Prediction of MHC class II binders using the ant colony search strategy

Oleksiy Karpenko\textsuperscript{a}, Jianming Shi\textsuperscript{b, *}, Yang Dai\textsuperscript{a, **}

\textsuperscript{a} Department of Bioengineering (MC063), University of Illinois at Chicago, 851 South Morgan Street, Chicago, IL 60607, USA
\textsuperscript{b} Department of Computer Science and Systems Engineering, Muroran Institute of Technology, 27-1 Mizumoto-Cho, Muroran, Hokkaido 0508585, Japan

Received 17 November 2004; received in revised form 22 January 2005; accepted 22 February 2005

KEYWORDS
MHC class II binding peptide; Ant colony system; Multiple alignment; Scoring matrix

Summary

Objective: Predictions of the binding ability of antigen peptides to major histocompatibility complex (MHC) class II molecules are important in vaccine development. The variable length of each binding peptide complicates this prediction.

Methodology: Motivated by the search properties of the ant colony system (ACS), a method for the identification of an alignment for a given set of short protein peptides has been developed. This alignment is further used for the derivation of a position specific scoring matrix. The distinguishing feature of this method is the use of the collective optimized search strategy of ants for the selection of the alignment.

Results: The performance of the new model has been evaluated with several benchmark datasets. It achieves better or comparable results as compared to the performance of existing methods.

Conclusion: The experiments demonstrate that the predictive performance of the scoring matrix embodies several promising characteristics.

© 2005 Published by Elsevier B.V.

1. Introduction

The ant colony system (ACS) was introduced by Dorigo et al. [1]. Their fundamental concept is derived from the documented behavior of colonies of ants that succeed in finding the shortest path from the nest to food sources by communicating via a collective memory consisting of pheromone trails. Since their pioneering work, many researchers have made significant contributions to algorithm development for important applications such as the symmetric and asymmetric traveling salesman problem (TSP) [1,2], the sequential ordering problem [3], the quadratic assignment problem [4], the vehicle routing problem [5], scheduling problems [6], the graph coloring problem [7], partitioning problems, and tel-
Molecules can accommodate much longer peptides longer than nine amino acids; the MHC class II molecules rarely bind peptides significantly nine amino acids long. However, the MHC binding motif of both MHC classes I and II is approximately nine amino acids. The presence of a binding core with a uniform length for MHC class I molecules simplifies the prediction of the peptide—MHC binding. Several methods have been developed to predict peptide—MHC binding, including simple binding motifs, quantitative matrices, hidden Markov models (HMMs), and artificial neural networks [12–17]. For MHC class I, these gap- and alignment-free methods can be readily applied, since the binding motif is well characterized and most peptides that bind MHC I molecules have similar lengths. It is noted that the supervised HMMs proposed by Mamitsuka [14] have been specially tailored for the detection of binding motifs in peptides binding to both MHC classes I and II and that it can take peptides with various lengths and labeled with their binding values.

The prediction of MHC class II binding peptides is far more difficult. MHC class II molecules bind peptides that are 10–30 amino acids long with a core region of 13 amino acids containing a primary anchor residue. Analysis of binding motifs has suggested that only a core of nine amino acids within a peptide is essential for peptide—MHC binding. In addition, MHC class II molecules contain a single primary anchor, a key feature, which is necessary for the peptide binding. Reported binding peptides usually have variable lengths and undetermined binding cores. Therefore, a search aimed at the binding core can circumvent the complication of variable peptide lengths.

Efforts have been focused on how to align the peptides so that corresponding blocks of the peptides can be identified as the binding cores. Since the alignment usually contains only weak sequence patterns, conventional methods such as ClustalW for multiple sequences will typically fail [18]. New methods developed specifically for the detection of this type of alignment include evolutionary algorithms [19] and the Gibbs sampling method [18]. The evolutionary algorithms use genetic mechanisms (mutation, cross-over and re-production) searching for multiple alignments of binding cores. The fitness of each multiple alignment is associated with its predictive power through a position specific weight matrix. The Gibbs method attempts to find an optimal local alignment by means of Monte Carlo Metropolis sampling of the alignment space. The objective function defined for each alignment is the sum of the entropies (or the Kullback–Leibler distances) of each position in the alignment. After obtaining the binding core, one can establish a scoring matrix which is used for the prediction [18]. Alternatively, a binary classifier, for example, artificial neural networks [15,19], could also be learned by using the amino acids in the binding cores as a positive training set and other non-binding peptides of nine amino acids as a corresponding negative training set.

The advanced search method described herein is inspired by the excellent coordinative ability of the ACS. Basically, the method attempts to find an optimal alignment for a given set of peptides based on the search strategy of the ant colony system. The pheromone guiding process used by ants is defined in terms of the information content of a log–odds matrix calculated from the alignment, which is a function of the sum of the entropies used in the Gibbs method. Ants will locate positions in sequences where an alignment is identified. It is the collective memory of ants that is expected to lead ants rapidly to the discovery of a multiple alignment of high quality. Our method shares a
certain similarity with the Gibbs sampler, since both methods attempt to find the optimal alignment with the same objective function. However, the different search strategies may identify distinct sub-optimal solutions. The difference between the evolutionary algorithm and our method is that the former includes an explicit predicting evaluation in the fitness function.

The remainder of the paper is organized as follows. Section 2 introduces the ACS through the traveling salesman problem and presents the new ACS algorithm for the optimal multiple peptide alignment. Section 3 is devoted to the evaluation of the new algorithm through the comparison with the Gibbs method. The conclusions are presented in Section 5.

2. Ant colony system and its application to the multiple alignment of short protein peptides

2.1. Ant colony system

The $n$-city TSP is used to illustrate how the ACS functions in this subsection. The distance between city $i$ and city $j$ ($i \neq j$) is denoted by $d_{ij}$. The system is assumed to be symmetric, that is, $d_{ij} = d_{ji}$. The TSP is defined by the problem of finding the minimum distance tour when each city is visited exactly once.

The basic operations of the ACS can be described as follows. Each path followed by an ant is associated with a candidate solution of the problem. When an ant follows a particular path, the amount of pheromone deposited on that path is proportional to the quality of the corresponding candidate solution of the problem. At the initial step, each pair of cities is allocated the same quantity of pheromone $\frac{1}{nD}$, where $D$ is the distance corresponding to the initial tour.

An ant $l$ chooses the next city according to the following rules. Let $p$ be a random value generated following a uniform distribution over $[0,1]$. For a given parameter $p_0$ if $p \leq p_0$, the ant chooses the next city $s$ by the following formula (1):

$$ s := \arg \max_{u \in U} \{ \tau(r,u) \cdot \eta(r,u)^{\beta} \}, $$

(1)

where $U$ is the set of unvisited cities, $\tau(r,u)$ the pheromone on the path between city $r$ and city $u$, $\beta$ is a parameter, and $\eta(r,u)$ is defined as $\frac{1}{D}$. If $p > p_0$, the ant chooses the next city $s$ by the following probability (2):

$$ p(r,s) = \frac{[\tau(r,s)]^\beta \cdot [\eta(r,s)]^\beta}{\sum_{u \in U} [\tau(r,u)]^\beta \cdot [\eta(r,u)]^\beta}, \text{ for } s \in U. $$

(2)

Note that the value $p(r,s)$ is generally distinct for cities $s \in U$. When the next city $s$ is determined, ant $l$ moves to city $s$ and the local pheromone between city $r$ and city $s$ is updated by

$$ \tau(r,s) := (1 - \rho) \cdot \tau(r,s) + \rho \cdot \Delta \tau(r,s), $$

(3)

in which $\Delta \tau(r,s)$ is the local contribution of pheromone made by ant $l$ from its moving, and $\rho \in (0,1)$ is a parameter. The value $\Delta \tau(r,s)$ is set to $\eta(r,s)$. When all tours have been completed by all ants, the amount of pheromone on the globally best tours is updated by

$$ \tau(r,s) := (1 - \alpha) \cdot \tau(r,s) + \alpha \cdot \Delta \tau(r,s), $$

(4)

for $(r,s)$ on the current best tour

where $\Delta \tau(r,s)$ is the amount of pheromone contributed from the best tour and $\alpha \in (0,1)$ is a parameter characterizing the evaporation rate of the old pheromone. Similarly, $\Delta \tau(r,s) = \frac{1}{D^*}$, with $D^*$ being the total distance of the best tour. Note, in this framework, the local search and pheromone update are controlled by formulas (1) and (3), while the global search and pheromone update are governed by formulas (2) and (4). Moreover, an alternative way for the local update of the pheromone can be made after a tour is established by ant $l$, that is

$$ \tau(r,s) := (1 - \rho) \cdot \tau(r,s) + \rho \cdot \Delta \tau(r,s), $$

(5)

for each $(r,s)$ on the tour,

in which $\Delta \tau(r,s)$ is the local contribution of pheromone arising from the new tour completed by ant $l$. The value of $\Delta \tau(r,s)$ is usually defined as $\frac{1}{nb l}$ in which $D_l$ is the total distance of the tour.

The algorithm based on the ACS for the TSP can be summarized as follows:

Algorithm 1 (ACS-TSP).

Step 0: Initialize parameters.
Step 1: Set all ants on starting cities.
Step 2: Each ant chooses the next city by the transition rules (1) and (2); update the pheromone by (3).
Step 3: When all tours are completed, update the pheromone by (4).
Step 4: If the stopping condition is satisfied, then stop, otherwise go to Step 1.

It has been reported that the performance of the algorithm is sensitive to the choice of parameters in the ACS. Several experiments have indicated that the following values are efficient for the TSP:
\[ p_\theta = 0.9, \alpha = p = 0.1, \text{ initial pheromone } \tau_0 = \frac{1}{n^2} \text{ and } \\
\beta = 2[20]. \text{ For the multiple alignment problem, we found that the appropriate parameters are quite different from those for the TSP. We will discuss this issue in detail below.} \]

### 2.2. Identification of binding core of peptides based on the ACS

As described in Introduction, the success of the prediction of MHC class II binders depends on the identification of an accurate binding core of peptides from a given set of peptides with various lengths. Recall that the binding cores consist of a block with nine amino acids from appropriately aligned peptides. In contrast to the general multiple alignment problem, the multiple alignment considered here has to be ungapped. Fig. 1 shows the binding cores in the alignment for a set of peptides.

Suppose that \( n \) peptide sequences \( s_j (j = 1, \ldots, n) \) are given. We denote the length of \( s_j \) by \( |s_j| \), and the \( i \) th amino acid of \( s_j \) by \( s_{j,i} \). Our task is the direct identification of the starting position of the binding core for each peptide. In order to design a suitable framework for the ACS, we consider an alignment of nine amino acids from any starting positions as a feasible solution for our optimization problem. More precisely, given a starting position \( l_j \) in each sequence \( j \) such that \( l_j + 8 \leq |s_j| \), we have \((s_{1,l_1}, \ldots, s_{n,l_n})\) as a feasible solution. We define a binding core matrix \( M \) as follows:

\[
M(s_{1,l_1}, \ldots, s_{n,l_n}) = \begin{bmatrix}
s_{1,l_1} & s_{1,l_1+1} & \ldots & s_{1,l_1+8} \\
s_{2,l_2} & s_{2,l_2+1} & \ldots & s_{2,l_2+8} \\
\vdots & \vdots & \ddots & \vdots \\
s_{n,l_n} & s_{n,l_n+1} & \ldots & s_{n,l_n+8}
\end{bmatrix}
\]

The objective function corresponding to the alignment \((s_{1,l_1}, \ldots, s_{n,l_n})\) is defined as

\[
E(s_{1,l_1}, \ldots, s_{n,l_n}) = \sum_{i=1}^{9} \sum_{l=1}^{|s_j|} \left( p_{l,j} \log \left( \frac{p_{l,i}}{q_i} \right) \right),
\]

where \( p_{l,i} \) is an occupancy number for amino acid \( l \) at position \( i \) in the alignment and \( p_{l,1} \) is the pseudo-count and sequence weight corrected amino acid frequency of amino acid \( l \) at position \( i \) in the alignment. Finally, \( q_i \) is the background frequency of amino acid \( l \), which is taken from the SWISS-PROT database [21] in this study. The calculation of \( p_{l,1} \) is based on the method described by Altschul et al. [22]:

\[
p_{l,1} = \frac{a \cdot f_{l,i} + b \cdot g_{l,i}}{a + b},
\]

where \( f_{l,i} \) is the observed frequency of residue \( l \) at position \( i \), \( g_{l,i} \) the pseudo-count frequency of residue \( l \) at position \( i \), \( a \) the effective sequence number, and \( b \) the weight on the pseudo-count correction. The value of \( g_{l,i} \) is calculated with the method by Altschul et al. [22]. The observed frequency \( f_{l,i} \) is weighted based on the method described by Henikoff and Henikoff [23] in this study. The value of \( a \) is determined by the mean number of different amino acids in the alignment, while \( b \) is a prescribed parameter.

Obviously, the objective function \( E(\cdot) \) associated with each alignment is equal to the sum of the relative entropies. This function aims at an estimate of the significance of an alignment found by an ant through the comparison of the information content with a null model, one that is defined in terms of the background amino acid frequencies. This form of evaluation for an alignment has been used extensively in studies of multiple sequence alignment [18].

We now apply the ACS to find the global optimal solution \((s_{1,l_1}, \ldots, s_{n,l_n})\) which maximizes the function \( E(\cdot) \) over all feasible alignments. Four technical issues have to be resolved for this application: (a) how to define a proper tour for ants, (b) how to assign a proper pheromone form and quantity, (c) how to define the distance between two positions in two consecutive sequences, and (d) how to set specific parameters to obtain better solutions efficiently. We explain (a)–(c) below and discuss (d) in Section 3.

In order to assemble a tour, a dummy sequence 0 with only one residue is assigned a number 0 and is placed at the top of the given sequences. All ants are ready to depart from this dummy sequence to look for a position on the subsequent sequence 1. After the ants have moved from sequence 0 to sequence 1, they...
Prediction of MHC class II binders using the ant colony search strategy

The move of an ant from one position $s_{ij}$ in a sequence $j$ to a position $u_{j+1j+1}$ in the next sequence $j + 1$ corresponds to the generation of an augmented alignment which is different only at the aligning position in sequence $j + 1$. The solid line in Fig. 2 is the case. Accordingly, it is reasonable to consider the pheromone between the move of the ant by the objective function value corresponding to the augmented alignment. Let this alignment be denoted by $M(s_{1l_1}, \ldots, s_{jl_j}, u_{j+1l_{j+1}}, s_{j+2l_{j+2}}, \ldots, s_{nl_n})$. Then the pheromone associated with this move is defined as $u_{j+1j}$ that is not same

$$
\eta(s_{ij}, u_{j+1,j+1}) = \frac{1}{n[E(s_{1l_1}, \ldots, s_{jl_j}, u_{j+1l_{j+1}}, s_{j+2l_{j+2}}, \ldots, s_{nl_n})]^{1.5}}.
$$

Here, the value $|E(s_{1l_1}, \ldots, s_{jl_j}, u_{j+1l_{j+1}}, s_{j+2l_{j+2}}, \ldots, s_{nl_n})|$ is considered as the “distance” between the two positions. Thus, the pheromone defined in (8) has a reciprocal proportion of the distance that makes sense in ACS. Based on this value, we can evaluate the consequence of moving from position $s_{ij}$ in sequence $j$ to a position $u_{j+1,j+1}$ in sequence $j + 1$.

Figure 2  A set of sequences and their possible alignment positions. A tour for an ant starts from the dummy sequence, connects two aligning positions of two consecutive sequences, and returns to the dummy sequence after the last sequence is visited.
2.3. Algorithm

Following the discussion above, we propose the algorithm ACS-MULTIALIGNMENT. At the initial step, we randomly select one multiple alignment for each ant \( k \), and calculate its function value \( E_k^0 \). We then assign \( \frac{1}{n^k} \) as the initial pheromone of \( r \) for each pair of possible positions between two consecutive sequences \( j \) and \( j + 1 \) (\( j = 0, 1, \ldots, n - 1 \)), where \( E_k^0 := \max_k \{ E_k \} \). All ants start from sequence 0. When an ant moves from one sequence to the next, we calculate the value of (8) to determine the best current position in sequence \( j + 1 \). The complete algorithm is listed as follows.

Algorithm 2. ACS-MULTIALIGNMENT

Step 1
Form an initial feasible solution \( (s_{j,1}^k, \ldots, s_{n_h}^k) \) for each ant \( k \) (\( k = 1, \ldots, m \)).

Step 2
For each pair of \( (s_{i,j}^k, s_{j+1,1}^k) \) \( j = 0, 1, \ldots, n - 1 \), \( i, j \in PS_j \),

\[ \tau(s_{i,j}^k, s_{j+1,1}^k) := \frac{1}{n^k} \]

Step 3
for each \( j = 0, 1, \ldots, n - 1 \) do

\[ \tau(s_{j,j}^k) := \frac{1}{n^k} \]

Step 4
for each ant \( k \) \( (k = 1, \ldots, m) \)

\[ \tau(s_{j,j}^k) := \frac{1}{n^k} \]

Step 5
Create \( p \in (0, 1) \) randomly, then choose a starting position \( u \) for peptide \( j \) as follows.

if \( p \leq p_0 \), compute \( \max_{u \in PS_j} \{ \tau(r, u) \times (\eta(r, u))^\beta \} \)

and obtain \( u_0 \) such that

\[ \tau(r, u_0) \times (\eta(r, u))^\beta \]

\[ = \max_{u \in PS_j} \{ \tau(r, u) \times (\eta(r, u))^\beta \} \]

if \( p > p_0 \), compute probabilities

\[ p(r, u) = \frac{\tau(r, u) \times (\eta(r, u))^\beta}{\sum_{r \in PS_j} \tau(r) \times (\eta(r))^\beta} \]

for each \( u \in PS_j \),

and choose \( u_0 \) from \( PS_j \) with the above probabilities.

\[ s_{j+1,j+1}^k := u_0 \]

Move all ants back to sequence 0.

Step 3 (Global pheromone updating)

compute \( E_{best} := \max_{k=1,\ldots,m} E(s_{j,j}^k, \ldots, s_{n_h}^k) \), and obtain an ant \( k \), such that

\[ E(s_{j,j}^k, \ldots, s_{n_h}^k) = E_{best} \]

for each pair of \( (s_{i,j}^k, s_{j+1,1,j+1}) \) \( j = 0, 1, \ldots, n - 1 \) do

\[ \tau(s_{i,j}^k, s_{j+1,1,j+1}) := (1 - \rho) \tau(s_{i,j}^k, s_{j+1,1,j+1}) + \rho \frac{1}{E_{best}} \]

Step 4 go to Step 2 and repeat several times.

end of Algorithm.

3. Computational experiments

In this section, we present the details of the computational study, discuss the behavior of the algorithm, and assess the quality of the position specific scoring matrix obtained from the alignment through the comparison with the Gibbs method.

3.1. Parameters

Several parameters are associated with the algorithm ACS-MULTIALIGNMENT. Their values were determined on the basis of the search of quadruplet \( (\alpha, \beta, \rho_0, \rho) \) in a region formed by the prescribed ranges of parameters for a small scale dataset described below. In the following experiments, they were set to the values \( \alpha = 0.8; \beta = 0.2; \rho_0 = 0.7; \rho = 0.8 \). The parameter \( b \) in the definition of \( p_i^j \) of (7) was set to \( b = 50 \).

3.2. Evaluation with a small scale dataset of experimentally confirmed binding cores

A dataset of 68 binders to MHC class II molecule HLA-DR4(B1*0401) was retrieved from the SYFPEITHI database [17]. The nine-residue core for each of these binders had been determined experimentally. The possible starting position for the putative binding core in each sequence was restricted to hydrophobic and neutral residues. This set was used for the detailed evaluation of the algorithm. In the training process, all nine-length residues starting from a hydrophobic and neutral residue were considered as putative binding cores. The set of 68 binders was divided into 5 almost equally sized subsets. A negative testing set of 68 peptides was prepared by randomly selecting the entries from a dataset of 161 known HLA-DR4(B1*0401) non-bin-
To evaluate a testing peptide, the scores of all putative binding cores from the peptide based on the position specific scoring matrix were computed. The highest value among all the scores was considered as the score of the given peptide. The prediction of a binder or a non-binder was made by a prescribed threshold of the score. The area under the relative operative characteristics (ROC) curve, denoted as $A_{roc}$, was used for the evaluation of the performance of the prediction.

The Gibbs method of Nielsen et al. [18] was implemented for the comparison. The same objective function and associated parameters for the ACS were used. The $T$ value in the Gibbs method was initialized to 0.15 and decreased to 0.001 in 10 uniform steps; 5000 Monte Carlo moves were performed at each value of $T$ and a total of 50,000 moves for one run. This process was repeated 10 times and the final scoring matrix was derived based on the same procedure described above for the ACS method.

The $A_{roc}$ values for each of the five testing datasets are shown in Fig. 3. The average of the $A_{roc}$ values is 0.817 with a standard deviation of 0.076 for the ACS algorithm; the corresponding values for the Gibbs method are 0.791 and 0.093, respectively. This indicates that the ACS and the Gibbs methods yield almost identical performance.

In order to examine the rate of convergence of the algorithm, the best function values found by the proposed algorithm at each iteration for one of the training sets is plotted (dashed dot line) in Fig. 4. Clearly, a relatively large increase in the function value was achieved at around 250 iterations for our algorithm. Beyond that value, it stayed at a stable level, experiencing only small occasional deviations from that level, to which it returned rapidly. This implies that 500 iterations may be sufficient for obtaining a good solution for a dataset of this size.

In order to represent the objective function values found by the Gibbs method in the same figure, the 50,000 values were divided into 500 groups, i.e., the first 100 values were in group 1 and the second 100 values were in group 2, and so on. The average of the 100 values in each group was then calculated. This procedure gives a total of 500 average values. These 500 values were plotted (solid line) in Fig. 4. The curve corresponding to the Gibbs method indicates an earlier settlement to local optimal solutions. However, the corresponding objective value is lower than that of the ACS method.
method. Furthermore, boxplots of the 50 objective function values obtained from the 10 runs of each method for the five different training sets are presented in Fig. 5. These results demonstrate the exceptional ability of the ACS method in the search for better solutions.

The straight line in Fig. 4 corresponds to the objective function value of the alignment from the experimentally determined binding cores. Surprisingly, the line is located significantly below the converged function values, an outcome that indicates that the alignments corresponding to the higher function values are not necessarily representing actual binding cores. This result implies that the parameters in (7) have not been optimally tuned. Nevertheless, the $A_{roc}$ values obtained from the experiment imply that these alignments may be quite close to the valid ones. The alignments found at different iterations by the ACS algorithm are given in Fig. 6 for a set of 10 peptides.

### 3.3. Evaluation on benchmark datasets

The training data were extracted from MHCPEP [27] and SYFPEITHI [17] for a set of 535 peptides binding to the MHC class II molecule HLA-DR4(B1*0401). Peptides that do not allow hydrophobic or neutral residues at the first $N - L + 1$ positions were removed, where $N$ is the peptide length and $L$ is the motif length. Furthermore, the dataset was reduced by removing unnatural peptides with more than 75% of alanine. The identical peptides were also removed. This reduction gives a total of 454 peptides in the training set. Eight testing datasets were obtained from MHC-Bench [25]. Peptides with an associated binding value of zero were assigned to be non-binders and all other peptides were binders. The homology reduction was performed so that no peptide in the testing sets has a sequence identity of more than 90% over an alignment of at least nine amino acids in the training peptides. The sizes of training and testing data sets are summarized in Table 1.

The performance of the proposed algorithm was evaluated on the eight datasets and compared with that of the Gibbs method [18]. The experimental setting was the same as that for the small scale dataset. The $A_{roc}$ value for each dataset was shown in Fig. 7. The average $A_{roc}$ value and the standard deviation over the eight testing sets are respectively

<table>
<thead>
<tr>
<th>Set</th>
<th>Original</th>
<th>Homology reduced</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$N_b$</td>
<td>$N_{nb}$</td>
</tr>
<tr>
<td></td>
<td>$N_b$</td>
<td>$N_{nb}$</td>
</tr>
<tr>
<td>Training</td>
<td>535</td>
<td>454</td>
</tr>
<tr>
<td>Set1</td>
<td>694</td>
<td>323</td>
</tr>
<tr>
<td>Set2</td>
<td>381</td>
<td>292</td>
</tr>
<tr>
<td>Set3A</td>
<td>373</td>
<td>217</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Testing</th>
<th>$N_b$</th>
<th>$N_{nb}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set3B</td>
<td>279</td>
<td>216</td>
</tr>
<tr>
<td>Set4A</td>
<td>323</td>
<td>323</td>
</tr>
<tr>
<td>Set4B</td>
<td>292</td>
<td>292</td>
</tr>
<tr>
<td>Set5A</td>
<td>70</td>
<td>47</td>
</tr>
<tr>
<td>Set5B</td>
<td>48</td>
<td>37</td>
</tr>
</tbody>
</table>

$N_b$ and $N_{nb}$ denote the numbers of binders and non-binders, respectively.

![Figure 5](image-url) Box plots of 50 objective function values found by 10 runs of the ACS and the Gibbs methods for five training sets derived from the set of 68 binders, respectively. The line in the box indicates the median of the values; the top and bottom lines indicate the 25 and 75% percentiles, respectively.

![Figure 6](image-url) For an input set of peptides (left panel), the alignments found by the algorithm ACS at the end of the 1st, 200th and 500th iterations are shown in the subsequent panels, respectively. The underlined part is the experimentally identified binding core.

**Table 1** Description of the MHC class II training and testing datasets

<table>
<thead>
<tr>
<th>Set</th>
<th>Original</th>
<th>Homology reduced</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$N_b$</td>
<td>$N_{nb}$</td>
</tr>
<tr>
<td></td>
<td>$N_b$</td>
<td>$N_{nb}$</td>
</tr>
<tr>
<td>Training</td>
<td>535</td>
<td>454</td>
</tr>
<tr>
<td>Set1</td>
<td>694</td>
<td>323</td>
</tr>
<tr>
<td>Set2</td>
<td>381</td>
<td>292</td>
</tr>
<tr>
<td>Set3A</td>
<td>373</td>
<td>217</td>
</tr>
<tr>
<td>Testing</td>
<td>$N_b$</td>
<td>$N_{nb}$</td>
</tr>
<tr>
<td>Set3B</td>
<td>279</td>
<td>216</td>
</tr>
<tr>
<td>Set4A</td>
<td>323</td>
<td>323</td>
</tr>
<tr>
<td>Set4B</td>
<td>292</td>
<td>292</td>
</tr>
<tr>
<td>Set5A</td>
<td>70</td>
<td>47</td>
</tr>
<tr>
<td>Set5B</td>
<td>48</td>
<td>37</td>
</tr>
</tbody>
</table>

$N_b$ and $N_{nb}$ denote the numbers of binders and non-binders, respectively.
that is lower than that of the matrix derived from the Gibbs method for all cases; the Gibbs method outperformed the TEPITOPE for all cases except one. By indirect induction, it is reasonable to conclude that the ACS method will compare favorably with the existing predicting methods with optimally tuned parameters.

Currently, the ACS algorithm presents a relatively slower convergence compared to that of the Gibbs method. However, this can be readily improved through efficient implementation of the algorithm. If a longer training process leads to a better predictor, it is still practically important, since the training is only required to be performed once.

5. Conclusion

An algorithm based on the concept of the ant colony system (ACS) for solving multiple alignment problem arising in the identification of binding cores for a set of MHC class II binding peptides has been developed. A position specific scoring matrix was derived from the alignment and was employed for the discrimination between the major histocompatibility complex class II binders and non-binders. The effectiveness of the algorithm was demonstrated on several benchmark datasets. The chief outcomes of this study are the findings that (1) the predictive power of the scoring matrix is comparable as one of the most advanced predictors presently in use (i.e., the Gibbs method) and (2) the method can converge to better alignments.

Acknowledgments

The work of JS was mainly done during his visit at the University of Illinois at Chicago. This author is partially supported by Satellite Venture Business Laboratory (SVBL), Muroran Institute of Technology. The work of YD is supported in part by the National Science Foundation (EIA-022-0301) and Naval Research Laboratory (N00173-03-1-G016). We are grateful to anonymous referees for their valuable suggestions.

References


Figure 7 The $A_{ROC}$ values of ACS and the Gibbs methods for the eight benchmark datasets.

0.602 and 0.089 for the ACS algorithm with corresponding respective values of 0.599 and 0.093 for the Gibbs method. Again a comparable performance of the two approaches was evidenced for the benchmark datasets.

4. Discussion

From the computational study conducted in this work, the new proposed method achieved comparable performance to that of the Gibbs approach. However, the reported $A_{ROC}$ values for the same testing sets in Neilsen et al. [18] were higher than those obtained from our experiments. We considered the parameter setting in the objective function as the major cause of this outcome. Furthermore, the assignment of different weights to different positions in the definition of the objective function (6) may reduce this difference. For example, positions 2, 3 and 9 could be assigned higher weights, since they are the anchor positions for the peptide binding. We note specifically that Neilsen et al. demonstrated that weighting was important for the achievement of the high accuracy of their prediction.

The direct comparison of the ACS method with the other existing predictors was not conducted in this work, since Neilsen et al. has reported an extensive computational study to compare the Gibbs method with the other prediction methods such as HMMER and SYFPEITHI [18]. According to their work, the Gibbs method achieved better predictive power for most of the evaluation sets, including the eight sets used in this study. In addition, they also compared the performance of the Gibbs method with the TEPITOPE [28] and a ClustalW derived weight-matrix for the benchmark datasets. The gap penalty was taken high to ensure ungapped alignment in using the ClustalW. They demonstrated that the ClustalW weight-matrix has a performance


