A genetic algorithm for scaled-based translocon simulation

Sami Laroum[1], Béatrice Duval[1], Dominique Tessier[2], and Jin-Kao Hao[1]

[1] LERIA, 2 Boulevard Lavoisier, 49045 Angers, France
[2] UR 1268 Biopolymères Interactions Assemblages,
INRA, 44300 Nantes, France
{laroum,bd,hao}@info.univ-angers.fr
{tessier}@nantes.inra.fr

Abstract. Discriminating between secreted and membrane proteins is a challenging task. A recent and important discovery to understand the machinery responsible of the insertion of membrane proteins was the results of Hessa experiments [9]. The authors developed a model system for measuring the ability of insertion of engineered hydrophobic amino acid segments in the membrane. The main results of these experiments are summarized in a new "biological hydrophobicity scale". In this scale, each amino acid is represented by a curve that indicates its contribution to the process of protein insertion according to its position inside the membrane. We follow the same hypothesis as Hessa but we propose to determine "in silico" the hydrophobicity scale. This goal is formalized as an optimization problem, where we try to define a set of curves that gives the best discrimination between signal peptide and protein segments which cross the membrane. This paper describes the genetic algorithm that we developed to solve this problem and the experiments that we conducted to assess its performance.

Keywords: Membrane Proteins, Classification, Optimization, Genetic Algorithm.

1 Introduction

Membrane proteins play an important role in many processes in living cells, and they are the targets of many pharmaceutical developments. In fact, 50% of these proteins are used in human and veterinarian medicine [4]. Despite their number and their importance, membrane proteins with known three-dimensional structures represent only 2% of the protein data bank (PDB) [2]. It is difficult to determine their structure because they are difficult to express and crystallize [13]. The great importance of these proteins promoted their study and particularly the search of the machinery responsible of addressing these proteins towards the membrane [23].

The proteins transported across the endoplasmic reticulum (ER) membrane include soluble proteins and membrane proteins. Soluble proteins completely

cross the membrane and usually include a short N-terminal segment[3], called Signal Peptide (SP), that will be cleaved after transport. Membrane proteins have one or several segments that get inserted into the membrane, called transmembrane (TM) segments. Both types of proteins use the same machinery for transport across the ER membrane, a protein complex located in the ER membrane called the translocon. The translocon channel allows the soluble proteins to cross the membrane and permits the hydrophobic TM segment of membrane proteins to fit in the membrane. SP and TM segments have very close biochemical properties, and particularly they both contain a hydrophobic region. Nevertheless, a TM segment possess the "key" to open sideways the translocon, which permits the insertion of the protein in the membrane.

Bioinformatics methods suggest some solutions to deal with the recognition of membrane proteins. One of the first prediction method was proposed by Kyte and Doolittle [16]. This method was based on an experimentally determined hydrophobicity index where each amino acid was given a score based on its preference to water or lipid. A hydrophobicity plot was performed by summing the hydrophobicity index over a window of a fixed length and values superior to a cutoff threshold indicates possible TM segments. However, this method performed poorly and is outperformed by machine learning algorithms. More recent works propose methods based on hidden Markov models such as TMHMM [15] and HMMTOP [21], on artificial neural networks such as PHDhtm [20] and Memsat [10]. Other methods combine the prediction of TM segments and SP such as Phobius [14], Philius [19], and SPOCTOPUS [22]. These methods give good discrimination performances, but it is difficult to link their results to a biological interpretation of the translocon machinery. Furthermore, they still sometimes confuse a SP and a TM segment. This is particulary true for the first TM segment of a membrane protein, that is located in the N-terminal region.

In 2005, Hessa *et al.* carried out a series of in vitro experiments with the aim to compute the energy required for the insertion of a designed TM segment in the membrane [8]. Unlike Kyte and Doolittle which attribute to each amino acid a single hydrophobicity index, Hessa *et al.* determine for each amino acid a contribution profile - the potential of insertion - according to its position in the segment. As a result, Hessa *et al.* suggested a 'biological hydrophobicity scale' [9] where each amino acid is represented by a curve. The experiments leading to these curves are very complex to realize, and the predictive system issued from this work such as SCAMPI [3] was only designed to predict TM segments. In fact, SCAMPI does not offer a good distinction between SP and TM segments.

In our work, we follow the same hypothesis as Hessa *et al.* and we assume that we can elaborate "in silico" a new scale for the amino acids, by studying two sets of protein segments which cross the translocon and share the same chemical hydrophobic profile: SP and TM segments. This scale could benefit from a large quantity of data stored in the protein databases and consequently could be more

[3] This paper only considers the primary structure of a protein, represented by a sequence of amino acids. One extremity of this sequence (the first synthesized by the ribosome) is called N-terminal.

precise than the scales derived by in vitro experiments. This paper introduces a genetic algorithm to optimize this scale. As suggested by Hessa *et al.*, an amino acid may have different hydrophobic indexes according to its position inside the translocon channel, and consequently we represent its hydrophobic profile by a symmetric curve. However, we shall see that for some amino acids the profile can be represented by a straight line.

The remainder of this paper is organized as follows. In section 2, we describe our formalization of this problem and the genetic algorithm designed to optimize the curves. Section 3 presents the learning dataset that we have built, and the validation protocol while Section 4 reports the experimental results. Finally conclusions are provided in section 5.

2 Optimization of a hydrophobicity scale

2.1 Overview of our approach

Figure 1 summarizes the approach that we propose to determine the amino acid insertion profiles (curves of amino acids). Our study relies on a learning dataset composed of two types of sequences: SP (class SP) and TM segments (class TM). We define a simple classifier inspired from the work of Kyte and Doolittle. This classifier slides a window to compute an insertion score for each sequence in order to correctly recognize the SP and the TM segments of the learning dataset. The classifier is determined by 20 curves that represent the insertion profiles of the 20 amino acids. To obtain the best discrimination between SP and TM, a genetic algorithm is executed in order to optimize the set of 20 curves. In the following of this section, we describe more precisely each component of this approach.

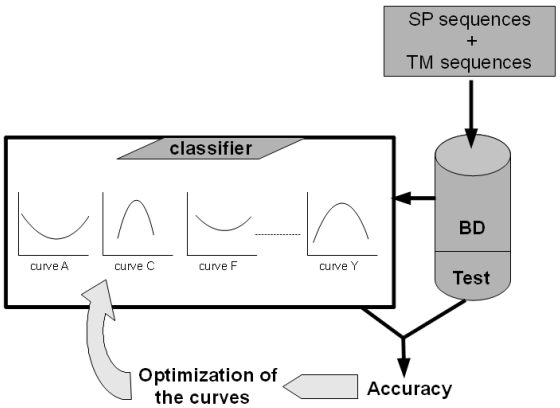

Fig. 1. In silico determination of a hydrophobicity scale.

2.2 Discrimination function: a sliding window classifier

If a denotes one of the 20 amino acids, we note $C[a]$ the curve associated to a in the scale. The curve $C[a]$ gives the value of the hydrophobicity index of the amino acid a depending on its position during the process of protein insertion in the membrane. As the membrane thickness is about 20 amino acids, the curves are defined on a window of length l with $l \simeq 20$. An appropriate window length l will be determined experimentally as explained in section 3.3.

For $j \in [1, l]$, $C[a, j] = C[a](j)$ denotes the index of the amino acid a when it is in position j in the window. For a sequence Seq of amino acids of length l, we use the notation $Seq = < a_1 a_2 \dots a_l >$ and we define the insertion average of this sequence as the average of its indexes :

$$E(Seq) = \frac{\sum_{j=1}^{l} C[a^j, j]}{l} \tag{1}$$

In the case of a longer sequence $Seq = < a_1 a_2 \dots a_n >$ of length $n > l$, a sliding window of fixed length l is scanned on the sequence and we define the insertion index of this sequence as the maximum average calculated on a sub-sequence of length l:

$$E_{max}(Seq) = \max_{1 \leq k \leq n-l+1} \{E(Seq_k)\} \tag{2}$$

where $Seq_k = < a_k a_{k+1} \dots a_{k+l-1} >$.

Classifier SP/TM: The distinction between the class SP and the class TM is given by the insertion index $E_{max}(Seq)$ and a threshold τ. $E_{max}(Seq)$ corresponds to the maximum value of hydrophobicity of the sequence, whereas the threshold τ represents a value separating the two classes SP and TM.

A segment TM is generally hydrophobic [16] and therefore our classification rule is:

$$\begin{cases} Seq \in class\ SP & \text{if} \quad E_{max}(Seq) < \tau \\ Seq \in class\ TM & \text{otherwise} \end{cases}$$

The set of curves and the threshold τ determine our classifier. Its quality is evaluated by the accuracy and the area under the ROC curve (AUC), which measure the ability of the curves to discriminate between SP sequences and TM sequences.

2.3 Curves encoding

According to the results of [9] that suggest symmetric insertion profiles, we represent each curve by a parabola defined by an equation $H = \alpha(x - X_0)^2 + \beta$, and we determine a curve by the pair of parameters $(H_{extremity}, H_{middle})$ (figure 2 (A)). $H_{extremity}$ is the value of the curve at the extremities of the window of length l ($X_{min} = 1$ and $X_{max} = l$), whereas H_{middle} is the value of the curve at the middle of the window.

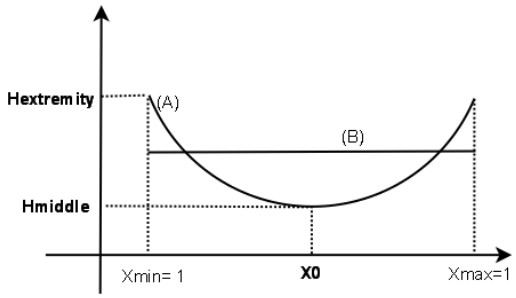

Fig. 2. Representation of an insertion curve.

To modify the curve we change $(H_{extremity}, H_{middle})$. The modification of $H_{extremity}$ means that we can act on the behavior of the amino acid at the interfaces of the membrane, while changing H_{middle} allows us to act on the behavior of the amino acid in the middle of the membrane. Note that the manipulated curves can be straight lines (figure 2 (B)) defined by $H_{extremity} = H_{middle}$ ($\alpha = 0$ for the equation of the parabola).

2.4 A genetic algorithm for optimizing the curves

To optimize the amino acid curves, we use a genetic algorithm (GA). Our algorithm follows the classic schema of a genetic algorithm by evolving a population of individuals. Each individual in our population is a set of 20 curves and each curve is coded by a couple $(H_{extremity}, H_{middle})$. The algorithm begins with an initial population generated by random modifications on a known hydrophobicity scale. The population evolves through the generations by the application of crossover and mutation operators that act directly on the curves of the individuals. An elitism mechanism is used to keep the best individuals of a population. This process is repeated until a predefined number of generations is reached.

Initial population: The litterature proposes several hydrophobicity scales where each amino acid is assigned a constant index. In our GA algorithm, we decide to generate an initial population by modifications applied on the Eisenberg hydrophobicity scale [5].

Each individual in our initial population is a set of 20 straight lines. To build such an individual, we first initiate a random number k between 1 and 20, which represents the number of amino acid indexes that will be modified. We randomly choose these k amino acids, and we change the value H_{middle} of these k amino acids by the addition of a real value Δ_{mid} randomly selected between $[-3, 3]$ (interval determined experimentally). This process is repeated to generate the required number n of individuals in the initial population. The population size is fixed at $n = 100$ in this work.

Fitness function: The purpose of the GA algorithm is to optimize a set of 20 curves, in order to correctly discriminate SP sequences from TM sequences. Therefore, the fitness function is given by the classification accuracy measured on the learning dataset.

Evolution: The individuals of the current population P are sorted according to the fitness function. The 10% best individuals of P are directly copied to the next population P' and removed from P. The remaining 90% individuals are then generated by using crossover between two parents selected from the current population by following the principle of wheel selection [7].

Crossover operator: Our crossover operator considers two parents and generates two children, by exchanging a certain number of curves between two parents (figure 3 (A)).

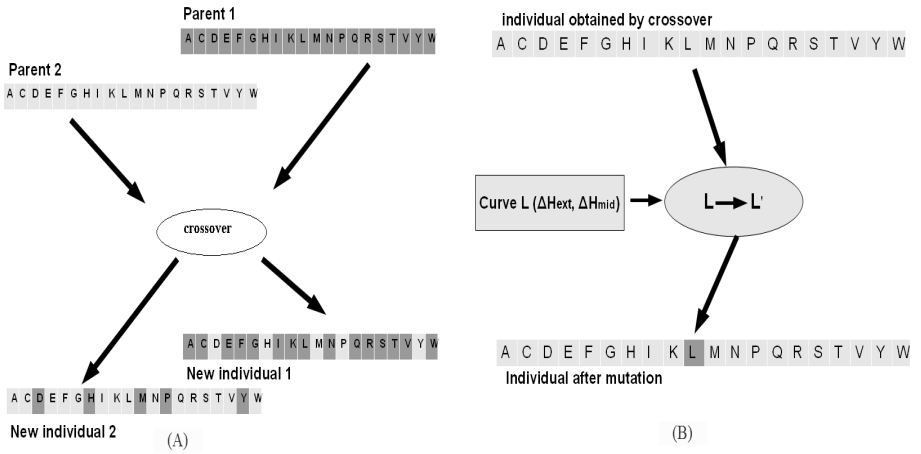

Fig. 3. Figure A shows the crossover operator between two parents where the curves of the amino acids aspartic (D), histidine (H), méthionine (M), proline (P), tyrosine (Y) are exchanged. Figure B shows the mutation operator modifying the curve of the amino acid leucine (L).

To generate the new individuals, we perform the following steps :

1. Randomly select in [1,15] (interval determined experimentally) the number M of amino acids that will be modified between the two parents.
2. Randomly choose the M amino acids which are going to be exchanged.
3. Exchange the curves of the amino acids of the the first parent with the curves of the same amino acids of the second parent.

Mutation operator: The mutation operator is designed to enrich the diversity of the population by manipulating the structure of individual. In our

mutation operator we apply a local modification (figure 3 (B)) to a given individual by performing the following steps:

1. Randomly select in [1,12] (interval determined experimentally) the number N of amino acids that will be modified.
2. Randomly choose the N amino acids which are going to be modified.
3. Modify the parameters $(H_{extremity}, H_{middle})$ of the curve by adding a couple $(\Delta_{Hext}, \Delta_{Hmid})$ of real values randomly selected between $[-2, 2]$. For a straight line, we modify only the parameter H_{middle} by adding a Δ_{Hmid}.

The mutation operator is applied every 10 generations on 90% of the population.

Strategies of optimization of the curves: In a previous work [17], we presented statistics of the amino acid frequencies in our datasets. We have observed that the amino acids Alanine (A), Phenylalanine (F), Isoleucine (I), Leucine (L), Glycine (G), Serine (S) and Valine (V) are the most frequent ones. For the other amino acids, the learning dataset provides few information and therefore the precise definition of their hydrophobicity curves relies on insufficient support. So, we propose to represent the insertion index of these amino acids by a straight line, for which the algorithm has to determine only one parameter. Conversely, for the 7 frequent amino acids, the algorithm has to determine a symmetric curve.

To explore the search space, our GA algorithm operates in two stages. In the first stage, the hydrophobicity indexes are represented by straight lines for all the amino acids, even the frequent ones and the algorithm has to determine the optimal values in this search space. The purpose is to position the sliding window and at the same time to determine the best individual (solution S_1) to discriminate between SP and TM segment. In the second stage, the algorithm fixes the values of the amino acids which are not frequent to the values of the solution S_1 and optimizes symmetric curves for the frequent amino acids by applying specific operators.

Specific operators: The specific operators are very similar to the precedent operators but they only modify the curves of the frequent amino acids. Thus, the specific crossover operator exchanges the curves of the frequent amino acids between two parents. It chooses randomly M amino acids ($M \leq 7$) among the frequent amino acids and exchanges their curves. As well, the specific mutation modifies the curves of the frequent amino acids. It chooses randomly N amino acids ($N \leq 7$) among the frequent amino acids and modifies their curves by adding a real value Δ_{mid} .

These specific operators allow us to limit the search space by limiting the number of parameters that must be optimized, only the curves of seven amino acids are optimized. The learning dataset provides more information for these amino acids and for this reason, it seems natural to concentrate our search on the frequent amino acids. The more information we have about the amino acids, the more accurately the algorithm can optimize their curves.

3 Experiments and results

3.1 Learning dataset: SWP

Our approach requires a dataset containing SP and first TM segments. So, we built a data set, called "SWP" by extracting from the UniprotKB/Swiss-Prot database [12] 684 proteins with TM segments and the same number of proteins with SP. In the case of a soluble protein, the sequence stored in SWP corresponds to the signal peptide (SP). We represent it by the first 35 amino acids of the protein because the length of SP for eucaryotic proteins ranges from 22 to 32 amino acids [1]. In the case of a membrane protein, the sequence stored in SWP corresponds to the first transmembrane segment as annotated in the database. As the annotation of the proteins in SwissProt is the result of TM prediction programs such as TMHMM [15] and MEMSAT [11], we consider that the TM segments in SwissProt are not precisely located on the sequence. So, we add to the TM segment representation the 10 adjacent amino acids before and after the annotated position. To summarize, in our dataset SWP, asecreted protein is represented by a SP which corresponds to the first 35 amino acids, while a membrane protein is represented by its first TM segment with the 10 adjacent amino acids before and after the annotated position.

Note that the constructed dataset relies on the lastest version of the Uniprot-KB/Swiss-Prot database and overlaps the datasets used by other methods. It seems unfair to learn on our dataset and test on other sets of proteins. So, we test all the methods on our dataset using the validation protocol described below.

3.2 Validation protocol

To assess the performance of our method, we perform a 10-fold cross-validation on the dataset SWP. The initial dataset SWP is split into $K = 10$ subsets of the same size. The method builds a classifier with $(K-1)$ subsets as training set and estimates the error on the remaining subset (test set). We repeat K times the same process by varying the subset that plays the role of test set. The accuracy estimated by K-fold cross-validation is then the average of the accuracies of these K experiments.

3.3 Experimental results

The purpose of this experiment is 1) to evaluate the influence of the length of the sliding window of the classifier and 2) to optimize the values of the curves for each amino acid. As explained before, the membrane length is about 20 amino acid positions. Hessa *et al.* optimize a profile contribution for each amino acid on a window of 19 amino acids which means that the length of the curves is 19 amino acids. However, the statistical distribution of TM segments in proteins with known 3D structure shows that most TM segments have a length ranging between 21 and 30 amino acids [18]. So, in this experiment we assess a window with 19, 21, and 23 amino acids.

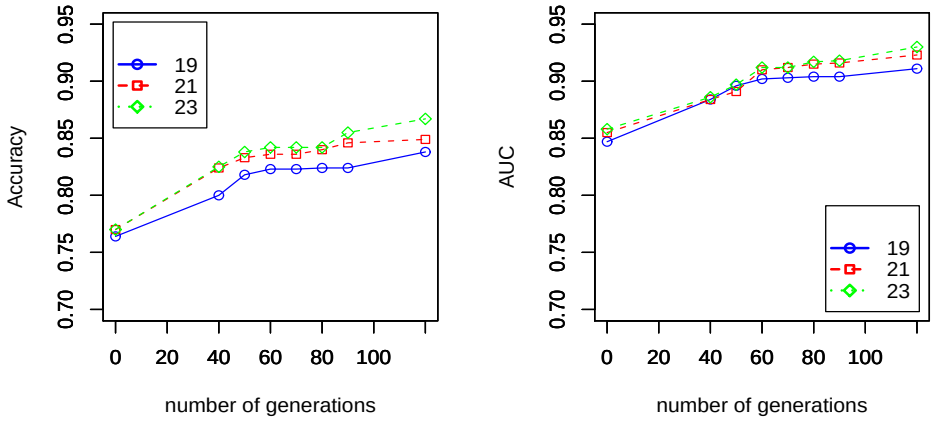

Fig. 4. Performance comparison with different window lengths. The figure shows the average accuracy and average AUC of the best individual of the population.

We run our GA according to the protocol described in section 2.4. In figure 4, the X axis represents the number of generations, while Y axis represents the average accuracy (left figure) and the AUC (right figure) of the best individual of a population. The best results of the accuracy and the AUC are obtained when we use a window with 23 amino acids (green line). This means that the curves with 23 amino acids are better suited to discriminate between signal peptides and TM segments. The figure also displays the performance evolution through the generations. From the generation 1 to the generation 80, our algorithm is in its first stage and it optimizes a straight line for each amino acid. The second stage starts from the optimum solution obtained in the first step to optimize the curves of the frequent amino acids. We observe that this second stage, which only modifies the curves of 7 frequent amino acids, still improves the performance. We can also notice that several runs of our GA give similar results.

This evolution process ends when a predefined number of generations is reached. We do not use any system to avoid overfitting in the training phase.

Table 1 summarizes a comparison with other algorithms. We compare our algorithm with Kyte & Doolittle (KD), Eisenberg (EIS), Engelman (GES) [6] scales and two of the best methods for the discrimination between signal peptides and TM segments: Phobius and Philius. The first method, Phobius, is based on Hidden Markov model and the second method, Philius uses Dynamic Bayesian Networks for its prediction. We also compare the GA algorithm with a method that we developed previously, MN-LS [17] which uses a local search approach for determining the curves of the amino acids. Each prediction method is applied on each fold and then we calculate the average accuracy evaluated according

to the same process of 10-fold cross-validation, and table 1 reports the average accuracy with the standard deviation.

Method	KD	EIS	GES	Phobius	Philius	MN-LS	GA algorithm
average accuracy	0.765	0.755	0.768	0.855	0.842	0.853	0.867
Standard deviation	0.031	0.029	0.0035	0.035	0.032	0.022	0.031

Table 1. Comparison of our GA and other methods. The table gives the average accuracy and the standard deviation obtained on SWP dataset.

The prediction methods based on the hydrophobicity scales slide a fixed length window along the sequence and use a cutoff value to decide if the sequence is a possible TM segment or SP. These methods, as well as MN-LS and GA method, only require as inputs SP and TM segments. For Phobius and Philius, both methods only accept the complete sequence in their web server. These methods are trained to predict SP sequences and TM segments using the complete protein sequence which allow them to take into account additional information like the different composition between cytoplasmic or reticulum exposed loops. MN-LS and GA are developed to optimize curves representing the potential contribution of each amino acid during the insertion of segments in the membrane and use only the SP or the TM segment.

A performance evaluation is presented table 1. We can observe that the hydrophobicity scales perform poorly on SWP dataset, while GA gives the best result. Our GA improved the values of the Eisenberg scale which we used to generate the initial population. However, Phobius and Philius obtain also good predictive performances, but it is easier to drive intuitively-simple reason related to the translocon mechanism for each prediction produced by MN-LS or GA.

Our previous MN-LS method uses a local search approach to determine the curves. As a result, the curves depend on the values of the initial solution, while the GA algorithm optimizes the curves by exploring a large search space in a diversified way and with a good discrimination. The standard deviation shows that the performances of the method are stable on the different runs of the 10-fold cross-validation.

4 Conclusion

In this work, we have presented a genetic algorithm to optimize the curves that represent the contribution of the 20 amino acids to the mechanism of insertion into the membrane. By using a simple sliding window classifier which computes an insertion score of sequences, we demonstrated that the GA algorithm is able to optimize a set of curves that discriminate between two classes of close sequences: signal peptides and TM segments. Despite the simplicity of the classifier, our approach provides classification performances that are equal to two of the best methods of the domain. Furthermore, our approach provides a clear biological

interpretation of the insertion phenomenon, which is not the case of sophisticated machine learning methods. Indeed, the curves which we optimize provide an explanation of the contribution of the amino acids during the insertion of the proteins in the membrane.

For future work, we want to introduce more knowledge about the phenomena of membrane proteins insertion to provide more effective guidance of the genetic algorithm.

5 Acknowledgments

This research was partially supported by the region Pays de la Loire (France) with its "Bioinformatics Program" (2007-2011) and Radapop Project (2009-2013). The authors are grateful to the reviewers for their useful comments.

References

1. J.D. Bendtsen, H. Nielsen, G. von Heijne, and S. Brunak. Improved prediction of signal peptides: SignalP 3.0. *Journal of Molecular Biology*, 340(4):783–795, 2004.
2. H.M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T.N. Bhat, H. Weissig, I.N. Shindyalov, and P.E. Bourne. The Protein Data Bank. *Nucleic Acids Research*, 28(1):235–242, 2000.
3. A. Bernsel, H. Viklund, J. Falk, E. Lindahl, G. von Heijne, and A. Elofsson. Prediction of membrane-protein topology from first principles. *Proceedings of the National Academy of Sciences of the Unites States of America*, 105(20):7177–7181, 2008.
4. J.M. Cuthbertson, D.A. Doyle, and M.S.P. Sansom. Transmembrane helix prediction: a comparative evaluation and analysis. *Protein Engineering Design and Selection*, 18(6):295–308, June 2005.
5. D. Eisenberg, R.M. Weiss, and T.C. Terwilliger. The helical hydrophobic moment: a measure of the amphiphilicity of a helix. *Nature*, 299(5881):371–374, 1982.
6. D. M. Engelman, T. A. Steitz, and A. Goldman. Identifying nonpolar transbilayer helices in amino acid sequences of membrane proteins. *Annual review of biophysics and biophysical chemistry*, 15:321–353, 1986.
7. D.E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley Professional, 1 edition, January 1989.
8. T. Hessa, H. Kim, K. Bihlmaier, C. Lundin, J. Boekel, H. Andersson, I. Nilsson, S.H. White, and G. von Heijne. Recognition of transmembrane helices by the endoplasmic reticulum translocon. *Nature*, 433(7024):377–381, 2005.
9. T. Hessa, N. M. Meindl-Beinker, A. Bernsel, H. Kim, Y. Sato, M. Lerch-Bader, I. Nilsson, S.H. White, and G. von Heijne. Molecular code for transmembrane-helix recognition by the Sec61 translocon. *Nature*, 450(7172):1026–U2, 2007.
10. D. T. Jones. Improving the accuracy of transmembrane protein topology prediction using evolutionary information. *Bioinformatics*, 23(5):538–544, 2007.
11. D.T. Jones, W.R. Taylor, and J.M. Thorton. A model recognition approach to the prediction of all-helical membrane protein structure and topology. *Biochemistry*, 33(10):3038–3049, 1994.
12. V.L. Junker, R. Apweiler, and A. Bairoch. Representation of functional information in the SWISS-PROT data bank. *Bioinformatics*, 15(12):1066–1067, 1999.

13. L. Kall. Prediction of transmembrane topology and signal peptide given a protein's amino acid sequence. *Methods In Molecular Biology*, 673:53–62, 2010.
14. L. Kall, A. Krogh, and E.L.L. Sonnhammer. A combined transmembrane topology and signal peptide prediction method. *Journal of Molecular Biology*, 338(5):1027–1036, 2004.
15. A. Krogh, B. Larsson, G. von Heijne, and E.L.L Sonnhammer. Predicting transmembrane protein topology with a hidden markov model: application to complete genomes. *Journal of Molecular Biology*, 305(3):567 – 580, 2001.
16. J. Kyte and R.F. Doolittle. A simple method for displaying the hydropathic character of a protein. *Journal of Molecular Biology*, 157(1):105–132, 1982.
17. S. Laroum, B. Duval, D Tessier, and J-K. Hao. Multi-neighborhood search for discrimination of signal peptides and transmembrane segments. In Clara Pizzuti, Marylyn D. Ritchie, and Mario Giacobini, editors, *EvoBio*, volume 6623 of *Lecture Notes in Computer Science*, pages 111–122. Springer, 2011.
18. C. Pasquier, V.J. Promponas, G.A. Palaios, J.S. Hamodrakas, and S.J. Hamodrakas. A novel method for predicting transmembrane segments in proteins based on a statistical analysis of the SwissProt database: the PRED-TMR algorithm. *Protein Engineering*, 12(5):381–385, 1999.
19. S.M. Reynolds, L. Kaell, M.E. Riffle, J.A. Bilmes, and W.S. Noble. Transmembrane Topology and Signal Peptide Prediction Using Dynamic Bayesian Networks. *Plos Computational Biology*, 4(11), 2008.
20. B. Rost, P. Fariselli, and R. Casadio. Topology prediction for helical transmembrane proteins at 86% accuracy. *Protein Science*, 5(8):1704–1718, 1996.
21. G.E. Tusnady and I. Simon. The HMMTOP transmembrane topology prediction server. *Bioinformatics*, 17(9):849–850, 2001.
22. H. Viklund, A. Bernsel, M. Skwark, and A. Elofsson. SPOCTOPUS: a combined predictor of signal peptides and membrane protein topology. *Bioinformatics*, 24(24):2928–2929, 2008.
23. S.H. White and G. von Heijne. How translocons select transmembrane helices. *Annual Review of Biophysics*, 37:23–42, 2008.