WFQ | | | | | | | | | |
| | RMK& WFQ | 63.089( | 29.932, | 8.845, | 23.312), | 10, | 10, | 22, | 9611, | 9611 |
| 1211 | EEXQ& GDXXXP | 133.891( | 37.061, | 9.589, | 87.241), | 17, | 14, | 62, | 9568, | 9571 |
| 1214 | FNQ& PFN& QLXQ | | | | | | | | | |
| | AYXQ& FNQ& GGA | | | | | | | | | |
| | AAY& FNQ& PFXQ | 57.691( | 46.557, | 8.408, | 2.726), | 12, | 12, | 12, | 9621, | 9621 |
| 1222 | SWXF& WVXXXXS | 74.666( | 37.061, | 8.659, | 28.946), | 19, | 17, | 17, | 9614, | 9616 |
| 1237 | KQP& MXKC | 43.065( | 31.932, | 8.408, | 2.725), | 12, | 12, | 12, | 9621, | 9621 |
| 1266 | GXXQR& SGXXXQ& DEP | 128.421( | 52.201, | 8.616, | 67.605), | 20, | 15, | 16, | 9612, | 9617 |
| 1341 | FHM& WFT | | | | | | | | | |
| | FHM& IYP | 41.197( | 29.932, | 8.520, | 2.746), | 14, | 14, | 14, | 9619, | 9619 |
| 1361 | GXFP& FXXXXWP | 48.263( | 37.061, | 8.466, | 2.736), | 13, | 13, | 13, | 9620, | 9620 |
| 1415 | GXXDXG& RVD& PGH | 60.307( | 48.879, | 8.659, | 2.769), | 17, | 17, | 17, | 9616, | 9616 |
| 1674 | HXXKXXQ&HXXNA | 49.441( | 38.383, | 8.346, | 2.713), | 11, | 11, | 11, | 9622, | 9622 |
| 1676 | HXDK& HPXXG | 73.323( | 35.253, | 9.068, | 29.002), | 10, | 10, | 30, | 9603, | 9603 |
| 1680 | MWE& PLC | 41.068( | 29.932, | 8.408, | 2.725), | 12, | 12, | 12, | 9621, | 9621 |
| 1681 | DXXFP& QHXXXXW | 56.476( | 38.383, | 8.466, | 9.628), | 11, | 11, | 13, | 9620, | 9620 |
| 1682 | GEXW& YXRH | 45.532( | 33.932, | 8.811, | 2.789), | 21, | 21, | 21, | 9612, | 9612 |
| 1683 | GXXXRXG&YXXCK | 75.557( | 38.383, | 8.877, | 28.078), | 24, | 20, | 23, | 9609, | 9610 |
| 1687 | QWA& WKXXF | 44.824( | 33.253, | 8.811, | 2.789), | 21, | 21, | 21, | 9612, | 9612 |
| 1688 | PXXXXKK&WXXXRXL | 84.573( | 40.190, | 8.965, | 35.418), | 21, | 20, | 26, | 9606, | 9607 |
| 1691 | YXXLXL& YPXXXL& VPXXXXL | 85.052( | 38.530, | 8.659, | 38.063), | 10, | 10, | 17, | 9616, | 9616 |
| 1705 | GXCXXC& LXXXEK& RCXXC | 98.012( | 36.523, | 9.138, | 52.331), | 23, | 23, | 34, | 9599, | 9599 |
| 1707 | CPXC& LXCXE | 122.703( | 35.253, | 9.016, | 78.431), | 20, | 16, | 26, | 9601, | 9605 |
| 1712 | KAI& NXXFP& PXX1G | 62.335( | 51.201, | 8.408, | 2.725), | 12, | 12, | 12, | 9621, | 9621 |
| 1717 | PPXXXP& TPW | 57.787( | 34.253, | 8.659, | 14.874), | 13, | 13, | 17, | 9616, | 9616 |
| 1718 | CPXXC& PXWXXL | 49.176( | 37.576, | 8.811, | 2.789), | 21, | 21, | 21, | 9612, | 9612 |
| 1719 | FPP& PXDP& SPR | 57.841( | 46.557, | 8.569, | 2.754), | 15, | 15, | 15, | 9618, | 9618 |
| 1721 | CRH& WEE | 46.224( | 29.932, | 8.466, | 6.827), | 12, | 12, | 13, | 9620, | 9620 |
| 1779 | CXXXCV& CXXXGXP | 70.832( | 39.383, | 8.408, | 23.041), | 11, | 10, | 12, | 9620, | 9621 |
| 1838 | SPF& WGXXQ | 44.728( | 33.253, | 8.700, | 2.774), | 18, | 18, | 18, | 9615, | 9615 |
| 1916 | CXCXXXH&GCXF | 61.617( | 37.061, | 8.201, | 16.355), | 10, | 9, | 9, | 9623, | 9624 |
| 1917 | HXSXXY& LWR | 82.483( | 34.253, | 8.346, | 49.883), | 10, | 7, | 11, | 9619, | 9622 |
| 2052 | CXYS& YQI | | | | | | | | | |
| | GWXW& KFH | 43.463( | 31.932, | 8.776, | 2.785), | 20, | 20, | 20, | 9613, | 9613 |
| 2054 | GWVE& QYQ | 40.906( | 29.932, | 8.277, | 2.698), | 10, | 10, | 10, | 9623, | 9623 |
| 2056 | DXGW& GHXXC | 72.991( | 35.253, | 8.776, | 28.961), | 22, | 20, | 20, | 9611, | 9613 |
| 2059 | CXTXY& WGN | 44.226( | 33.253, | 8.277, | 2.698), | 10, | 10, | 10, | 9623, | 9623 |
| 2065 | MGQ& YIXXW | 49.302( | 33.253, | 8.700, | 7.348), | 17, | 17, | 16, | 9615, | 9615 |
| 2069 | GXXGH& NCXXXXH | 88.040( | 38.383, | 8.845, | 40.812), | 25, | 22, | 22, | 9608, | 9611 |
| 2074 | NXPXXXXK&RXXXXRXXNE:YXDD | 163.907( | 39.856, | 9.219, | 94.832), | 45, | 37, | 37, | 9586, | 9596 |
| 2078 | RPC& WGC | 41.493( | 29.932, | 8.776, | 2.785), | 20, | 20, | 20, | 9613, | 9613 |
| 2082 | CCP& GGXC | 117.582( | 31.932, | 9.137, | 76.514), | 35, | 30, | 33, | 9595, | 9609 |
| 2088 | RXXXWXS&YWXL | 55.446( | 37.061, | 8.520, | 9.867), | 12, | 12, | 14, | 9619, | 9619 |
| 2092 | UXSH& PXRXM& VPL | 61.307( | 49.879, | 8.659, | 2.769), | 17, | 17, | 17, | 9616, | 9616 |
| 2097 | CPXXK& HCXXC | 61.552( | 36.576, | 8.466, | 16.511), | 14, | 13, | 13, | 9619, | 9620 |
| 2098 | RBW& RXXRXXW | 59.478( | 35.061, | 6.616, | 15.802), | 11, | 11, | 16, | 9617, | 9617 |
| 2111 | WXXNP& WXXWP | 47.634( | 36.575, | 8.346, | 2.713), | 11, | 11, | 11, | 9622, | 9622 |
| 2112 | CXXYXXY&FNXXXL | 54.649( | 39.383, | 8.520, | 6.947), | 13, | 13, | 14, | 9619, | 9619 |
| 2122 | SXXMG& YAM | 51.640( | 33.253, | 8.520, | 9.867), | 12, | 12, | 14, | 9619, | 9619 |
| 2135 | NCXXXXC&QXTA | 63.087( | 37.061, | 8.908, | 17.119), | 29, | 20, | 24, | 9609, | 9609 |
| 2140 | CXPXM& FMV | 104.836( | 33.253, | 8.408, | 63.173), | 17, | 12, | 12, | 9616, | 9621 |
| 2146 | FXXXXGF&SXQXXI& GXXGXXW | 163.648( | 59.137, | 9.589, | 114.921), | 72, | 62, | 62, | 9561, | 9571 |
| 2148 | NFXXG& QXXXMXC | 30.184( | 38.383, | 8.992, | 2.809), | 27, | 27, | 27, | 9606, | 9606 |
| 2149 | CAXC& QMXXXXN | 48.036( | 37.061, | 8.277, | 2.698), | 10, | 10, | 10, | 9623, | 9623 |
| 2181 | HXVXXXL&LHXXXW | 57.156( | 39.383, | 8.408, | 9.366), | 10, | 10, | 12, | 9621, | 9621 |
| 2183 | CXXWH& DRL | | | | | | | | | |
| | CXXWH& SPQ | 48.892( | 33.253, | 8.845, | 2.793), | 22, | 22, | 22, | 9611, | 9611 |
| 2185 | MXKM& RMXK& TQXE | 62.125( | 50.557, | 8.845, | 2.793), | 22, | 22, | 22, | 9611, | 9611 |
| 2201 | WKXM& YIN | | | | | | | | | |
| | GXVQ& WKF | | | | | | | | | |
| | PYQ& WXRM | | | | | | | | | |
| | CXEY& PYQ | 43.790( | 31.932, | 9.044, | 2.814), | 29, | 29, | 29, | 9604, | 9604 |
| 2202 | FXPXXXW&NGN | 46.036( | 35.061, | 8.277, | 2.698), | 10, | 10, | 10, | 9623, | 9623 |
| 2204 | NWK& WWQ | | | | | | | | | |
| | TLC& WWQ | | | | | | | | | |
| | NYQ& WWQ | | | | | | | | | |
| | CDY& NEW | 41.672( | 29.932, | 8.937, | 2.804), | 25, | 25, | 25, | 9608, | 9608 |
| 2212 | CKY& KYM | 40.906( | 29.932, | 8.277, | 2.698), | 10, | 10, | 10, | 9623, | 9623 |
| 2253 | DTR& NRX1& RFP | 57.552( | 46.557, | 8.277, | 2.698), | 10, | 10, | 10, | 9623, | 9623 |