行政院國家科學委員會補助專題研究計畫成果報告

※※※※※※※※※※※※※※※※※※※※※※※※※

※ 利用核糖核酸結構預測與核糖核酸-蛋白質互動關係分析推論 ※

※ 蛋白質結構(3/3) ※

※※※※※※※※※※※※※※※※※※※※※※※※※

計畫類別：□個別型計畫 ■整合型計畫

計畫編號：NSC 96－2627－B－009－003－

執行期間：96 年 8 月 1 日至 97 年 7 月 31 日

計畫主持人：胡毓志

共同主持人：

計畫參與人員：劉康平，劉怡馨，陳彥修，許彥超

成果報告類型(依經費核定清單規定繳交)：□精簡報告 ■完整報告

本成果報告包括以下應繳交之附件：

□赴國外出差或研習心得報告一份

□赴大陸地區出差或研習心得報告一份

■出席國際學術會議心得報告及發表之論文各一份

□國際合作研究計畫國外研究報告書一份

執行單位：交通大學 資訊工程系

中　華　民　國　97 年　9　月　15　日

行政院國家科學委員會專題研究計畫成果報告

國科會專題研究計畫成果報告撰寫格式說明

Preparation of NSC Project Reports

計畫編號：NSC 96-2627-B-009 -003-
執行期限：96 年 8 月 1 日至 97 年 7 月 31 日

主持人：胡毓志　　交通大學資訊工程系
計畫參與人員：劉康平，劉怡馨，陳彥修，許彥超 交通大學生資所

中文摘要

本計劃首要目標在建立序列－結構－功能間的關係，並深入瞭解 RNA/DNA 與蛋白質分子間交互作用。RNA/DNA、蛋白質及其最後生物功能間的複雜關係與疾病、藥物、生醫研究密不可分，仍有極大的發展空間，這也是新興的蛋白質體學亟待解決的問題。本計畫的研究成果，有助於高精確度地分析、預測 RNA/DNA 及蛋白質結構與生化網絡的構成方式。簡言之，即是以序列及結構資訊為基礎，解析蛋白質－RNA/DNA 交互作用系統。

本計劃包括核醣核酸(RNA)二級結構預測及分類(Chapter 1)、蛋白質－去氧核醣核酸(DNA)交互作用系統(Chapter 2)，以及蛋白質－核醣核酸交互作用系統(Chapter 3)。三者間的緊密結合，可以涵蓋由 RNA 序列到生物功能間各層次的完整研究。

本計畫所針對的目標有六：

一、 建立預測 RNA 二級結構及 RNA 分類模組。(Chapter 1)

二、 利用 RNA 二級結構及 protein-RNA 嵌合工具預測 protein-RNA 交互作用。(Chapter 1、3)

三、 將蛋白質功能區域(domain)及蛋白質結晶結構作為定義 protein-DNA 交互作用的基礎，結合從已知生化路徑中萃取的 protein-DNA 間作用關係，預測未知生化路徑或擴張已知生化網絡。(Chapter 2)

四、 以已發展完成的 protein-ligand 嵌合工具(GEMDOCK)尋找可能的 protein-RNA 交互作用。(Chapter 3)

五、 比較 protein-RNA 與 protein-DNA 交互作用之特性，以建立更完善之預測系統。(Chapter 1、2、3)

六、 發展 protein-RNA 嵌合預測工具及更精確的計分函式，此工具結合物理及生化知識，可較現有工具更準確地預測 protein-RNA 交互作用及分子表面結合位置。(Chapter 3)

針對上述提出的六大目標，在本計畫的三年執行期限間(2005-2008 年)，研究團隊成員共發表論文 5 篇，研究成果十分豐碩。整體而言，我們相信在本計畫實行的三年間，研究團隊已順利達成執行目標，並取得豐富的研究成果。這些成果對於序列－結構－功

能間的關係及 protein-RNA 及 protein-DNA 交互作用相關領域的後續研究將有所助益。

Abstract

Our central theme is the study of the sequence-structure-function relationships and protein-RNA and protein-DNA interactions. RNA/DNA molecules are the key players in the biochemistry of the cell, playing many important roles in regulation, catalysis and structural support. These biological interaction networks still are not only hot issues in system biology but also very useful in practical biological research. Hence, it becomes increasingly important for computational biologists to develop reliable and efficient computational approaches to study sequence-structure-function relationships, protein-RNA and protein-DNA interactions, and to predict reliable 3D structures from the sequence level in order to help functional genomics research.

This project covers research areas from RNA, DNA and protein networks of a biological system. Close cooperation between the RNA secondary structure prediction and clustering (Chapter 1), protein-DNA interaction (Chapter 2), and protein-RNA docking system (Chapter 3) will be advantageous and valuable to researchers to find RNA sequence-structure-function relationships and protein-RNA interactions.

The major objectives of this project are listed as follows:

1. Developing a prediction system of RNA secondary structure and clustering. (Chapter 1)
2. Predicting protein-RNA interaction based on RNA structure prediction and protein-RNA docking system. (Chapter 1 and 3)
3. Deriving domain-domain interactions from known protein-DNA complexes and known biochemical pathways to predict protein-DNA interactions and find new biochemical pathways. (Chapter 2)
4. Predicting docking conformation of protein-RNA interaction using GEMDOCK. (Chapter 3)
5. Compare protein-DNA and protein-RNA interaction characteristic to improve prediction quality of protein-DNA and protein-RNA interactions. (Chapter 1, 2 and 3)
6. Developing a new evolutionary protein-RNA docking method and creating a new protein-RNA binding model by integrating physical-based and knowledge-based scoring functions to reduce calculating quantity and to improve prediction quality of protein-RNA interactions. (Chapter 3)

In summary, we have published 5 papers during 2005-2008. We believe that we have achieved fruitful results in this integrated project. This interdisciplinary research project covers research areas from protein-RNA interactions to protein-DNA interactions. We consider that these achievements will be advantageous and valuable to researchers to study sequence-structure-function relationships and protein-RNA and protein-DNA interactions.

Chapter 1:
Protein/RNA structure prediction and clustering

1.1 Introduction

RNA plays a crucial role in posttranscriptional regulation. Similar to transcriptional regulation, post-transcriptional regulation is often accomplished by the binding of proteins to specific motifs in mRNA molecules. Most of the current structural bioinformatics research is focused on proteins, and yet thousands of genes produce transcripts exerting their functions without ever producing protein products. A fundamental principle of biology is that a stable 3D structure is essential for biological functions. Many functional RNAs have evolutionarily conserved secondary structures in order to fulfill their roles in a cell. In another word, unlike DNA binding proteins, which recognize motifs composed of conserved sequences, RNA protein binding sites are more conserved in structures than in sequences. Various computational methods for the prediction of RNA secondary structures have been developed. According to the search strategies applied and the structure representations used, they can be roughly classified into the following categories: **(1)** free energy minimization [1-3] **(2)** comparative sequence analysis [4, 5] **(3)** stochastic context-free grammars [6-8] **(4)** heuristics [9-11] **(5)** graph theoretical approach [12, 13] and **(6)** hybrid [14-17]. A lot of works have been done for single RNA structure prediction; however, as more RNA sequence data have been produced, finding characteristic structure motifs within RNA families becomes very important.

The goal of this section is to understand relationship of RNA/protein structure and function by secondary structure prediction and clustering. We have studied RNA/protein structure prediction and clustering in the past three years, and published four papers as fellow:

1. **Y. Hu**, "RNA Clustering and Secondary Structure Prediction", International Conference on Mathematics and Engineering Techniques in Medicine and Biological Science, 2005.
2. S. Ku and **Y. Hu**, "A Multistrategy Approach to Protein Structural Alphabet Design", Biocomp 2006.
3. K. Chen and **Y. Hu** "Bicluster Analysis of Genome-wide Gene Expression", IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology, 2006
4. C. Huang and **Y. Hu** "A Two-stage Approach to Finding Common Structure Elements in Unaligned RNA Sequences", Biocomp 2007

In the first year (2005), we have proposed a new adaptive method that conducts structure

prediction and clustering simultaneously, since some current approaches can now identify common structure motifs from a set of RNAs, they typically assume the given set forms a single family, which is not necessarily correct. The performance of this study is demonstrated on several real RNA families, and showed very promising results.

In the second year (2006), we demonstrated how the structural alphabet can be used with conventional 1D sequence alignment algorithms and presented its results. A comparative study of our alphabet with one of recently developed structural alphabets also showed a competitive result. Moreover, we proposed a new biclustering method based on the framework of market basket analysis in which a bicluster is described as a frequent itemset. As a feasibility test, we compared it with several standard clustering algorithms on a genome-wide yeast microarray dataset, and it showed very promising results.

In the third year (2007), unlike some methods that find consensus structures from a multiple sequence alignment if available or others that align sequences and structures simultaneously, we have developed an approach which separates consensus motif finding from sequence folding. After applying RNA folding algorithms to each sequence of given RNAs as a preprocess, we then combine structure decomposition and Gibbs sampling techniques to identify common structure motifs in unaligned RNA sequences. To demonstrate the performance, we tested it on several RNA families in Rfam. The experimental results show our new approach is competitive with other current prediction systems.

1.2 Motivation

RNA molecules are the key players in the biochemistry of the cell, playing many important roles in regulation, catalysis and structural support. Like proteins, their functions generally depend on their structures. Although structural genomics, the systematic study of all macro-molecular structures in a genome, is currently focused more on proteins, thousands of genes produce transcripts exerting their functions without ever producing protein products. Most of the current structural bioinformatics research is focused on proteins, and yet thousands of genes produce transcripts exerting their functions without ever producing protein products [18-20]. We can easily argue that the comprehensive understanding of the biology of a cell requires, besides proteins, the knowledge of the identities of all functional RNAs (both noncoding and protein-coding) and their molecular structures.

A fundamental principle of biology is that a stable 3D structure is essential for biological functions. Many functional RNAs have evolutionarily conserved structures in order to fulfill their roles in a cell. Some of the functions can be presented by functional motifs, such as several well-understood structurally conserved RNA motifs in viral RNAs, e.g. the TAR and RRE structures in HIV and the IRES regions in Picornaviridae [21]. Although experimental assays for basepairing in RNAs constitute the most reliable method for secondary structure determination, yet it is often difficult and expensive to acquire the 3D spectrum data of RNA molecules [22].

1.3 Methods

We have studied about relationship of Protein/RNA structure and function in the past three years. In the first year, in order to understand RNA structure and function, we have studied how to cluster RNA by its function and predict its secondary structure. This study provides RNA functional classification and RNA structure prediction. The results have been published in International Conference on Mathematics and Engineering Techniques in Medicine and Biological Science. In the second year, in order to understand relationship between RNA and protein structure, we have studied protein structural alphabet design and bicluster analysis of genome-wide gene expression. These studies provide how to analyze/predict protein structure and a new biclustering method based on the framework of market basket analysis in which a bicluster is described as a frequent itemset. In the third year, we have proposed a two-stage approach to finding common structure elements in unaligned RNA sequences, in order to model RNA structure and predict its function more correctly. The detail of these studies are described as fellow:

First Year:

1.3.1 RNA Clustering and Secondary Structure Prediction

Unlike previous studies of RNA secondary structure prediction whose input is either a single RNA sequence or a known class of functionally related sequences, our new method is instead applied to a set of unaligned RNA sequences which consist in an unknown number of classes. In order to find a reasonable partition for a given set of unaligned RNAs without knowing beforehand how many clusters actually existing in this set, we assume that each cluster is likely a functional family that contains characteristic structure motifs. Based on this assumption, our new method is focused on finding significant consensus structure motifs that can be used to characterize the families of RNAs. Since the number of clusters and its size are unknown in advance, we take a generateand-test strategy that iteratively adjusts the hypothesized cluster size until some significant consensus structure elements can be found

associated with this cluster. After a cluster is obtained, all its members are then removed from the given RNAs. We repeat the same separateand-conquer strategy to identify other clusters from the remaining RNAs.

Generate-and-Test

The generate-and-test strategy we use is an adaptive approximation approach that systematically revises the hypothesized cluster size. During the generate-and-test process, the cluster size is defined by a range between an upper bound U and a lower bound L. Without any prior information of clusters, the cluster size is initialized within a range between an upper bound U=n and a lower bound L=0, that is, we first assume that all the given RNA sequences consist in an entire family. To the entire family, a genetic programming-based structure prediction method is applied to look for the fittest consensus structure motifs. If the specificity of the structure motifs associated with a cluster exceeds or equals some pre-specified threshold, the hypothesis of the cluster is accepted, and the cluster along with the associated structure elements will be reported. On the other hand, low specificity suggests that the current hypothesized cluster size is too big to be real and needs to be decreased. In this case, we reduce the current hypothesized cluster, and search the fittest consensus structure motifs and evaluate their specificity again. If the specificity is still lower than the threshold, we further decrease the cluster size. The same process for cluster size reduction can be repeated till we find a cluster with structure motifs of highspecificity. On the contrary, if the specificity is over or equal to the threshold, one of the two possibilities holds: **(1)** the current cluster is real, and any more sequences added will be harmful to the specificity of consensus structures, or **(2)** the current cluster found is only a subset of a bigger real cluster. To verify which event actually happens, we increase the cluster size and a new search for the fittest consensus structure motifs is conducted. As each update generates a tighter range for cluster size, we expect the cluster size will eventually converge to the appropriate one.

Secondary Structure Element Prediction by Genetic Programming

The objective here is to learn the structure elements that can be used to distinguish the given functionally related sequences from the random sequences. We modify the fitness function of our previous work [23] on RNA consensus secondary structure prediction to find significant structure elements from a dataset that may contain multiple variable-sized clusters of unaligned sequences.

The fitness function is used to measure the quality of individuals (i.e. candidate structure elements) in a population. The higher the fitness of an individual, the better its chances of survival to the next generation. In the previous work, the input dataset was assumed to be a single class of functionally related RNA sequences. We were interested in those structure

elements that can reflect the characteristics conserved in a family, e.g. the RNA protein binding sites. Derived from the F-score, the fitness function was aimed to balance the importance of two measures, recall (i.e. sensitivity) and precision (i.e. positive predictive value) [10]. It assigns higher values to those structural motifs commonly shared by the given family of RNAs, and rarely contained in random sequences. For a given set of RNA sequences that form a single family only, the fitness function used in [10, 23] can effectively guide the evolutionary process in genetic programming. Nevertheless, when the input dataset contains multiple functional classes, the recall measure may dominate the calculation of F-score if the fitness function treats the entire dataset as a single class. This will mislead the system to find overgeneral elements shared by most sequences. To alleviate the bias, we define a new measure of recall, and present the fitness function as below, where p is the number of positive examples containing $motif_i$, Q is the total number of positive examples, R is the total number of examples containing $motif_i$, and U is the upper bound of the hypothesized range for cluster size.

$$Fitness(motif_i) = \frac{2 \times recall(motif_i) \times precision(motif_i)}{recall(motif_i) + precision(motif_i)}, \qquad \text{(Eq. 1.3.1.1)}$$

where $recall(motif_i) = p/Q$, if $p < U$ or $recall(motif_i) = 1$, if $p > U$. The value of $precision(motif_i) = p/R$. By taking cluster size into account, we can better constrain the search space and allow conserved clusters to emerge more likely instead of being buried in bigger but much less coherent clusters.

Consensus Structure Specificity and Separate-and-Conquer Strategy

The GP (Genetic Programming)-based structure prediction method can find the fittest secondary structure elements according to a given range of the cluster size, while the significance of the cluster found along with its characteristic structure elements highly depends on the range we choose. With proper adjustment of cluster size through the generate-and-test procedure combined with the GP-based prediction method, we can identify a meaningful cluster and the associated characteristic structure elements. The adaptive adjustment of cluster size in the generate-and-test procedure is controlled by the consensus structure specificity. It is defined as the Laplace prior precision. The Laplace prior approach has also been applied to inductive leaning to evaluate the significance of inductive rules [24]. The Laplace prior precision of cluster C_i is given by the formula:

$LaplacePriorPrecision(C_i)$ = (number of positive examples in C_i +1) / (total number of examples in C_i +2), (Eq. 1.3.1.2)

We consider the Laplace prior in the calculation of precision with the aim to avoid well

conserved clusters whose size is too small. For example, the Laplace prior precision of a cluster of 50 positive examples and five negative examples is better than that of a cluster of only five positive examples. Note that the Laplace prior precision is only used to determine the significance of a cluster found, unlike the F-score, which is used to direct the optimization process to find the best structure elements under the constraints of the cluster size. Based on the comparison of the Laplace prior precision with a pre-specified threshold, we adjust the range of cluster size accordingly, and then re-run the GP-based method to predict new structure elements and a new cluster they characterize.

Once a significant cluster is found, we separate all its members out of the given dataset of RNA sequences. We then apply the same procedure to those that still remain in the dataset until the entire set is emptied. This separateand-conquer strategy is effective when no prior knowledge of the identities of the clusters is given. It can automatically partition the given dataset into meaningful clusters, and also identify their characteristic structure elements.

Second Year:

1.3.2. A Multi-strategy Approach to Protein Structural Alphabet Design

The use of frequent local structural motifs embedded in polypeptide backbone has recently shown improvement in protein structure prediction [25-27]. Its success has shed some light on further studies of structural alphabet. We used the proteins classified to all-α fold within the SCOP database (version 1.65) in our study with the aim to build the structural alphabet suitable for all-α proteins. The same approach can be easily applied to other databanks as well.

There are three issues addressed in our study. They are: (1) protein fragment representation, (2) alphabet size determination and (3) structural alphabet definition. Like others, we transform each protein backbone into a series of the dihedral angles (φ and ϕ, neglecting ω) [26, 28]. Adapted from [28], the analysis is limited to fragments of five residues since they are adequate for describing a short α helix and a minimal β structure. With the fixed window size of five residues, we slid the window along each all-α protein in SCOP, advancing one position in the sequence for each fragment, and collected a set of overlapped 5-residue fragments. As the relation between two successive carbons, $C\alpha_i$ and $C\alpha_{i+1}$, located at the ith and $(i+1)$th positions, can be defined by the dihedral angles ψ_i of $C\alpha_i$ and φ_{i+1} of $C\alpha_{i+1}$, a fragment of L residues can then be defined as a vector of $2(L-1)$ elements. Thus, in our study, each protein fragment, associated with α-carbons $C\alpha_{i-2}$, $C\alpha_{i-1}$, $C\alpha_i$, $C\alpha_{i+1}$ and $C\alpha_{i+2}$, is represented by a vector of eight dihedral angles, i.e. [ψ_{i-2}, φ_{i-1}, ψ_{i-1}, φ_i, ψ_i, φ_{i+1}, ψ_{i+1}, φ_{i+2}]. Based on this representation, we totally gathered 1,143,072 fragment vectors.
Self-organizing maps (SOM) are widely used as a data mining and visualization tool for

complex data sets. A self-organizing map usually consists of a regular 2D grid of so-called map units, each of which is described by a reference vector $m_i = [m_{i1}, m_{i2}, m_{i3},\ldots, m_{id}]$, where d is the input vector dimension, e.g., $d = 8$, in our case of fragment vectors. The map units are usually arranged in a rectangular or hexagonal configuration. The number of units affects the generalization capabilities of the SOM, and thus is often specified by the researcher/user. It can vary from a few dozen to several thousands. An SOM is a mapping from the ensemble of input data vectors (X_i=$[x_{i1}, x_{i2}, x_{i3},\ldots, x_{id}] \in R^d$) to a 2D array of map units. During training, data points near each other in input space are mapped onto nearby map units to preserve the topology of the input space [29, 30]. The SOM is trained iteratively. In each training step t, distances between a randomly picked input vector x_j and all the reference vectors are computed. The unit with the least distance is then selected as the winner unit and denoted by w. The winner unit and its topological neighbors are updated to move closer to input vector x_j in the input space by the following rule:

$$m_i(t+1) = m_i(t) + \alpha(t) h_{wi}(t) \,|\, x_j - m_i(t) \,|, \qquad \text{(Eq. 1.3.2.1)}$$

where t is time, $\alpha(t)$ is the adaptation coefficient, $|x_j\text{-}m_i(t)|$ is the component-wise difference between the input vector and the ith reference vector, and $h_{wi}(t)$ is the neighborhood function acting on the array of units, whose form includes bubble kernel, Gaussian kernel and other more complicated ones. In our study, we used the bubble kernel [29, 31]. Unlike previous works that directly apply SOM to obtain clusters of backbone fragments as the basis to define the structural alphabet, our approach instead uses SOM only for the visualization purpose to predetermine the number of letters in the alphabet.

By visual inspection of the trained SOM, we can get a preliminary idea of the number of clusters on the map. The unified distance matrix (U-matrix) is one of the most widely used methods for visualizing the clustering result on the SOM. It shows distances between neighboring reference vectors, and can be efficiently visualized using grey shade [32], as shown in Figure 1.3.2.1(a). In spite of the initial idea of the cluster structure provided by the U-matrix, a systematic method to determine the number of clusters on the map is still desired. We implement a post-process on the Umatrix that is based on the minimum-spanning-tree algorithm. Given the grey levels in the U-matrix, we can build the minimum spanning tree for all the map units, e.g., in Figure 1.3.2.1(b), all map unit are linked in the spanning tree. Based on a threshold of the grey level, we can partition the entire tree into several disconnected subtrees, by removing the links between map units with grey levels below the threshold, as shown in Figure 1.3.2.1(c). Conceptually, it means that we break the links of a distance longer than some threshold. Furthermore, those relatively smaller subtrees left can be also deleted later such that the remaining clusters can maintain a reasonable size, as presented in Figure 1.3.2.1(d). The number of the subtrees finally kept becomes the structural alphabet size. As the SOM can be viewed as a topology preserving mapping from input space onto the 2D grid of map units [30], the number of map units can affect the clustering result. We systematically increase the number of units, and repeat the above process till the alphabet size stabilizes.

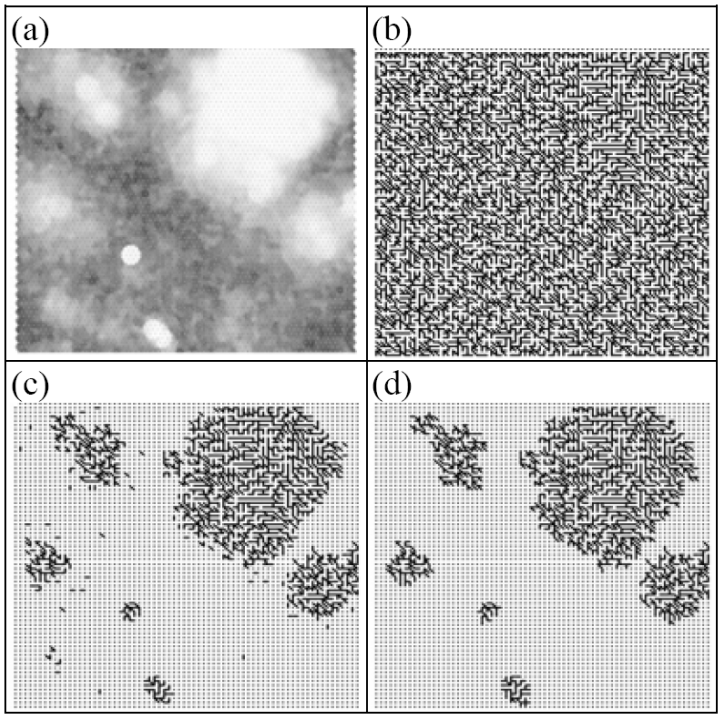

Figure 1.3.2.1. Visualization of the trained SOM. (a) the grey shade of the trained SOM, where darker areas mean larger distances, **(b)** the minimum spanning tree for the map units, **(c)** the disconnected subtrees after removing the links below some threshold and **(d)** the final disconnected subtrees after discarding those relatively small ones.

Rather than adapt the two-level approach that first trains the SOM, then performs clustering of the trained SOM [30], after determining the alphabet size, we apply the k-means algorithm to the input data vectors directly to obtain the clusters. The SOM established a local order among the set of reference vectors in such a way that the closeness between two reference vectors in the R^d space is dependent on how close the corresponding map units are in the 2D array. Nevertheless, an inductive bias of this kind may not be appropriate for structural alphabets since the local order does not always faithfully characterize the relation between structural building blocks, and can sometimes be misleading, e.g. forcing the topology to preserve mapping from the input space of α-helix and β-strand to a 2D grid of units could be harmful to clustering. As a result, we use the SOM only for visualization the alphabet size, and rely on the k-mean algorithm to extract the local features from the input data that can actually reflect the characteristics of the clusters respectively. The centroid of each cluster forms the prototypical representation of each alphabet letter. Given the clustering result by the k-means algorithm as the basis of the structural alphabet, we can transform a protein into a series of the alphabet letters by matching each of its fragments against our alphabet prototypes. The control flow of our system named SMK is illustrated in Figure 1.3.2.2.

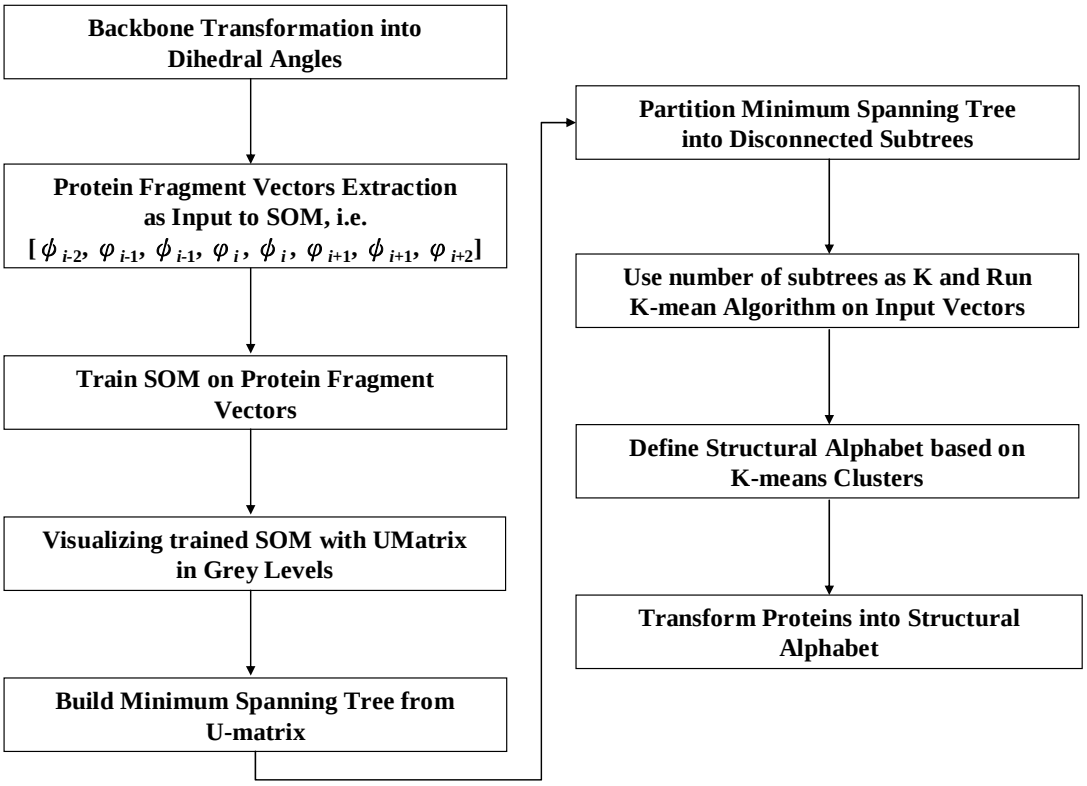

Figure 1.3.2.2. The system control flow of SMK

1.3.3. Bicluster Analysis of Genome-Wide Gene Expression

As the advent of microarray technologies, numerous datasets generated by massive microarray experiments have drawn a lot of attention to the need for efficient and effective computational methods for gene expression data analysis [33-35]. In general, expression datasets are described by 2-D arrays. One axis represents the genes; the other, the conditions. Each element in the array records the expression level of a gene as a real number, which is usually derived by taking the logarithm of the relative abundance of the mRNA of that gene in a specific condition. Genes with compatible expression patterns are believed to be under identical or related regulatory control. Given appropriate gene clusters, there can be many further applications based on expression behaviors when combined with other biological information such as subcellular localizations, metabolic pathways and intermolecular interactions, and so on [36]. This demonstrates the importance of the finding of expression clusters.

Clustering can be applied to either genes or conditions to obtain expression clusters, i.e., grouping genes according to all conditions, or grouping conditions according to all genes, separately. The most common clustering algorithms include hierarchical clustering [37], self-organizing maps (SOM) [32] and k-means clustering [38], etc. Nevertheless, most of

them only consider global similarity between expression profiles or between condition samples, thereby missing local relationships. They typically assume that functionally related genes behave similarly over all measured conditions, and the conserved condition patterns run across all measured genes. Those clusters found are in a sense aimed to reflect a global pattern of expression data, and yet for most cases in the real cellular processes, expression patterns are common only to a subset of genes under certain experimental conditions [39]. In order to characterize the expression behaviors more accurately, we need a local model instead of a global one. Identifying such local patterns will provide a deeper insight into genetic pathways that cannot be revealed from the point of a global view. As a result, our objective is to develop, beyond the common clustering paradigm, a method capable of discovering in the microarray data local expression patterns in terms of submatrices which we call biclusters. The rows and columns of a submatrix correspond to a subset of genes with similar expression behaviors under a certain subset of conditions, respectively.

condition / gene	…	hir2	Rad27	nrf1	rtg1	hpt1	pau2	…
YAR002C-A	…	2	3	3	4	9	9	…
YBL084C	…	2	3	3	4	7	8	…
YBR138C	…	9	8	8	7	5	4	…
YBR141C	…	9	8	8	7	5	4	…
YCR028C-A	…	1	2	8	7	5	4	…
…	…	…	…	…	…	…	…	…

Figure 1.3.3.1. A sample of biclusters. A sample gene expression result is represented as a 2D array. In this array, we show two overlapping biclusters that are formed by different subsets of genes and conditions

For example, in Figure 1.3.3.1 we show a sample gene expression result represented by a 2D array, where the rows stand for different genes; the columns, various conditions. In this array, there are two submatrices each of which consists of different subsets of genes and conditions.

Several methods from various research fields have been proposed to perform clustering which take into account rows and columns at the same time. Though named differently, such as biclustering, local clustering, coclustering, direct clustering, bidimensional clustering, or block clustering, etc. [40-42], they all refer to the same class of algorithms which identify subsets of rows and subsets of columns by performing simultaneous clustering of these two dimensions. These methods can be further categorized according to the types of biclusters they intend to identify. The categories of biclusters may vary from the simplest one which contains only constant gene expression levels, or those with constant values in the rows or the columns, to more complicated ones which contain coherent expression levels with low variance, or others that only maintain a coherent expression trend without considering

specific expression values [43]. It is inevitable that there exists a tradeoff between the complexity of search strategy and the expressiveness of the biclusters identified. The type of biclusters to find determines the complexity of search strategies applied by the biclustering algorithms. The approaches previously proposed include greedy iterative search [40], divide-and-conquer [41], exhaustive enumeration [44] and probabilistic modeling [45], etc. In this study, we propose a new approach called PIFP (Progressive Iterative Frequent Pattern) to biclustering based on frequent itemset identification [46], which is not aimed at resolving all the issues in the biclustering problem, but instead at demonstrating the feasibility of tackling the problem from a different perspective. To fairly compare biclustering methods is not easy as each approach may formulate the same problem differently, and consequently applies different algorithms. Due to the inherent bias, their performance may vary in different scenarios. Recently, a systematic comparison methodology for biclustering has been proposed, which defines the common settings for most of the biclustering approaches [47]. Based on this testing methodology, we conducted the validation using prior knowledge, i.e. gene ontology. Several conventional clustering methods and some current biclustering systems were also tested for comparison in our studies. The conventional ones are hierarchical clustering [37], k-means [38], self-organizing maps (SOM) [32] and principle component analysis (PCA) [48]; the biclustering algorithms include OPSM [39], ISA [49], CC [40], xMotifs [50], BiMax [47] and SAMBA [49]. We demonstrate that our new approach outperformed these systems in the experiments.

Unlike previous approaches, we transform the biclustering problem into a frequent itemset finding task, which is a welldefined activity in market basket analysis. To illustrate the idea, let us first consider a market basket containing a collection of items purchased by a customer in a single transaction. Retailers usually accumulate huge collections of transactions over time, and one common analysis run against a transaction database is to find sets of items (or *itemsets*) that appear together in many transactions. We call an itemset a frequent itemset if the percentage of the transactions that contain this itemset, which is called *support*, is above a userspecified threshold. As the items in a frequent itemset co-occur in many transactions, some association is expected among them. Motivated by this observation, we characterize a bicluster containing related genes and relevant conditions by a frequent itemset where genes or conditions are treated as items, and the conserved expression behaviors correspond to the associations.

In microarray experiments, there are different sources of systematic errors [51]. Normalization refers to the attempts to remove such error that affect the measured expression levels. Although normalization alone cannot control all the systematic variations, it is still crucial to subsequent expression data analyses as the expression data may vary significantly from different normalization processes. We tested several normalization methods, including geometric mean normalization, rank normalization, quintile normalization and Spellman et

al.'s Z-score [52, 53]. As some pilot tests favored the Z-score approach, we decided to adopt the normalization procedure of Spellman *et al.*, who normalized the expression level of each gene so that the mean and the deviation for each row and column are zero and one respectively. In addition to normalization, discretization is another preprocess required of PIFP as a bicluster is described by a frequent itemset. We partitioned numeric gene expression values into a set of intervals; as a result, the original expression data matrix can be viewed as a database of transactions. Several discretization procedures were tried to define intervals [54, 55]. The intervals may be determined by some specific thresholds, such as the median, the top *N*% expression value, or even pre-specified ad hoc values. On the other hand, we can partition expression levels into *J* intervals of equal size, e.g. *J*=10. It is usually difficult to define appropriate thresholds beforehand, and our pilot tests showed that the equal-interval partition method worked reasonably well, thus in the following experiments we adopted this discretization procedure with *J* set to 10.

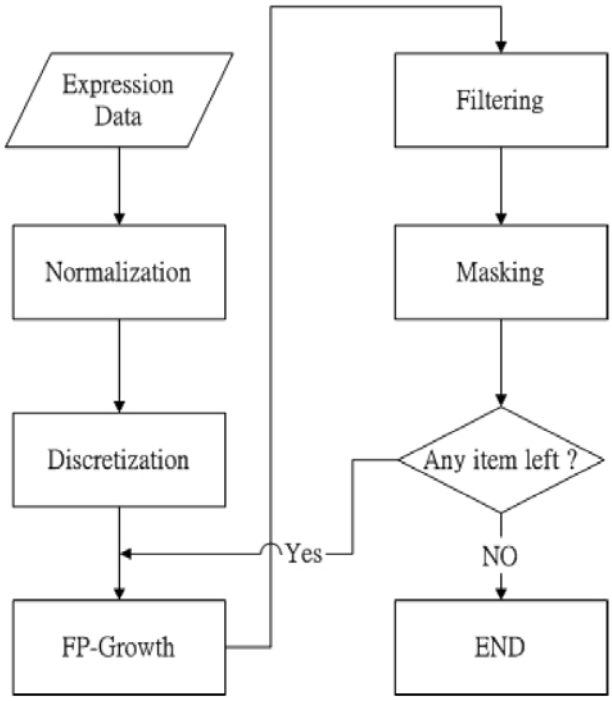

Figure 1.3.3.2. The PIFP system flow, in which we added the filtering and masking procedures to rule out spurious biclusters and the ones already found. These procedures have proved effective in producing more biclusters than the original FP-growth method.

There have been many various approaches to frequent itemset finding, and one of the widely used methods is FPgrowth [46]. Unlike its relevant pioneer approaches, e.g. Apriori [55, 56], FP-growth only needs to scan the database twice without the time-consuming candidate-generation process. The first scan of a database derives an ordered list of frequent itemsets above some pre-specified support threshold. For the second scan, FP-growth performs a database projection of the frequent itemsets in the order that they are in the ordered list, and constructs a compact data structure called FP-tree, which stores the complete

frequent itemset information. This significant reduction on the number of necessary database scans dramatically increases its efficiency. We developed PIFP based on FP-growth. It extends FPgrowth in two directions. First, we embed a filter in PIFP to rule out those spurious itemsets by removing any itemset of less support if it overlaps with others over 75%. Second, we add a feedback loop to PIFP in order to identify weaker (i.e. with less support) frequent itemsets which may have been clouded by other stronger frequent itemsets. By masking out from the expression matrix the itemsets already found, PIFP is able to progressively identify more frequent itemsets than conventional FP-growth approaches. Our ablation experiment in the following section proves the effectiveness of these extensions. Given only two parameter values, minimum support *s* and minimum itemset size *i*, PIFP is capable of identifying overlapped biclusters presented as frequent itemsets. The system flow of PIFP is presented in Figure 1.3.3.2. A sample output of a bicluster is shown in Fig. 1.3.3.3. It contains information such as the total number of genes in this bicluster, who they are (e.g. gene or ORF names), the positive-correlated genes, the negative-correlated genes, and the corresponding conditions, etc.

Bicluster 18:	4 Genes
No. of Conditions:	3
TAF17=1, alpha84=4, SWI2=10	
No. of Positive Genes:	3
YAL009W, YAL037W, YAR053W	
No. of Negative Genes:	1
YAR009C	

Figure 1.3.3.3. A sample output of PIFP. It shows there are 4 genes in bicluster18, and three conditions involved. Three of the genes are positive-correlated, and only one is negative-correlated.

Third Year:

1.3.4. A Two-stage Approach to Finding Common Structure Elements in Unaligned RNA Sequences

Given functionally related RNA sequences, there are currently three main approaches to the finding of common secondary structures [57]. The first approach aligns sequences using standard multiple sequence alignment tools, e.g. ClustalW [58], and then detects consensus

secondary structures based on mutual information, free energy or sequence covariance [4, 59, 60] etc. However this approach strongly depends on a reliable multiple sequence alignment. It is not suitable for RNAs with low sequence similarity. An alternative approach is to fold sequences and align structures at the same time. Though this approach can be applied in the case of unavailability of multiple sequence alignment, its high computational complexity restricts its practical use [61-63]. If there is no enough sequence conservation, and the complexity of structure motifs exceeds the pragmatic limit of the above approaches, we may take the last approach. It first predicts the secondary structure for each sequence, and then aligns the structures directly [16, 64, 65].

There are several important features in our method. First, it is applicable to unaligned RNA sequences with long flanking regions and low sequence similarity. Second, it has flexibility in incorporating new tools for single RNA global structure prediction in the first stage. Third, secondary structures predicted in the first stage are transformed to an abstract form that helps constrain the search space of consensus motifs.

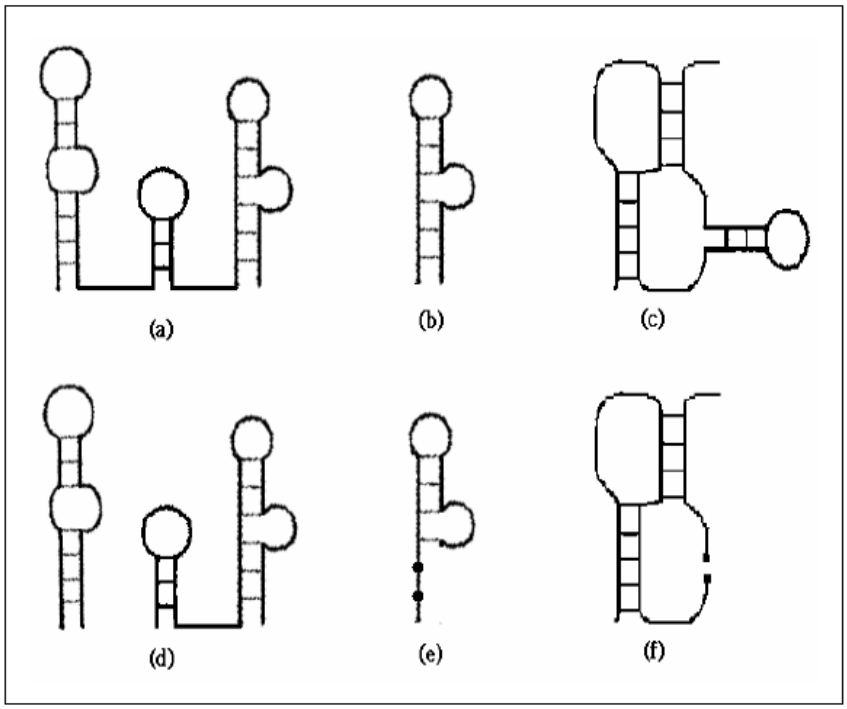

Figure 1.3.4.1. Positive and negative examples of structure motifs. (a)(b)(c) are legal structure motifs. Each satisfies all three conditions of a legal structure motif. **(d)(e)(f)** are the negative examples relative to **(a)(b)(c)** respectively, where (d)(f) are not continuous structures, and (e) has an unpaired segment which is supposed to be a stem.

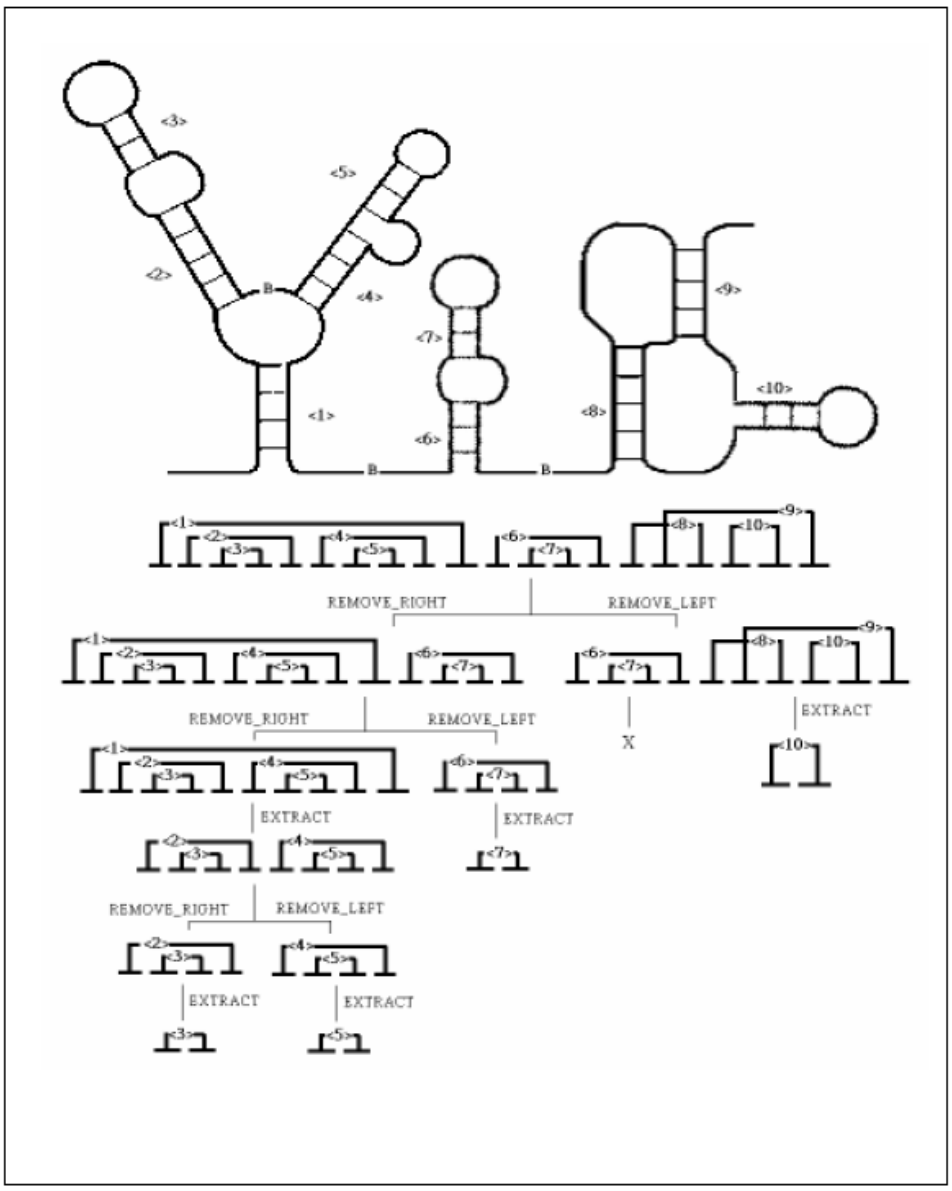

Figure 1.3.4.2. An example of structure decomposition. The given structure (stem <1> to <10>) is not a component motif as it has several bifurcation sites, marked by "B". We first remove its rightmost and leftmost component respectively to produce two legal substructures, one composed of stem <1> to <7>; the other, <6> to <10>. We later generate two substructures from stem <1> to <7>. They are stem <1> to <5> and stem <6> to <7>. These two structures are component motifs since they have no bifurcation site, and smaller legal substructures can be extracted from within, e.g., stem <2> to <5> extracted from structure <1> to <5>. We keep decomposing a structure until we reach its basic singlestem component, e.g. <3>, <5> and <7>. On the other branch, stem <6> and <7> are found to already occur in the left branch, therefore, they are considered redundant and will not be further processed. Stem <8> to <10> is also a component motif which is a pseudoknot containing stem <10>, the last legal substructure extracted.

Our proposed method adopts the last approach, but the objective of our system is to find consensus structure motifs within a set of RNAs rather than a multiple global structure alignment. We define a legal structure motif for an RNA family as a commonly shared

structure: **(1)** that is folded from continuous nucleotides, **(2)** that begins with a 5'segment of a stem, and ends with a 3'segment which may be the half of another stem or paired with the first 5'segment, and **(3)** that has no unpaired 5'or 3'segment between the first 5'segment and the last 3'segment. Some positive and negative examples are shown in Figure 1.3.4.1. For each predicted structure from a folding algorithm, we exhaustively decompose it and enumerate all the possible substructures that comply with the three constraints above. A legal structure motif can be further defined as a component motif if it satisfies all three constraints above, and cannot be broken into smaller legal structure motifs, *e.g.* Figure 1.3.4.1(b) and (c). For a structure with only one component, we can discard either the outermost paired segments or pseudoknots to extract a smaller legal substructure by EXTRACT. On the other hand, given a structure with more than one component, we can divide it into two legal substructures by removing the leftmost (i.e. REMOVE_LEFT) or the rightmost component (i.e. REMOVE_RIGHT). One example of structure decomposition is illustrated in Figure 1.3.4.2. By applying EXTRACT and REMOVE recursively when applicable, we can decompose any given structure and enumerate all its possible legal substructures that will be later transformed into the search space of consensus motifs.

The occurrences of a consensus motif in a family are rarely the same in every detail of their structures. For example, the size of stems or loops may vary among motif occurrences in different family members, and some may even contain extra bulges or internal loops, e.g. X83878.1/168-267 and Z99107.2/86084-86183 of Purine riboswitches in Rfam with their secondary structure motifs shown in Figure 1.3.4.3 (a). According to the alignment, the first stem of Z99107.2/86084-86183 has a symmetric bulge (small internal loop) consisting of CUCA, which does not exist in X83878.1/168-267. Besides, X83878.1/168-267 has a smaller symmetric bulge in the second stem and a shorter third stem when compared with Z99107.2/86084-86183, while it has longer first and second stems. To accommodate these minor differences between motif patterns (i.e. motif occurrences) in a family, we represent the decomposed substructures by an abstract shape [64], which corresponds to the common secondary structure, as illustrated in Figure 1.3.4.3 (b). By abstraction, these abstract shapes form a much smaller search space where we can find consensus motifs more efficiently.

Figure 1.3.4.3. Motif structure patterns in two members of the Purine riboswitch family in Rfam and the abstract shape for the motifs. (a) Consensus secondary structure between X83878.1/168-267 and Z99107.2/86084-86183 of Purine riboswitches in Rfam. (b) Ignoring the minor differences (*e.g.* different sizes of stems, loops or bulges) and focusing on the common relationship among the stems, we can represent the consensus structures by one abstract shape.

Gibbs sampling is one of the MCMC (Markov Chain Monte Carlo) algorithms. In Gibbs sampling, we iteratively sample each variable conditioned on the most recent values of the other variables. Starting with a set of initial values of all the variables, we cycle through the sampling process for each single variable in any order until the values of all the variables converge to a stable state. Under this framework, the given RNAs are the variables, and the motif patterns are treated as their values. Our goal here is, by Gibbs sampling, to find the appropriate value (i.e. motif pattern) for each variable (i.e. RNA) such that the values can reach a stable distribution which corresponds to a consensus structure motif.

We define the similarity between two structure patterns, s_i and s_j, as the following:

if $RLD(s_i, s_j) > \theta_L$, $sim(s_i, s_j) = 0$; (Eq. 1.3.4.1)

otherwise $sim(s_i, s_j) = w_1 * seqalign(s_i, s_j) + w_2 * structalign(s_i, s_j)$. (Eq. 1.3.4.2)

$RLD(s_i, s_j) = |\, len(s_i) - len(s_j) \,| \,/\, max(\, len(s_i), len(s_j) \,)$. (Eq. 1.3.4.3)

where RLD(si , sj) is the relative length difference between si and sj, len(sk) is the sequence length of structure sk. The seqalign(si , sj) is the sequence alignment score based on the Needleman-Wunsch algorithm [66], assuming match=1, mismatch=0 and gap=-1, and structalign(si,sj) is the structure alignment score computed by RSmatch [67]. Both seqalign(si,sj) and structalign(si,sj) are normalized between zero and one. Note that we assign zero to sim(si,sj) directly when RLD(si,sj) is greater than θ L to save the time for the computation-intensive alignment procedures. The motivation behind this is our observation of most families in Rfam shows that the relative length difference between family members is usually insignificant, which makes it an effective filter. In eq.(1.3.4.2), sim(si,sj) is computed as the weighted sum of the sequence and structure alignment scores, where w1+w2=1.

The Gibbs sampling process in our system starts with an initial state of a consensus motif represented by a set of seeds, SEED, each of which is a possible occurrence of the motif in a particular RNA sequence. In each iteration, we sample the motif patterns for one RNA, e.g. R, conditioned on the currently selected motif occurrences in the others, and a structure pattern pi $\in$ R will be chosen as a new seed (i.e. a new motif occurrence) if it satisfies either of the following conditions.

If R does not currently have a seed in SEED, then

$$p_i = \arg\max_{p_j} 1/|\,SEED\,| \cdot \sum_{s_k \in SEED} sim(p_j, s_k) \quad \text{(Eq. 1.3.4.4)}$$

under the constraint that

$$1/|\,SEED\,| \cdot \sum_{s_k \in SEED} sim(p_i, s_k) > \theta_s$$

If *R* already has a seed in *SEED*, then

$$p_i = \arg\max_{p_j} 1/(|\,SEED\,| - 1) \cdot \sum_{s_k \in SEED, s_k \notin R} sim(p_j, s_k) \quad \text{(Eq. 1.3.4.5)}$$

under the constraint that

$$1/(|\,SEED\,| - 1) \cdot \sum_{s_k \in SEED, s_k \notin R} sim(p_i, s_k) > \theta_s$$

As we iterate over every RNA, we can either add new patterns as new motif occurrences when the above condition is satisfied, or delete old seeds from the seed set if they no longer meet the constraint. We update the set *SEED* with the aim to increase the total pairwise pattern similarity $sim_{total}(SEED)$ defined below. We repeat the same sampling process until no change of motif occurrences can be made to improve $sim_{total}(SEED)$.

$$sim_{total}(SEED) = \sum_{i \neq j} sim(s_i, s_j), \forall s_i, s_j \in SEED, \quad \text{(Eq. 1.3.4.6)}$$

The initial seeds determine where and how fast Gibbs sampling converges, and the size of the initial seed set does not need to be equal to the total number of the RNAs given. Since we can start Gibbs sampling with different initial seeds, it can terminate at various sets of final seeds. When Gibbs sampling stops after it converges, the size of converged *SEED* will ideally be equal or approximate to that of the given RNA family, and the seeds *per se* are the predicted occurrences of a consensus motif. According to sim_{total}, we rank all the motifs to which Gibbs sampling converges, and report them in a sorted list after the user specifies the number of top-ranked motifs required in the output.

1.4 Result

First Year:

1.4.1. RNA Clustering and Secondary Structure Prediction

Two types of quality were considered to evaluate the performance of our method. One is to measure the agreement between the predicted clusters and the actual cluster identities; the other, to quantify the agreement between the predicted structure elements and the actual structure assignment. Since no other current approaches known to perform clustering and structure prediction in parallel, no comparative study can be done. Instead we applied the widely-used precision and recall to measure the first quality; the Matthews correlation coefficient [68], to measure the second quality.

For each sequence in the data set, two secondary structure assignments were compared by counting the number of true positives P_t (base pairs exist in actual assignment and are predicted), N_t true negatives (base pairs do not exist in actual assignment and are not predicted), false positives P_f (base pairs do not exist in actual assignment but are predicted) and N_f false negatives (base pairs exist in actual assignment but are not predicted), respectively. The Matthews correlation coefficient can then be computed as:

$$C = \frac{P_t N_t - P_f N_f}{\sqrt{(N_t + N_f)(N_t + P_f)(P_t + N_f)(P_t + P_f)}} \qquad \text{(Eq. 1.4.1.1)}$$

Given that the sequence length is sufficiently large, the Matthews correlation coefficient can be approximated in the following way [69].

$$C \approx \sqrt{\frac{P_t}{P_t + N_f} \cdot \frac{P_t}{P_t + P_f}} \qquad \text{(Eq. 1.4.1.1)}$$

With the published/curated alignments, we can calculate the Matthews correlation coefficient. Higher correlation coefficients mean more accurate structure predictions.

Table 1.4.1.1. Summary of the RNA families used in experiments. The first row shows the total number of sequences in each data set. Row 2 to 4 present the minimum, the maximum and the average sequence length respectively. The fifth row gives the standard deviation of sequence length.

Dataset	16s RNA	IRE-like	Viral 3'UTR
Total Sequences	34	56	18
Min Seq Length	90	117	37
Max Seq Length	108	330	137
Avg Seq Length	97.59	202.93	63.89
Seq Length Std	3.77	59.31	25.95

Our algorithm is designed to automatically partition a given set of unaligned RNA sequences into meaningful clusters, each with characteristic conserved secondary structure elements. The number of real clusters and the distribution of cluster size may affect the prediction of partitions and characteristic structure elements. To measure their effect on the performance, we tested our method on different datasets with various RNA families. We used three families, including 16S RNA, IRE (Iron Response Element) and viral 3'UTR as summarized in Table 1.4.1.1, to prepare the test datasets. They have been used in previous experiments and published in literature [10, 69]. The sequence data and the correct structure elements can be accessed at public databases [70, 71]. The 16S RNA dataset contains 34 archaea 16S ribosomal sequences originally derived from a set of 311 sequences extracted from the SSU rRNA database. The archaea set of 311 sequences was further reduced to 34, filtering out the sequences that miss base assignments or are greater than 90% identical. The IRE dataset was constructed by Gorodkin et al.[69] from 14 sequences from the UTR database. They modified the IREs and their UTRs to make the search more difficult. By iteratively shuffling the sequences and randomly adding one nucleotide to the IRE conserved region, they built a set of 56 IRE-like sequences from the 14 IRE UTRs. The third data set includes 18 viral 3'UTRs each of which contains a pseudoknot. Seven of the RNA sequences are the soil-borne rye mosaic viruses; the others are the soil-borne wheat mosaic viruses.

Table 1.4.1.2. **Summary of the experimental results.** Table (a), (b) and (c) present the result for the dataset containing IRE and viral 3'UTR, 16S RNA and viral 3'UTR, IRE and 16S RNA, respectively.

(a)

IRE+viral 3'UTR	Recall	Precision	Matthews
IRE	0.97	0.99	0.97
Viral 3'UTR	0.71	0.95	0.79

(b)

16s RNA+viral 3'UTR	Recall	Precision	Matthews
16s RNA	0.97	0.95	0.83
Viral 3'UTR	0.77	0.98	0.77

(c)

IRE+16s RNA	Recall	Precision	Matthews
IRE	0.73	0.99	0.85
16s RNA	0.81	0.73	0.67

On the basis of the three real families of RNA sequences, we tested our method on each possible pair of the families, i.e. 16S RNA/IRE, 16S RNA/viral 3'UTR, and IRE/viral 3'UTR. In each run of the experiment, no information regarding the number of families or the family size was given to the algorithm beforehand. One purpose of this experiment is to analyze the effect incurred by the distribution of cluster size in a dataset. Furthermore, as the real conserved structure elements differ in various families, we can also observe how the interleaving of distinct structure motifs within a single dataset may affect the prediction process. The results are presented in Table 1.4.1.2, and some partial predicted secondary structures are shown in Figure 1.4.1.1.

```
***** IRE *****

> seq_D15071.1
  41     45   47      51           58     62 63      67
t g c g g u c c u g g c c a g u g a g c u g g g c c g c

predicted:
. ( ( ( ( ( . ( ( ( ( ( . . . . . ) ) ) ) ) ) ) ) ) )

published:
. ( ( ( ( ( . ( ( ( ( ( . . . . . ) ) ) ) ) ) ) ) ) )

***** 16S RNA *****

> U51469
 13     20 23      31    37       46   52       61    67      75 78     85
guuucauugaaguuugcuuuuagugaggugacgucuaauuggcguuaucgaacuugugguaagcgacaagggaaaa

predicted:
.(((((((..(((((((((.....((((((((((.....)))))))))).....)))))))))..)))))))..

published:
.(((((((..((((((((((((..(((((((((.....)))))))))..))))))))))))..)))))))..

***** viral 3'UTR *****

> PKB183

  14  16  18         24 25  27        32          38
a c g u c g u g c a g u a c g g u a a a c u g c a c a u

predicted:
. ( ( ( . [ [ [ [ [ [ [ ) ) ) . . . . ] ] ] ] ] ] ] . .

published:
. ( ( ( . [ [ [ [ [ [ [ ) ) ) . . . . ] ] ] ] ] ] ] . .
```

Figure 1.4.1.1. **A partial result of the predicted RNA motifs.** The numbers above the sequences are the indices of the nucleotides. The predicted and the published motifs are both shown for reference.

Second Year:

1.4.2. A Multi-strategy Approach to Protein Structural Alphabet Design

We tested our approach on the all-α proteins in SCOP. By this experiment, we show that our method can produce an appropriate structural alphabet for describing these all-α proteins. After transforming protein backbones into dihedral angles and extracting protein fragments, we trained the SOM on these dihedral angle vectors.

Three issues were addressed in the experiments. First, the meaningfulness of the structural alphabet size in terms of the number of clusters was presented by showing the size stability given various parameters. Second, we demonstrated cluster cohesiveness by visual superimpositions of protein fragments as well as computed the intra-cluster and inter-cluster distance. Third, we proved the fragment clusters found were not arbitrary by comparing our result with that from a random background.

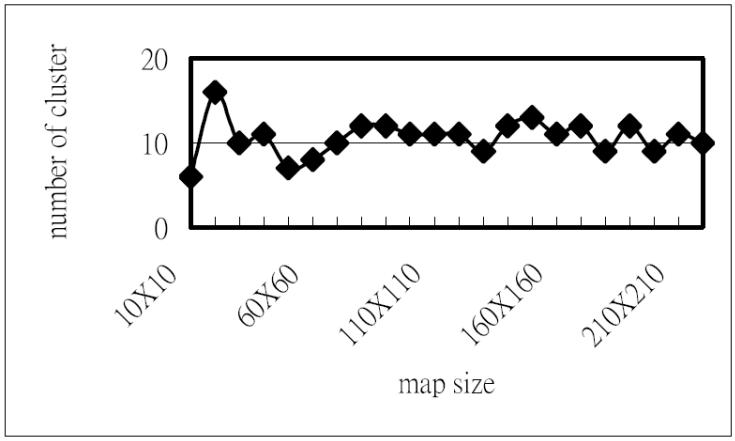

Figure 1.4.2.1. The variance in the number of clusters produced by the SOMs of varying sizes. There exists a distinctive plateau that suggests the cluster number has stabilized.

Since the number of map units has influence over the SOM's clustering behavior, to obtain the optimal number of clusters, we varied the number of units on the map until the number of clusters found became steady. The results are shown in Figure 1.4.2.1, which indicates a distinctive plateau within the range between nine and twelve. Because eleven is the most frequent number of clusters on the plateau, as shown in Figure 1.4.2.2, it is designated as the structural alphabet size.

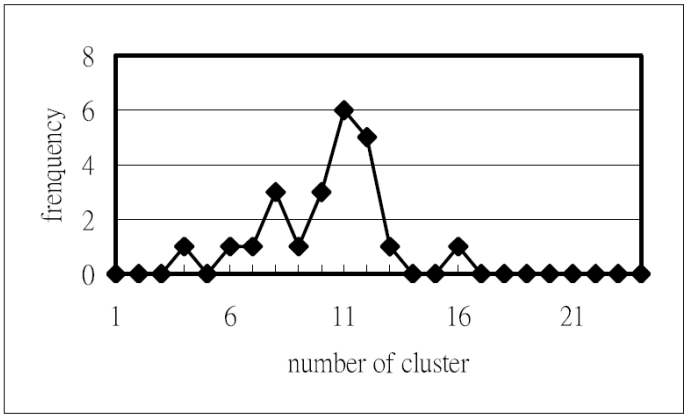

Figure 1.4.2.2. The frequencies of cluster numbers. It shows 11 is the most frequent number of clusters.

To further confirm the general geometric regularities characterized by the structural alphabet, we also built a negative all-α protein fragment set for comparison. The negative set was derived from the real all-α protein fragment vectors prepared earlier by rotating the dihedral angles at random (increase or decrease) within a certain degree, e.g. 30° in our analysis. We compared the clusters produced by clustering on the real vector set and on the negative control set. Insignificant difference suggests that the alphabet we found could be arbitrary. Our experiments (see Figure 1.4.2.3) show that clustering on the negative control

set cannot even produce consistent clusters, which supports our hypothesis that the clusters found from the real fragment vectors reflect the classes of local protein structures; otherwise, these clustering results would have been similar.

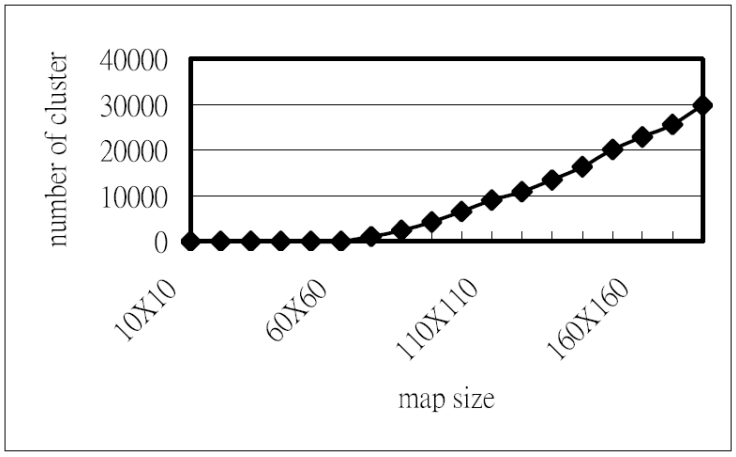

Figure 1.4.2.3. The variance in the number of clusters produced by the SOMs of varying sizes trained on a negative fragment set. It shows no sign of convergent cluster number.

Given the size, we ran the k-means algorithm on the input fragment vectors to find the twelve clusters by which to define the structural alphabet. Figure 1.4.2.4(a) and (b) shows the fragment superimpositions for the alphabet. Even though the fragment structures do not superimpose perfectly, yet the general structural cohesiveness of each category is quite evident. In addition, we computed the Euclidean distances from each fragment in a given cluster to its centroid. The average of these within-cluster distances was then compared with the center-to-center distances between clusters as presented in Table 1.4.2.1. It shows that in most cases, the center-to-center distance between any two clusters is greater than the mean distance of all vectors in that cluster from its center plus one standard deviation. The result indicates that the individual clusters are fairly well separated from each other.

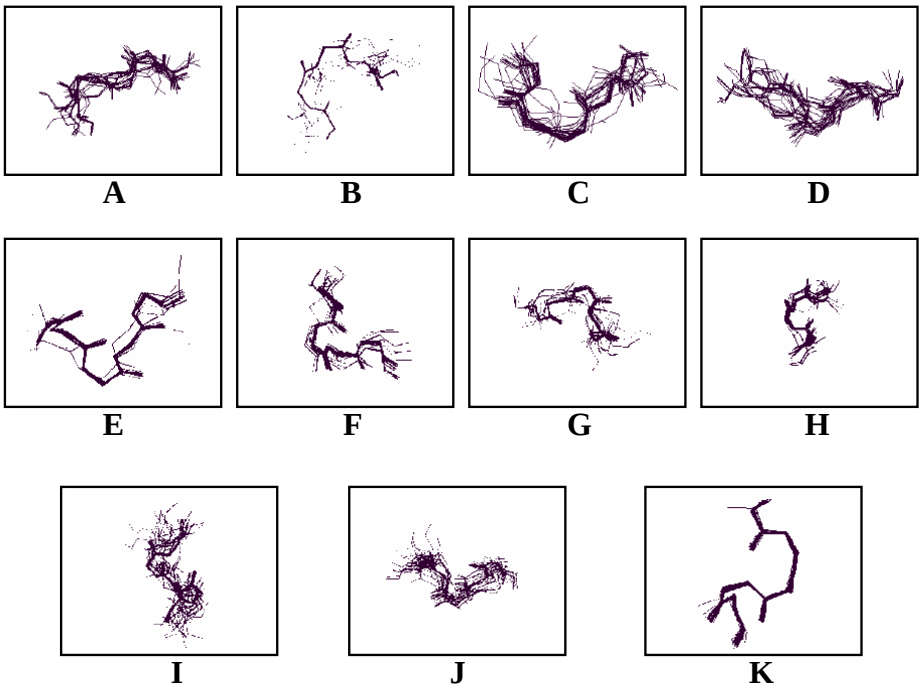

Figure 1.4.2.4(a). The superimposition in wireframe format for the structures of each structural cluster found by SMK.

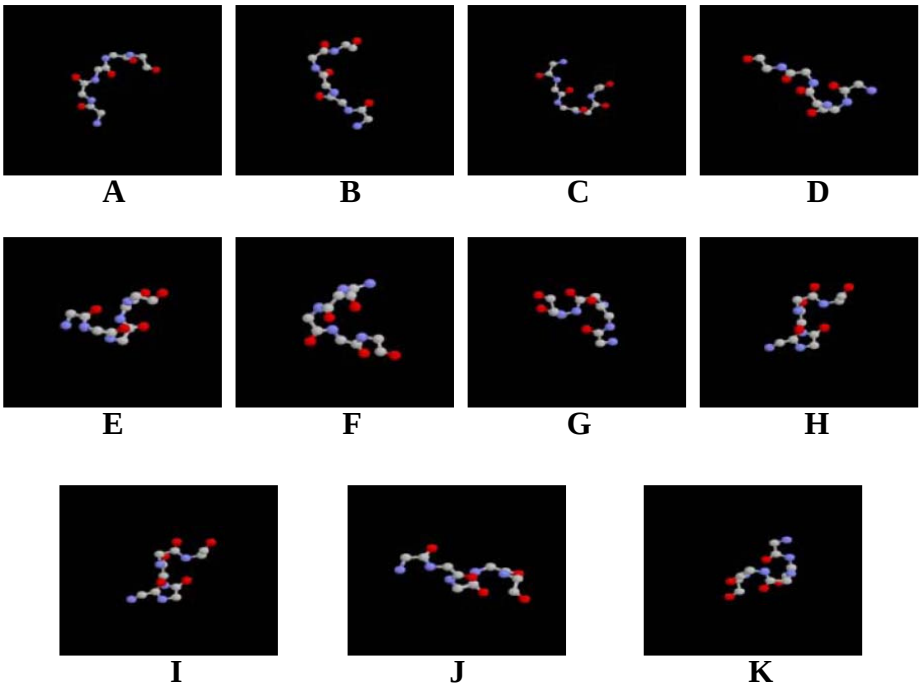

Figure 1.4.2.4(b). The superimposition of the structures of each structural cluster found by SMK in the balland-stick form.

Table 1.4.2.1. Summary of within-cluster distances and center-to-center distances.

	within-cluster	center-to-center									
	mean±sd	*A*	*B*	*C*	*D*	*E*	*F*	*G*	*H*	*I*	*J*	*K*
A	186.10±68.07	0	282.3	205.27	216.75	226.93	236.72	399.53	246.5	325.94	197.44	245.81
B	192.84±74.97		0	284.59	203.41	202.8	275.08	414.99	169.3	321.03	208.28	264.69
C	173.58±77.59			0	250.31	251.6	197.76	383.86	243.02	333.41	188	226.52
D	193.67±69.19				0	234.31	252.05	388.9	261.9	323.81	183.77	233.33
E	150.41±71.53					0	302.93	511.04	284.51	343.02	282.19	358.48
F	143.62±90.84						0	346.14	220.63	346.98	161.11	177.48
G	220.52±87.79							0	343.07	276.03	341.22	278.5
H	155.02±77.8								0	335.84	136.63	164.87
I	196.75±97.2									0	358.58	360.95
J	88.77±53.33										0	86.711
K	43.15±50.13											0

The detection and analysis of structural similarities between proteins allows deeper insight into their functional mechanisms and relationships. To search for structural similarities, the structural alphabet provides a good basis on which to work with a 1D representation. As a result, numerous 1D alignment algorithms can be used, with minor modifications, to detect structural similarities. In our experiments, we first transformed the 3D structures of proteins into a 1D sequence of the letters in our structural alphabet. To demonstrate the applicability of the alphabet, we used FASTA to search for structural similarities between a query protein and a bank of proteins, using an identify matrix of our alphabet letters to find maximal exact matches. For comparison, we also conducted the same tests also using FASTA but based on different structural alphabets, one developed by de Brevern *et al.* [72], the other by the two-level SOM approach [30]. As the baseline reference, we used BLAST with the standard 20 amino acid letters to find the best sequence hit.

Table 1.4.2.2. Summary of frequencies at the lowest common level. The first column shows the methods used in the experiments. The remaining columns present the frequency for different levels at which the query and the best hit are both located.

	frequency at different level			
Method	**class**	**fold**	**super family**	**Family**
BLAST	71	4	5	20
SMK	55	11	5	29
De Brevern	58	4	11	27
2-level SOM	73	6	14	7

Table 1.4.2.3. Summary of average RMSD and standard deviation between the queries and the best hits.

Method	mean (RMSD)	sd (RMSD)
BLAST	8.953744	4.764597
SMK	7.290972	3.934283
De Brevern	8.076746	4.819178
2-level SOM	10.38624	5.217078

The proteins used in the experiments were selected from the all-α proteins in SCOP. After filtering out those with more than 30% sequence similarity, we have totally 1055 proteins. For each run of the experiment, we randomly picked one protein as the query, and then matched it against the rest, using FASTA or BLAST with different alphabets. Given the best hit, we computed the RMSD between the query and the hit, and recorded the lowest level in the SCOP hierarchy at which the query and the hit are both located, i.e. class, fold, superfamily or family. Smaller RMSD and lower common level in SCOP hierarchy indicates higher structural similarity. We repeated the same experiment for 100 times and the results are summarized in Table 1.4.2.2 and Table 1.4.2.3. According to Table 1.4.2.2, we notice that our method SMK and de Brevern *et al.*'s both produced higher frequencies at lower common levels than the other two methods. This suggests that our structural alphabet and de Brevern *et al.*'s can better characterize the SCOP hierarchy. Table 1.4.2.3 shows that SMK has the lowest mean RMSD and standard deviation among all.

In this study, we propose a multi-strategy approach to designing the structural alphabet which allows local approximation of protein 3D structures as well as enables the applications of 1D alignment algorithms to search for 3D structural similarities. The success of the alphabet design depends on three crucial factors. First, it is the protein fragment representation, which determines what and how 3D structural characteristics to be approximated, e.g. thermodynamic stability, amino acid physicochemical properties, amino acid usage in known proteins, distances, dihedral angles, bond lengths, bond angles, etc. The effects of the representation selected are entangled with the performance of the learning approach we apply to develop the structural alphabet. Overcomplicated representations can sometimes lead to overfitting. To avoid this problem, we currently focus on the dihedral angles. Other features can be easily included in the representation if proved necessary.

The second factor is the size of the alphabet. We took advantage of the SOM as a visualization tool that helps determine the alphabet size. By systematically varying the number of map units on the map, we visualized the clustering behavior of the SOM. Our experiments showed a distinct plateau corresponding to the convergent number of clusters,

compared with the increasing number of clusters in the results of clustering on the random negative control dataset. This suggests that the structural alphabet size we found is not arbitrary.

Various types of algorithms have been applied to clustering local protein 3D fragments into a limited set of fold patters, e.g. self-organizing maps (SOM), hidden Markov models (HMM), neural networks, hierarchical clustering, k-means clustering, etc. Each has its own learning bias and inherent limitations. For example, the topology (e.g. number of layers or map units) of neural networks, the SOM and the HMM strongly affect the performance. The value of k in kmeans algorithm determines the clusters. As a consequence, the third factor is the learning algorithm. In our study, we took a multi-strategy approach. We first used the SOM and the minimum-spanning tree algorithm to determine the alphabet size, and then applied the k-means algorithm to group fragments into meaningful clusters. The number of map units in the SOM and the value of k in k-means are not prespecified in advance, but instead determined systematically. To verify the correspondence of our structural alphabet letter to the fold patterns, we computed the average within-cluster distance for each alphabet cluster as well as the distance across clusters. The small average within-cluster distance and the relatively large between-cluster distance demonstrate the significance of the structural alphabet we found. Furthermore, the visualized superimposition of protein fragments in each cluster also justifies the structural cohesiveness.

The objective of the study is to propose a new approach to developing the structural alphabet. To verify its usefulness, we tested it on the all-α proteins in SCOP, and the experimental results show its promising applicability. After the success on the all-α proteins in SCOP, we plan to test our method on different data banks to further verify its feasibility and generality. Also as mentioned above, the representation is a crucial factor in the alphabet design. We will consider other structural features besides dihedral angles, add more useful features to enhance our structural alphabet, and test the new approach on other families in SCOP.

1.4.3. Bicluster Analysis of Genome-Wide Gene Expression

There are two objectives in our experiments. One is to demonstrate the superiority of PIFP to those standard clustering methods in terms of identifying more meaningful gene groups related to GO categories. The other is to show PIFP's competitive performance compared with other current biclustering algorithms.

Two expression datasets were used in our analysis and comparison with other standard

clustering algorithms. One dataset contains 6335 genes with 121 conditions which were obtained by combining expression profiles from several gene expression experiments [37, 53, 73-78]. This dataset was used in our pilot tests to select the appropriate normalization and discretization procedures for PIFP. We tried different normalization and discretization methods as mentioned earlier, and settled on the one with the best performance. The second dataset is the one used by Hughes et al. [79], which contains 6325 genes and 300 conditions. We tested PIFP on this dataset and compared with several representative conventional clustering algorithms. In order to keep the consistency, instead of reimplementing these algorithms or using any ad hoc versions, we adopted in our experiments Cluster3.0 [37], which provides hierarchical clustering, k-means, SOM and PCA, and also has been used in other published similar experiments [49]. In addition, we also included the fuzzy c-means method in our experiments [80].

The quality measure for a cluster is the *p*-value based on the widely used hypergeometric distribution [80, 81], defined as below,

$$p = \frac{\sum_{x=z}^{\min(N,K)} \binom{K}{x}\binom{M-K}{N-x}}{\binom{M}{N}}, \qquad \text{(Eq. 1.4.3.1)}$$

where M is the total number of genes, N is the size of the cluster, K is the total number of genes annotated in some GO [82] category, and z is the number of genes within this cluster in common with this GO category. This measure takes into account both the cluster size and the number of clusters found with respect to the correlation with GO categories. The smaller the *p*-value, the more consistent the cluster with the annotations. We took the negative logarithm of the *p*-value to increase the readability.

Following [37] to use the default parameter settings in experiments, with Cluster3.0, we tested hierarchical clustering, k-means, SOM and PCA on Hughes expression data. We plot the –log(*p*-value) vs. cluster count distribution for each of the above methods, as shown in Figure 1.4.3.1(a) to Figure 1.4.3.1(d) respectively, and the results of fuzzy c-means and PIFP are presented in Figure 1.4.3.1(e) and Figure 1.4.3.1(f). In each histogram, we show the cluster count distribution for –log(*p*-value), and each bar represents the number of clusters with the corresponding –log(*p*-value) ranging from 1 to 30.

We also summarize the total number of clusters found, the mean and standard deviation of cluster size and –log(*p*-value) as well as the proportion of clusters with –log(*p*-value)>5 in Table 1.4.3.1. It is interesting to see that the standard deviation of cluster size is quite large. This seems to agree with the real world that various gene groups of different size perform

very different biological functions. We further divide the values of –log(*p*-value) into three intervals, and show their proportion distributions for all algorithms in Figure 1.4.3.1. It indicates that the quality, measured by –log(*p*-value), of the clusters found by the standard clustering algorithms mostly falls within the first interval (i.e. 1≤–log(*p*-value)≤5), but on the contrary, most of the biclusters identified by PIFP cover the other two. All the evidence provided by Figure 1.4.3.1, Figure 1.4.3.2 and Table 1.4.3.1 clearly shows that PIFP outperforms all the standard clustering algorithms in our experiments.

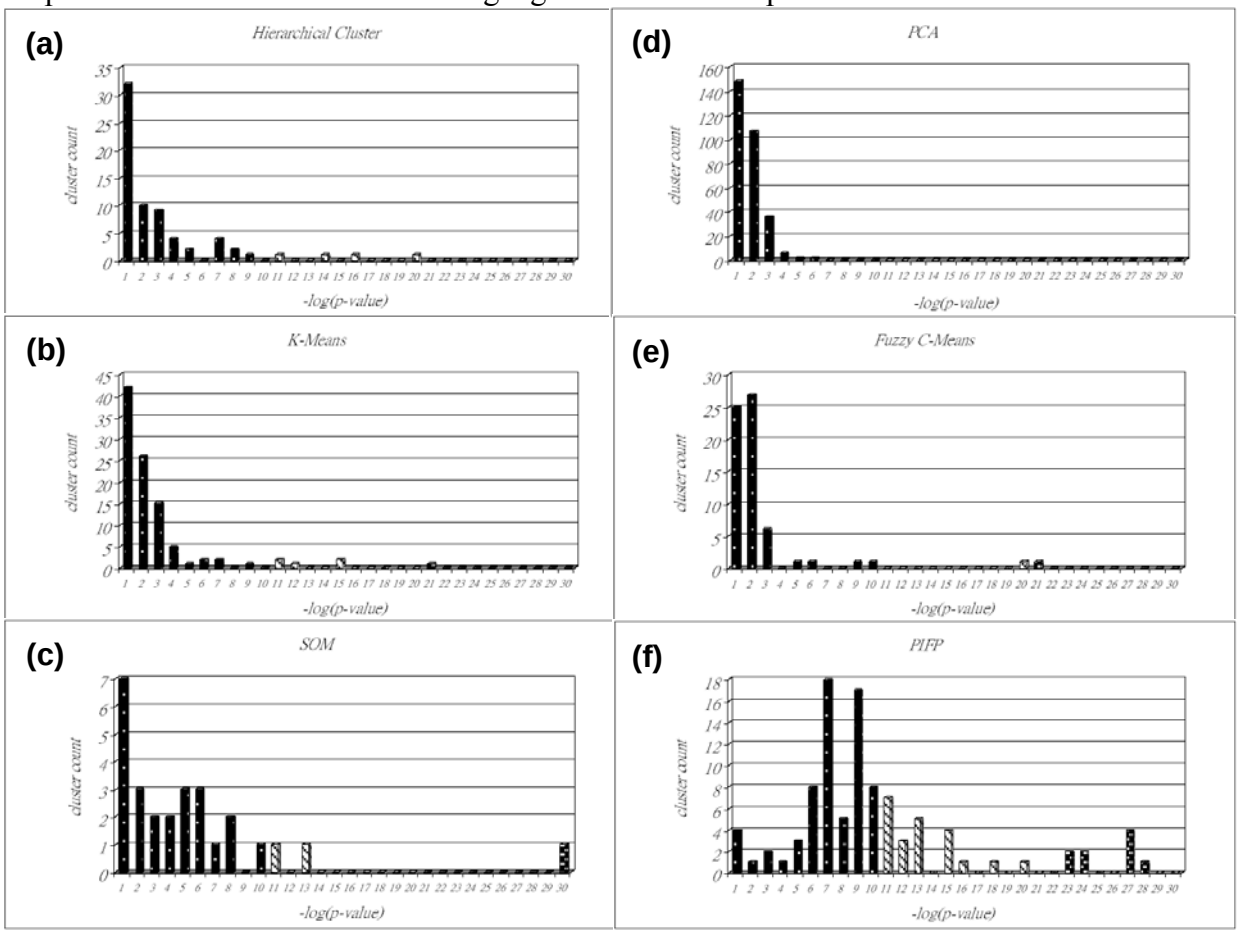

Figure 1.4.3.1. The cluster count distribution of –log(*p*-value) for each algorithm: (a) hierarchical clustering, (b) k-means, (c) SOM, (d) PCA, (e) fuzzy c-means and (f) PIFP. We partition the value of –log(*p*-value) into three intervals and present the distribution with bars of different styles to increase readability. Also note that the clusters with –log(*p*-value)>30 are included as 30 in order to save figure space.

Table 1.4.3.1. Summary of comparisons between standard clustering algorithms and PIFP

Algorithm	Total Cluster	mean±s.d. (cluster size)	mean±s.d. (–log(*p*))	-log(*p*)> 5
Hierarchical	68	89.37±61.66	3.22±3.72	16.18%
k-means	100	63.25±82.27	2.84±3.33	11.00%
SOM	27	233.41±91.90	5.44±5.83	37.04%
PCA	300	63.25±82.27	1.70±0.85	0.33%
Fuzzy c-means	64	95.83±343.29	2.63±3.60	7.81%
PIFP(s=17,i=21)	98	255.72±301.21	10.20±5.97	88.78%

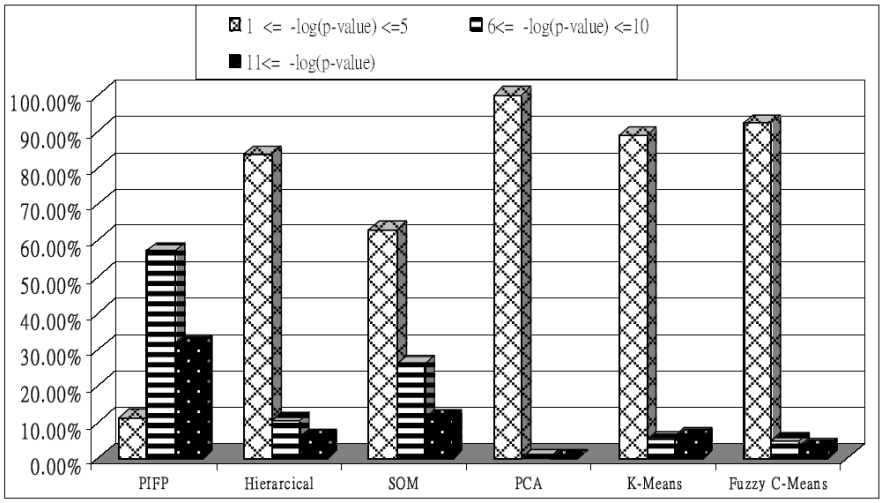

Figure 1.4.3.2. The proportion of –log(*p*-value) in different intervals. We divided the value of –log(*p*-value) into three intervals, 1≤−log(p-value)≤5, 6≤−log(pvalue)≤ 10 and 11≤−log(p-value). We present the percentage of clusters (or biclusters) with −log(p-value) falling within each interval in bars of different styles. It clearly shows that the clusters produced by the standard clustering algorithms mostly fall in the first interval. On the other hand, most of the PIFP's biclusters cover the other two intervals.

Since PIFP is controlled by two parameters, to verify the stability of PIFP in terms of its parameter settings, we varied the parameter values, 15<*s*<80 and 30<*i*<70, to generate over 2000 different parameter settings. We tested PIFP with these different parameter values on the same dataset. The average of –log(*p*-value) for all parameter settings is above 20, which is still better than the other conventional clustering algorithms.

Recently, Prelic *et al.* proposed a framework for systematic comparison and evaluation of biclustering algorithms, and developed a biclustering analysis toolbox [47]. Besides the representative clustering algorithms, we compared PIFP with those biclustering systems provided in this toolbox, including OPSM [39], ISA [49], CC [40], xMotifs [50] and BiMax

[47]. In addition, another biclustering system, Samba [44], was also included for comparison.

Table 1.4.3.2. Summary of parameter settings and total number of biclusters

Algorithm	Default Parameter Settings	Values Used	Total Biclusters
Samba	D=40,N1=4,N2=6,k=20,L=30	default	100
ISA	t_g =1.8~4.0(step 0.1), t_c =2.0, nr. seeds=20000	t_g =2.0, seeds=500	66
CC	α=1.2,δ=lower end of the expression value range	δ≤0.5	100
OPSM	*l*=100	default	12
xMotifs	n_s =10,nd=1000, s_d =7~10, α not given, *p*-value=10^{-10}, max_length not given	s_d =7,α=0.1, max_length=0.7m	306
BiMax	min no. of genes not given, min no. of chips not given	min_genes=12, min_chips=11	100
PIFP	*s*=10~20,*i*=10~25	*s*=11,*i*=12	100

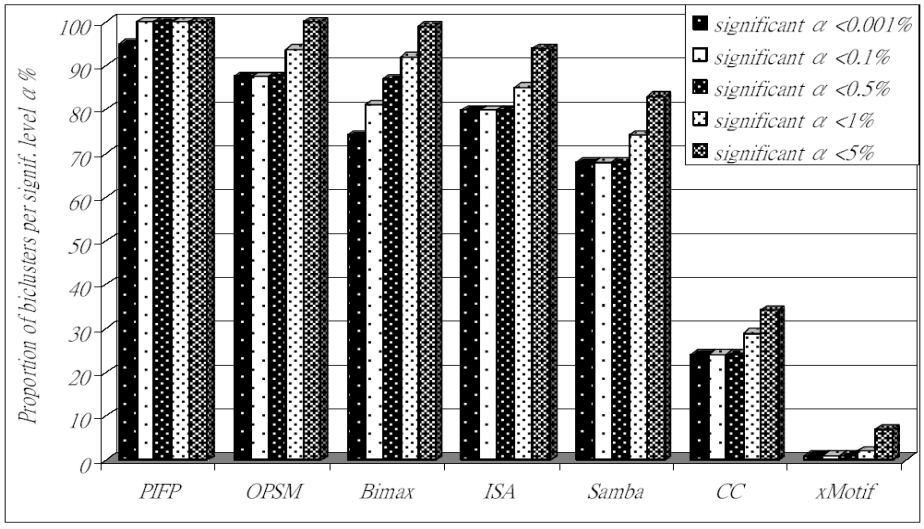

Figure 1.4.3.3. The proportion of biclusters significantly enriched by GO annotation categories for each biclustering algorithm. Different bars in a group represent the results obtained for five different significance levels (i.e. α).

Table 1.4.3.3. Partial results of Biclusters found by PIFP

Bicluster	*Size*	*p-value*	*Annotation*
Bicluster1	228	1.80E-119	cytosolic ribosome (sensu Eukaryota)/80S ribosome
Bicluster85	166	3.00E-39	ribonucleoprotein complex/RNP
Bicluster58	58	5.30E-32	Ribosome
Bicluster41	101	3.90E-26	ribosome biogenesis
Bicluster7	135	5.80E-24	protein complex
Bicluster11	120	2.60E-19	RNA metabolism
Bicluster69	64	5.40E-19	non-membrane-bound organelle
Bicluster13	128	4.00E-18	physiological process
Bicluster14	108	2.60E-16	cellular process
Bicluster76	70	8.00E-14	cytoplasm organization and biogenesis
Bicluster24	76	5.80E-12	cellular physiological process/cell growth and/or maintenance/cell physiology
Bicluster64	70	4.40E-11	translation factor activity, nucleic acid binding
Bicluster77	62	2.40E-10	mitochondrial ribosome
Bicluster83	33	7.10E-09	generation of precursor metabolites and energy/energy pathways
Bicluster67	42	1.20E-06	translation initiation factor activity
Bicluster82	19	7.70E-06	peroxisomal matrix
Bicluster35	14	9.60E-06	cellular biosynthesis
Bicluster34	23	3.60E-05	fatty acid elongase activity
Bicluster95	23	3.80E-05	molecular function unknown
Bicluster94	32	9.70E-05	binding/ligand

We used the same yeast dataset and the same parameter settings as in [47] in our experiments to keep the consistency. This dataset contains 2993 genes with 173 different stress conditions. The parameter settings and the number of biclusters identified are listed in Table 1.4.3.2. Details can be found in [39, 40, 44, 47, 49, 50]. Like Prelic *et al*., we evaluated biclusters by calculating the hypergeometric functional enrichment score with FuncAssociate [83] as the quality measure, and the results are also summarized in a histogram as presented in Figure 1.4.3.3. The histogram shows at different significance levels for each algorithm the fraction of all biclusters found with which one or more GO annotation categories are highly correlated. The result demonstrated that PIFP outperforms all the other biclustering algorithms. We present partial results of the biclusters found by PIFP in Table 1.4.3.3.

Table 1.4.3.4. Results of ablation study for PIFP

Dataset	Dataset in BicAT[16]		Hughes et al. Dataset[32]	
Algorithm	**PIFP w/o enhancement**	**PIFP with enhancement**	**PIFP w/o enhancement**	**PIFP with enhancement**
No. of Biclusters	36	100	20	98
GO annotation	9	20	3	9
Running time (Second)	48	70	443	665

We also did an ablation study for PIFP to demonstrate the effectiveness of the filtering and masking procedures. Using the same parameter settings, we tested PIFP with and without the filtering and masking procedures on the same dataset to compare their performance. The datasets used in the above experiments were again used in the ablation study. Our study shows that with the filtering and masking procedures, PIFP does not only produce more biclusters than without, but also covers all the biclusters found by the ablated version. The comparison results are presented in Table 1.4.3.4, including the number of biclusters found, the number of relevant GO annotation categories and the running time. It demonstrates that PIFP with the enhancement procedures can produce about three times the number of biclusters found by the ablated version with less than twice the computational time.

Most of the available expression analysis tools are based on clustering that try to establish either groups of genes which are co-regulated under all the measured conditions, or groups of conditions which are conserved across all the measured genes. Although analysis methods of this kind have proved useful in several applications, yet their biological validity of the global assumptions may be questioned especially when the analysis goal is to identify molecular networks [54]. Numerous biclustering algorithms have consequently been proposed to mitigate the problem. However, some are limited to finding simple biclusters, e.g. constant biclusters, and some, on the other hand, though capable of identifying more complicated bicluster structures, are hampered by high computational complexity. In this study, we presented a new method for the analysis of gene expression data with the aim to seek the balance between the expressiveness of biclusters and the complexity of search strategies.

Based on the framework of market basket analysis, we transform the biclustering problem into a frequent itemset finding problem. By extending the FP-growth method to develop PIFP, we are able to efficiently and effectively identify interesting biclusters described as frequent itemsets whose biological significance has been verified by the domain knowledge (i.e. Gene Ontology). To demonstrate PIFP's performance, we tested it on the same datasets that have been widely used by other researchers, and compared the results with those of other current biclustering methods. Our experiments showed PIFP outperformed the

others.

PIFP can be further extended in several directions. First, the current definition of our frequent itemsets can only represent biclusters with coherent values. With a more flexible definition, we will be able to identify biclusters with coherent trends as well. Second, along the same line of the above issue, it is possible to incorporate domain knowledge or user-defined constrains into the finding of frequent itemsets. This will not only accelerate the search process, but also enable the search strategy to focus on the desired items if necessary even when their support is below the pre-specified threshold. Third, from the frequent itemsets found, we can derive the association rules which may reflect the relationships among different members within a bicluster. Such relationships can be later generalized into transcription modules or even transcriptional regulatory networks if the knowledge of transcription factors is available.

Third Year:

1.4.4. A Two-stage Approach to Finding Common Structure Elements in Unaligned RNA Sequences

Several recent tools were selected for comparison, including MARNA [15], CMfinder [14], and RNAshapes [84]. As these algorithms were derived from different design philosophies, we followed Yao *et al.* [14] to test each algorithm on the same input data using default parameter settings to conduct a reasonably fair and consistent comparative study.

We picked 7 families of different sizes from the Rfam database as the test data. The seed alignment for each family is considered the consensus motif, whose number of hairpins varies from one to three among different families. Unlike Yao *et al.* [14], who included a fixed number of genomic sequence bases (e.g. 200 bases), we instead included genomic sequence flanking the motif such that the ratio of the motif length to the sequence total length is set between 0.1 and 0.6 at random, to reflect the reality that motif positions are usually unknown. The smaller the ratio, the larger the length difference between motifs and sequences. The average flanking genomic sequence length can then vary from 50 to more than 250 bases for different families. As the length of genomic flanking sequences has a larger deviation by our setting, the test data are more challenging than Yao *et al.'s*, and these test datasets are summarized in Table 1.4.4.1.

Table 1.4.4.1. Summary of Rfam families for testing

	Max/Min/Avg Seq Length	Avg Motif Length	No. of Hairpins in Motif	Total Sequences
ctRNA_pGA1	303/299/300	62	2	17
Entero_CRE	312/212/231	39	1	56
HepC_CRE	202/152/170	48	2	47
IRE	181/81/140	28	1	34
Lin-4	322/320/321	68	1	9
Purine	201/99/190	72	3	35
s2m	164/160/163	41	1	38

The performance was measured at the base pair level relative to the Rfam annotation. We compared the predicted motif against the annotated seed alignment provided in Rfam. Let P_t (true positive) denote the number of base pairs that exist in annotated seed alignments and are correctly predicted, P_f (false positives) denote the number of base pairs that do not exist in annotated alignments but are predicted, and N_f (false negatives) denote the number of base pairs that exist in seed alignments but are not predicted. The overall accuracy of a prediction is computed as the MCC (Matthews Correlation Coefficient) approximated by the geometric mean of sensitivity and positive predictive value [69].

$$MCC \approx \sqrt{\frac{P_t}{P_t + N_f} \cdot \frac{P_t}{P_t + P_f}}, \qquad \text{(Eq. 1.4.3.1)}$$

As MARNA has a lower limit on input size, for those families larger than 20 RNAs, we randomly picked 20 sequences for testing. The MCC for each method is presented in Table 1.4.4.2. For a complete comparison, we tested all the methods, except MARNA, on the full set of seed sequences in each family, and summarized the results in Table 1.4.4.3. According to Table 1.4.4.2 and Table 1.4.4.3, our approach outperformed RNAshapes and MARNA in most of the tests, and was comparable to CMfinder.

Table 1.4.4.2. Summary of prediction accuracies (MCC) for partial Rfam families

	MARNA	RNAshapes	CMfinder	Ours
ctRNA_pGA1	0.890	0.873	0.950	0.959
Entero_CRE	0.765	0.844	0.954	0.936
HepC_CRE	0.659	0.911	0.998	0.987
IRE	0.499	0.569	0.899	0.847
Lin-4	0.793	0.797	0.795	0.711
Purine	0.749	0.558	0.900	0.864
s2m	0.282	0.241	0.855	0.899

Table 1.4.4.3. Summary of prediction accuracies (MCC) for complete Rfam seed sets

	RNAshapes	CMfinder	Ours
ctRNA_pGA1	0.790	0.950	0.959
Entero_CRE	0.816	0.913	0.934
HepC_CRE	0.805	0.999	0.976
IRE	0.502	0.970	0.902
Lin-4	0.796	0.795	0.711
Purine	0.749	0.923	0.903
s2m	0.160	0.897	0.923

Table 1.4.4.4. Robustness comparison

	CMfinder	Ours
ctRNA_pGA1	0.950	0.959
Entero_CRE	0.913	0.934
HepC_CRE	0.999	0.976
IRE	0.862	0.871
Lin-4	0.478	0.660
Purine	0.923	0.903
s2m	0.897	0.923

To further compare our system with CMfinder in robustness, we added noise to the datasets by putting in 15 random non-family RNA sequences. We present the results in Table 1.4.4.4, and it shows no significant difference in all test datasets except two families, lin-4 and IRE. Note that the family size of lin-4 is much smaller than that of the others. It contains only nine seed sequences. After we added 15 noise sequences, the low signal/noise ratio affected CMfinder more significantly than our approach. On the other hand, though compared

with the others IRE is not a small family (34 RNAs), yet the IRE motif is relatively small. Its size is only 28 nts, which could be easily clouded by noise. Table 1.4.4.4 indicates that our system was more robust than CMfinder in these tests.

Given a set of unaligned RNA sequences, the goal is to find the consensus structure motifs in these RNAs. In this paper, we proposed a two-stage approach by separating motif finding from sequence folding. Within this framework, not only can new folding tools be easily added to increase reliability, but other optimization techniques than Gibbs can also be applied to improve accuracy. The competitive performance of the new approach was demonstrated by testing it on various Rfam families.

In the future work we plan to extend the approach in two directions. First, we will increase its applicability to finding characteristic structure motifs in mixed unaligned RNAs from multiple families. Second, we will develop an adaptive mechanism for parameter tuning of *RLD* and *sim* thresholds so the system can adjust the threshold automatically.

Chapter 2:
Protein-DNA interaction

2.1 Introduction

The double-strand DNA within cells is the most important element of living organism. The blueprint of cell processes like growth, cell division, and apoptosis are coded in the DNA. DNA-binding proteins play a key role in living organisms of many genetic activities such as transcription, recombination, rearrangement, DNA replication and repair. Some kind of DNA-binding proteins which are also called transcription binding factors (TFs) mediate the regulation of various genes. Such regulations play a key role in biological pathway and reconstructing the network of pathways is the primary goal of post-genomic era. One or more domains of these proteins interact with DNA, and they offer the specificity for direct and indirect readout of DNA [85]. To identify the DNA-binding domains is very important for understanding the regulation mechanisms. Most of the structural DNA-binding domains can be categorized into several classes according to their structures or binding type [86-88]. However, some DNA-binding domains can not be well categorized, and for some DNA binding domains structural information is unavailable [86, 89].

Recently, the rapidly increasing crystal data on the protein-DNA complex provide a rich source of information about the interactions between amino acids and DNA base pairs [90, 91]. Furthermore, the growing bioinformatics also help researchers to handle the vast amount of data of proteins generated by various approaches. Many easy-to-use databases which record important interaction information of protein and DNA are available on the internet. There are also many computational-based tools that can help us to predict novel DNA-binding proteins, the target sites of DNA-binding proteins, and possible interactions between proteins and DNAs. These resources offer a good basis for researchers to study this topic and to develop more efficient and accuracy methods for protein-DNA interactions.

The goal of this section is to understand relationship of protein/DNA structure and function by our new scoring method to provides the binding model and interacting amino acids and DNA bases of predicted partners. In the past three years, we had published one paper as fellow:

1. Y. L. Chang, H. K. Tsai, C.Y. Kao, Y. C. Chen, **Y. J. Hu,** and J. M. Yang, "Evolutionary conservation of DNA-contact residues in DNA-binding domains," *BMC Bioinformatics*, vol. 9 Suppl 6, pp. S3, 2008.

2.2 Motivation

As previous studies, we know DNA-binding domain is the key part for protein-DNA bindings. Experimental approaches for finding such pairs usually expensive and time-consuming. Rapidly increasing amount of protein-DNA complexes from X-ray crystallography and nuclear magnetic resonance (NMR) have enabled the use of structural-based approaches for identifying DNA-binding proteins. We propose computational approach called “3D-regulogs” to large scale infer protein-DNA binding partners by using the concept of regulog and the crystal structures of protein-DNA complex as templates. Such method also provides the binding model and interacting amino acids and DNA bases of predicted partners.

Our project is proceeded with two parts in three years. The special goals of every year will be descried detail as follows.

2.2.1 Evolutionary conservation of DNA- contact residues in DNA-binding domains

We propose this structure-based threading method by considering evolutionary conservation of DNA-contact residues in DNA-binding domains to identify DNA-binding domains. We use BLOSUM62 [92], an evolutionary-based scoring matrix for amino acid substitutions, to measure the degree of conservation of binding residues. Our method can achieve high precision and recall for 66 families of DNA-binding domains, with a false positive rate less than 5% for 250 non-DNA-binding proteins.

2.2.2 Evolutionary conservation and Interacting preference for identifying protein-DNA interactions

We considered both the evolutionary pressure and the protein-DNA interacting preferences of contact residues by modifying and enhancing our previous study [93], which identified DNA-binding domains using a consensus scoring function. Although the consensus scoring function has a good measurement on the evolution pressure of DNA-contact residues for identify DNA-binding domains, it can not reflect the binding affinities between proteins and DNAs. Here, we introduced a new scoring function combining evolutionary pressure and protein-DNA interacting preferences. The combination of these two scores is useful for

identifying DNA-binding domains and modeling protein-DNA interactions.

2.3 Background

Several studies used various computational approaches to predict potential DNA-binding proteins by using protein-DNA complexes structure features, such as the overall charges, electric moments, and shape of binding sites [94-100]. Since the charge and conformational complementarities of binding sites are essential for protein- DNA binding, these features provide a reasonable basis to identify DNA-binding proteins. Another trend is to consider the degree of conservation of residues [101-103]. Luscombe and Thornton [104] have studied 21 families of DNA-binding proteins and showed that those amino acids interacting with the DNA are better conserved than those not interacting with DNA. Stawiski et al. [105] found that electrostatic patches of DNA-binding proteins have a higher percentage of aromatic and positive residues. According to the general properties of 20 amino acids, they also showed that residues of the patch are conserved at property levels.

Some experimental technologies have been proposed to generate numerous binding data for studying the interactions of proteins and DNAs. One approach is the SELEX [106] which uses a particular protein to select DNA targets from a randomize oligo-nucleotide pool. The phage display [107, 108] is another experimental method which fixes the DNA target and randomized the specific position of the protein. One of the most successful cases of these two approaches is applied to zinc finger proteins which utilize three specific amino acids to recognize three consecutive DNA bases [109, 110]. Recently, the ChIP-on-chip is a large-scare technology which was firstly applied to identify binding sites of transcription factors in yeast [111-113]. It can large scale identify protein-DNA binding partners in a very efficient way and extend to whole genome analysis. In 1976, Seeman et al. proposed some recognition interacting patterns (i.e. formation of hydrogen bonds) between amino acid and DNA bases [114]. However, the increasing evidences showed that there is no simple code or general principle for protein-DNA recognition [115]. Based on the crystal structures of protein-DNA complexes, Margalit and co-worker proposed matrix-like parameters for quantitatively measuring the contact preferences of amino acids and DNA bases [116]. On the other hand, with the growing X-ray crystal structures, more and more computational approaches have been proposed for studying protein-DNA interactions. These scoring approaches can be roughly classified into two categories. One is to develop statistic-based or knowledge-based methods to predict the binding affinities of proteins-DNA targets. For example, Kono and Sarai [117] threaded DNA sequences into a protein-DNA complex structure and then a knowledge-based function was used to evaluate the affinity of the

threading sequences. They successfully predicted the DNA binding sites of a regulatory protein using this approach. Further, Liu *et al*. proposed another knowledge-based function which considered the distance between protein residues and DNA triplets to evaluate the protein-DNA interactions [118]. They achieved high accuracy in predicting binding free energies of zinc finger proteins and 48 public protein-DNA complexes. The second category utilizes energy-based functions to model the binding affinities between proteins and DNA. Baker and co-workers developed and parameterized a physical model for predicting protein-DNA interfaces and redesign and binding sites prediction [119]. They also predicted the position weight matrix (PWM) of several transcription factors on DNAs [120]. However, such atom-based approach also has challenged for homology modeling [120].

2.4 Methods

According to our different aims, our experiments are also carried on two steps. The detail procedures are described in proper order as bellow.

2.4.1 Evolutionary conservation of DNA-contact residues in DNA-binding domains

Figure 2.4.1.1 shows the flowchart of our proposed method. We quantitatively evaluated whether a given protein domain *M* has a similar DNA-binding function to a known crystal protein-DNA structure. For each crystal structure of protein- DNA complex in Protein Data Bank (PDB), we first identified the DNA-contact domain (*D*) using geometry information and domain definitions from Structure Classification of Proteins (SCOP, version 1.71) [121]. The structures and sequences of both protein-DNA complexes and their DNA-contact domains were collected in the template library. For a given protein sequence/structure *M*, we used sequence/structural alignment tools to find the homologous DNA-contact domain *D* from the template library. Finally, we proposed a score method to evaluate the similarity between *M* and *D* based on the BLOSUM matrix. Detailed descriptions are as follows.

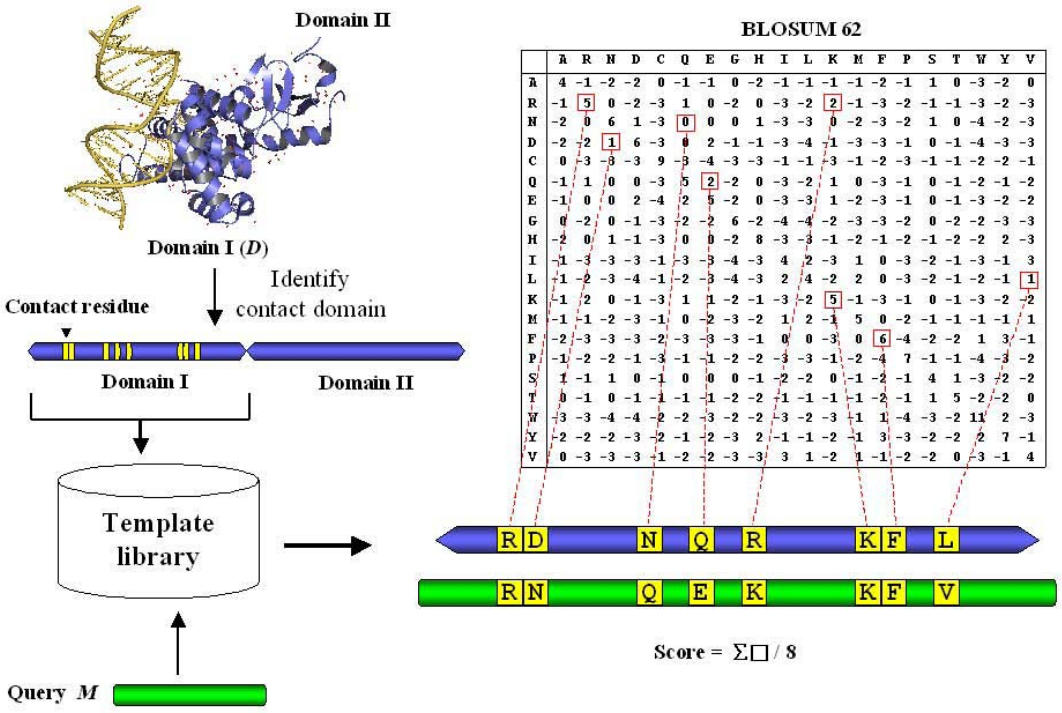

Figure 2.4.4.1. Flowchart of proposed method.

Template library

We first collected protein-DNA complexes from PDB and each complex should contain at least one protein chain and a double-strand DNA. As in Luscombe et al. [122], a complex was excluded if its DNA is single-stranded or the length of the DNA is less than 4 bases. For each protein-DNA complex, we then identify contact residues and contact domains of this protein. Contact residues, whose heavy atoms are within a distance (distance ≤ 4.5 Å) of any heavy atoms of the bound DNA, are considered as the core parts of the contact domain in a complex [120]. For each protein-DNA complex, we identified its DNA-contact domains according to contact residues and the definition of the SCOP database. Each domain must have more than 5 contact residues and the number of residues of this protein is more than 50 to make sure that the contact between the protein and DNA was reasonably extensive. Finally, 230 contact DNA-binding domains were identified and collected in the template library.

Homologous proteins searching

For a given protein sequence/structure *M*, we found a homologous DNA-binding protein from the template library using alignment tools. If *M* is a 3D-structure, we used a structure

alignment (i.e. CE [123]) to align *M* to all contact domains. The CE will return a *Z* score for each alignment representing the structure similarity of the two aligned structures. DNA-binding proteins are considered as homologous proteins of query *M* if CE *Z* scores of exceed 3.7 based on CE's statistical model. On the other hand, if *M* is a protein sequence, we used sequence alignment (i.e. FASTA [124-126]) to search the template library. Here, a DNA-binding protein is considered a homologous protein of *M* if the sequence identity exceeds 25% according to observations of previous studies [66, 127-131].

Scoring method

For an alignment of the query domain (*M*) and a contact domain (*D*) that satisfies the above criterion, we calculate the alignment score for the aligned contact residues by using the BLOSUM62 matrix. The scoring method is defined as:

$$S_M = \frac{\sum_{i \in CR} \text{BLOSUM62}(d_i, m_i)}{\#\text{contact residues}}, \qquad \text{(Eq. 2.4.1.1)}$$

where *CR* is the set of the contact residues between *D* and *M*; *di* and *mi* denote the corresponding *ith* contact residue of *D* and *M*, respectively. Here, the score of a misaligned residue is -4 which is the smallest in the BLOSUM62 matrix.

2.4.2 Evolutionary conservation and interacting preference for identifying Protein-DNA interactions

Figure 2.4.2.1 shows the scheme of our proposed scoring function for identifying DNA-binding domains and predicting protein-DNA interactions. We first compiled 1204 protein-DNA complex structures from the Protein Data Bank (PDB) [91]. Protein-DNA complex structures were then used as templates to identify potential DNA-binding proteins /domains. Third, the DNA-contact residues of these complexes are identified by using geometry information of the structures. For a given template structure *T* and a protein sequence/structure *P*, we obtained the alignment of *T* and *P* by using sequence/structure alignment tools. We then proposed a scoring function to quantitatively evaluate the function similarity between *T* and *P* based on conservation score of the DNA-contact residues and the interacting scores between contacted residues (protein side) and bases (DNA side) of the template *T*. Detailed descriptions are described as the following subsections.

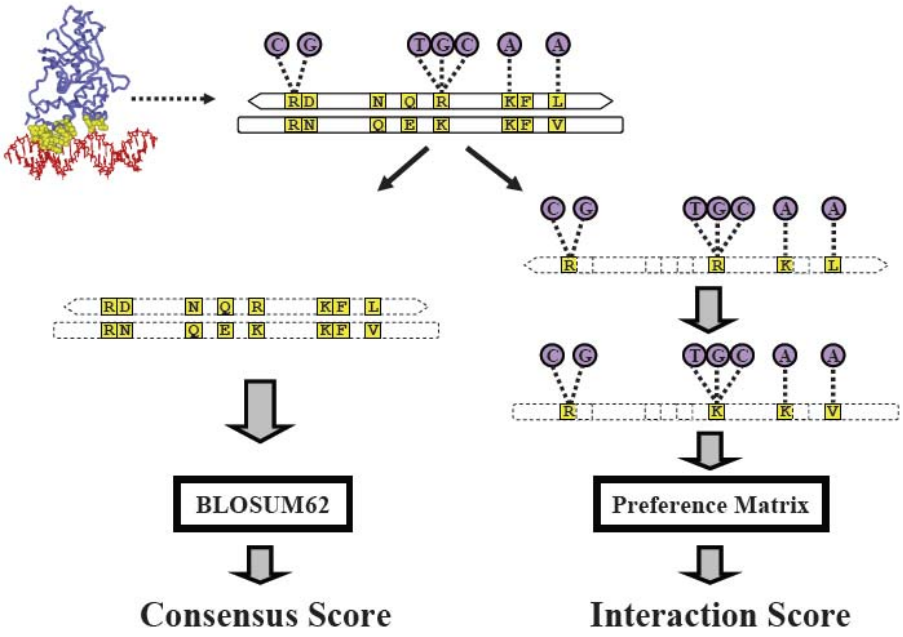

Figure 2.4.2.1. The scoring schema using the evolutionary conservation and interacting preference.

Template preparation

Protein-DNA complex structures solved by X-ray crystallography (resolution > 3.0 Å) and NMR were obtained from the December 2007 release of the PDB. According to the work proposed by Luscombe *et al.* [122], we selected 1043 complexes by excluding complexes which are single-strand binding complexes or the numbers of DNA bases are less than 4. For each protein-DNA complex in this selected set, we identified the contact residues, whose heavy atoms are within a distance (distance ≤ 4.5 Å) of any heavy atoms of the bounded DNA, of the DNA-binding protein. The DNA-contacting residues are considered as the core part of a DNA-binding protein. To obtain reasonably extensive interface of a protein-DNA complex, a DNA-binding protein is required to have more than five contact residues and the number of residues of this protein is more than 50. The residue-DNA bases interacting pairs were also obtained from the protein-DNA complex. A residue *R* and a DNA base *B* are defined as an interacting pair if any heavy atoms of *R* and any heavy atoms of *B* are within a distance 4.5 Å.

Alignment Tools

For a given protein template *T* and a query protein sequence/structure *P*, we obtained the alignment by the following steps: If *P* is a protein structure, we used a structure alignment tool CE [132] to align *T* and *P*. The CE will return a *Z* score for the alignment representing the structure similarity of these two structures. The *P* is considered as a homologous protein of *T* if the *Z* score exceeds 3.7 based on CE's statistical model. On the other hand, if *P* is a protein sequence, we applied the sequence alignment tool FASTA [126] to align the two

proteins (i.e. *T* and *P*). The *P* is considered as a homologous protein of *T* if the sequence identity exceeds 25% according to the observations of previous studies [66, 128, 130].

Scoring function

For a homologous protein *P* of a template *T* (i.e. the alignment of *P* and *T* satisfies above two criteria), we used three scoring methods to calculate the score of *P* based on aligned contact residues of *T*. These methods, including consensus score, interaction score, and combination score, are described in the following subsections.

Consensus Score

We calculate the consensus scores of *P* based on aligned contact residues of *T*. The BLOSUM62 matrix [92] is applied here to evaluate the change of contact residues. The consensus scoring function is defined as

$$S_{cons} = \frac{\sum_{i \in CR} BLOSUM62(t_i, p_i)}{\#\text{of contact residues}}, \qquad \text{(Eq. 2.4.2.1)}$$

where *CR* is the set of the contact residues between *T* and *P*; *ti* and *pi* denote the *ith* contact residue of *T* and its corresponding aligned residue of *P*, respectively. Here, the score of a misaligned residue is -4, which is the smallest value in the BLOSUM62 matrix.

Interaction Score

The interaction score is obtained by the following steps. For all contact residue-base pairs between protein and DNA, respectively, in template *T*, we first replace the residues of those pairs with aligned residues in *P*. We used the knowledge-based scoring matrix *M*, which was proposed by Margalit and co-worker [116] to measure the preference of residues and DNA bases, to score the binding affinity between the target protein *P* and DNA based on template *T* (Figure 2.4.2.1). Finally the interaction score is given as

$$S_{int} = \frac{\sum_{i} M(R_i)}{\#\text{contact pairs}}, \qquad \text{(Eq. 2.4.2.2)}$$

where *M*(*Ri*) is preference in matrix *M* and *Ri* is the *ith* contact pair in *P*. When a contact residue is aligned to gap, we used the smallest score (-3.93) in *M* to be the score.

Combination Score

The combination score, which is the linear combination of the consensus score and the interaction score, is given as

$$S_{combination} = \omega_1 \cdot S_{cons} + \omega_2 \cdot S_{int} \qquad \text{(Eq. 2.4.2.3)}$$

where *w1* and *w2* is the weight of the consensus and interaction scores, respectively. Here, we set both *w*1 and *w*2 to 1.

2.5 Result

2.5.1 Evolutionary conservation of DNA-contact residues in DNA-binding domains

Given a query domain, our method identified similar DNA-binding structures or homologous protein sequences from the template library. To evaluate the performance of our method, for each DNA-contact domain (*D*) in the template library we generated its corresponding positive and negative sets. The members in the positive set contain the domains similar to domain *D* based on SCOP, while domains in the negative set do not. By applying our method on these two sets, we found that the scores of the domains in the positive set are significantly higher than those of domains in the negative set. We further determined a threshold to achieve high precision and recall. Combining with the threshold, we applied our method on 66 known SCOP families of DNA-binding domains and 250 non-DNA-binding proteins to examine the performance.

Positive and negative set for each contact domain

We collected DNA-binding contact domains from SCOP database, the detail is described in Method. To remove redundant contact domains, domains with highly similar sequences (identity > 90%) are grouped using the NCBI software BLASTCLUST. In each group, the one with the maximal number of contact residues is chosen as the representative domain of a group. For a representative domain *R*, these protein domains in the same SCOP family are considered as the member of *R* according to SCOP95 (members whose similarity greater than 95% are excluded). Each member of *R* was aligned to *R* using the CE. We define a residue of *R* as misaligned if it is aligned to a gap. A family member is discarded if more than 20% contact residues of *R* are misaligned between *R* and this member. Family members that satisfy

the above criteria are considered to be in the positive set. If there are less than five members in the positive set of *R*, the entire family of *R* is discarded. We finally yielded 66 representative domains with corresponding positive sets. For each *R*, we artificially generated 1000 domains to be the negative set. To do this, for each artificial domain, we replicate its residues from *R*. Then we randomly mutated the residue type of each contact residue of *R*.

Determining the threshold of similar DNA-binding function of a contact domain

For each representative domain *R*, each member in the positive and negative sets was scored by the method we developed. Ideally, the scores of domains in the positive set should be on average significantly higher than those of the negative set. We used the Kolmogorov-Smirnov (KS) test to examine the above criterion. The KS test is a nonparametric test to determine if two distributions differ significantly. According to our results, the scores are significantly different for the positive set and the negative set in most domains (97% of 66 sets have a *p* value less than 0.05).

Further, given a contact domain, we would like to determine a threshold for determining which domains have a similar DNA-binding function. For the two sets (positive and negative) of a representative domain, we separately transform all members' scores to z-scores by

$$Z = \frac{s - \mu}{\delta} \quad \text{(Eq. 2.5.1.1)}$$

where s is the score of a member, μ is the mean score of the these two sets, and δ is the standard deviation. Figures 2.5.1.1(A) and (B) show the precision (ratio of the number of retrieved true positive data to all retrieved data) and the recall (ratio of the number of retrieved true positive data to all true positive) with various z-score thresholds, respectively. As shown in Figure 2.5.1.1(A), when we set the threshold greater than two, the precisions of using different thresholds are very similar (>90%).

If we set the z-score threshold to one, only 60% of families are with high precision. The results imply that larger thresholds will yield higher precisions, but the benefit is limited when the threshold is larger than two. Oppositely, as shown in Figure 2.5.1.1(B), larger thresholds will reduce the recall. According to these results, we take the z-score threshold as 2.0 and the domains with a z-score higher than the threshold will be considered as putative DNA-binding domains.

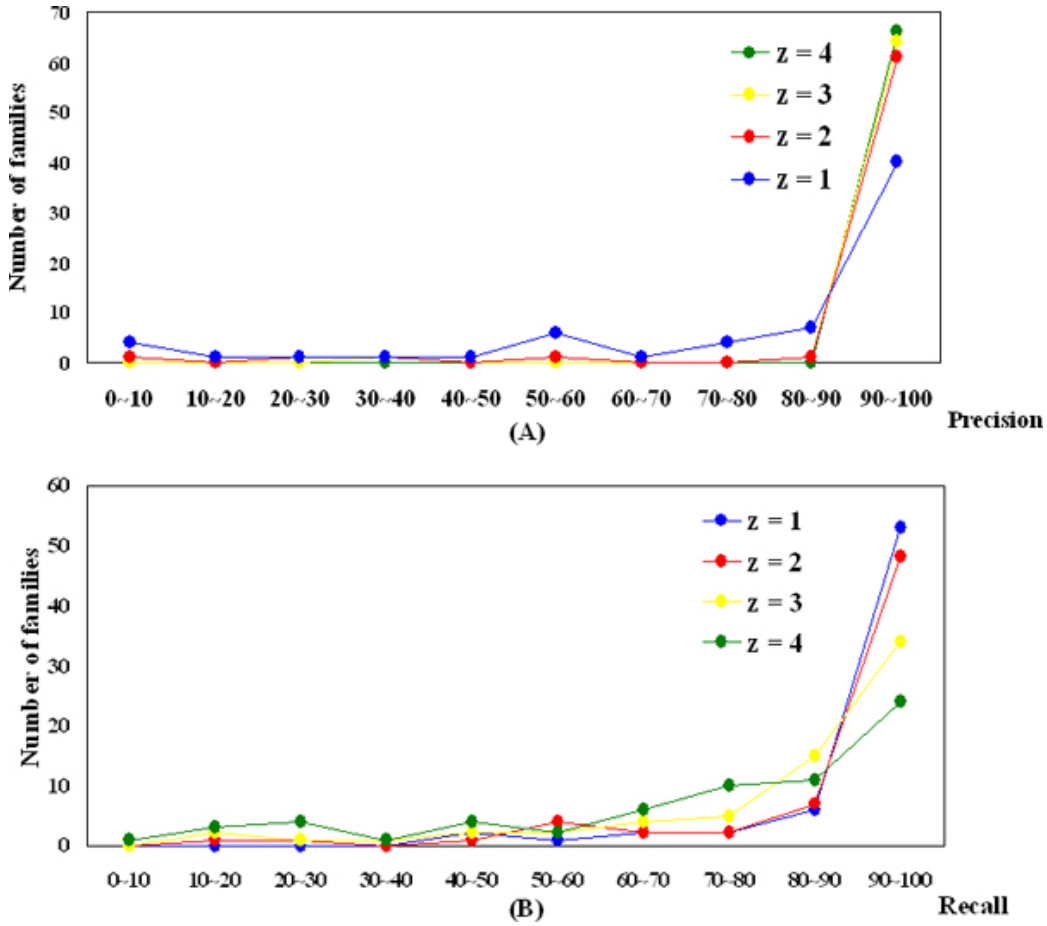

Figure 2.5.1.1. Precision and recall on different z-score thresholds. Our method results on different z-score thresholds for 66 representative domains. The distributions of the numbers of the families for **(A)** precisions and **(B)** recalls.

Non-DNA-binding proteins

We further apply our method to 250 non-nucleic-acid binding (non-DNA-binding) proteins, which were initially studied by Hobohm and Sander [133] and further specified by Stawiski *et al.* [105]. We align all non-redundant contact domains to those non-DNA-binding proteins using CE. Alignments whose z-scores (defined by CE) are greater than 3.7 with the misalign rate of contact residues less than 20% are chosen as non-DNA-binding domains. 177 non-DNA-binding domains pass the constraints among 250 proteins. We applied our method on these non-DNA-binding domains and transformed their scores to *z*-scores. Figure 2.5.1.2 shows the distribution of z-scores of non-DNA-binding domains. The scores approximately follow a normal distribution and the peak of the density occurred at Z = -1~0. Given a z-score threshold, the false positive rate is the ratio of number of domains whose z-score are beyond the threshold to the total non-DNA-binding domains. According to our previous analysis, we set the threshold to 2.0 and the false positive rate is less than 0.05. It shows that for non-DNA-binding domains, our method can recognize their non-binding with high accuracy.

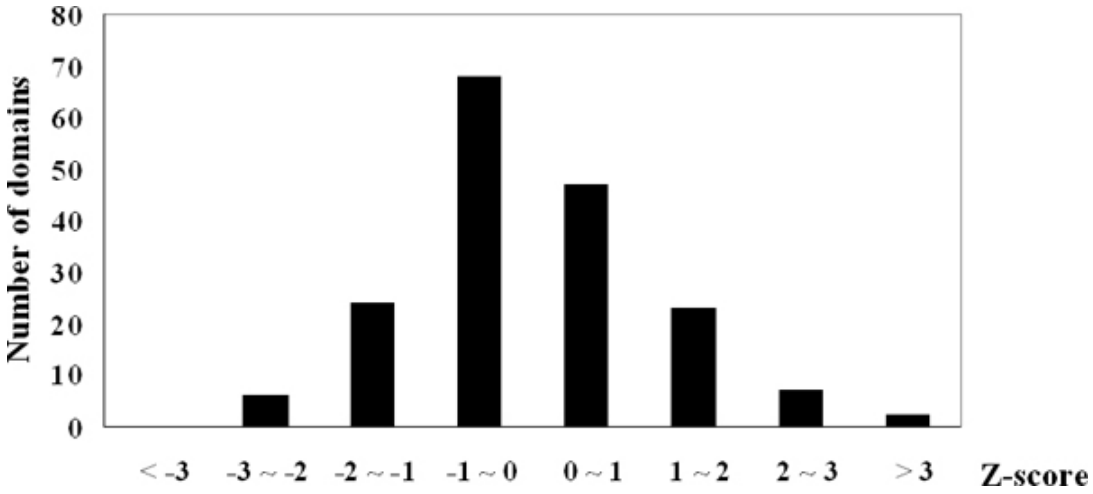

Figure 2.5.1.2. Distribution of z-score values of 177 non-DNA-binding domains.

2.5.2 Evolutionary conservation and interacting preference for identifying Protein-DNA interactions

Identifying DNA-binding domains

Proteins operate in biological processes by using their functional domains and the domains of the same families usually have similar functions. We applied our scoring functions to identify family members of a DNA-binding protein/domain. For each crystal structures of protein-DNA complex, we identified DNA-binding domains based on the domain definition of Structure Classification of Proteins (SCOP, version 1.71) [121]. To create a non-redundant and reasonable DNA-binding set for evaluation, we first select the domains, which have at least 50 residues and more than five contact residues. To remove the redundant DNA-binding domains, we applied the NCBI software BLASTCLUST to cluster highly similar sequences (sequence identity >90%) into one group. In each group, a DNA-binding domain which has maximal contact residue in this group is selected as the representative domain. We finally yield 69 representative DNA-binding domains.

The family members (according to the classification of the SCOP database) are aligned to their representative domain. Two protein-DNA interfaces are often different if their 20% contact residues are misaligned based on our observations. Here, we discarded the members if more than 20% misaligned contact residues. Each aligned member is scored by our scoring methods.

To show the statistical significance of the scores, we create 10,000 random domains for each representative domain by randomly mutating all contact residues of the

representative domain. We then translate the scores of the family members to *Z*-scores by

$$Z = \frac{s - \mu}{\delta}, \qquad \text{(Eq. 2.5.2.1)}$$

where s is the score of a member, μ and δ are the mean and the standard deviation, respectively, of 10,000 random domains. Figure 2.5.2.1 shows the distribution of Z scores in our scoring method. It shows that more than 80% members have statistic significant $Z > 2$ against random sets. This result indicates that the combination of the consensus and the interaction scoring function provides statistic meaning.

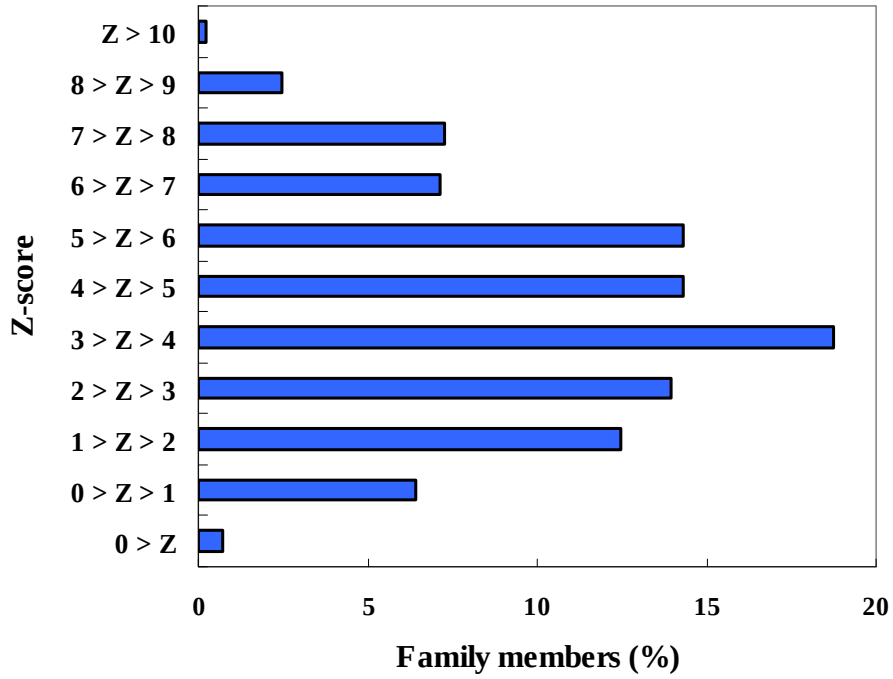

Figure 2.5.2.1. The distribution of *Z*-score of our scoring method.

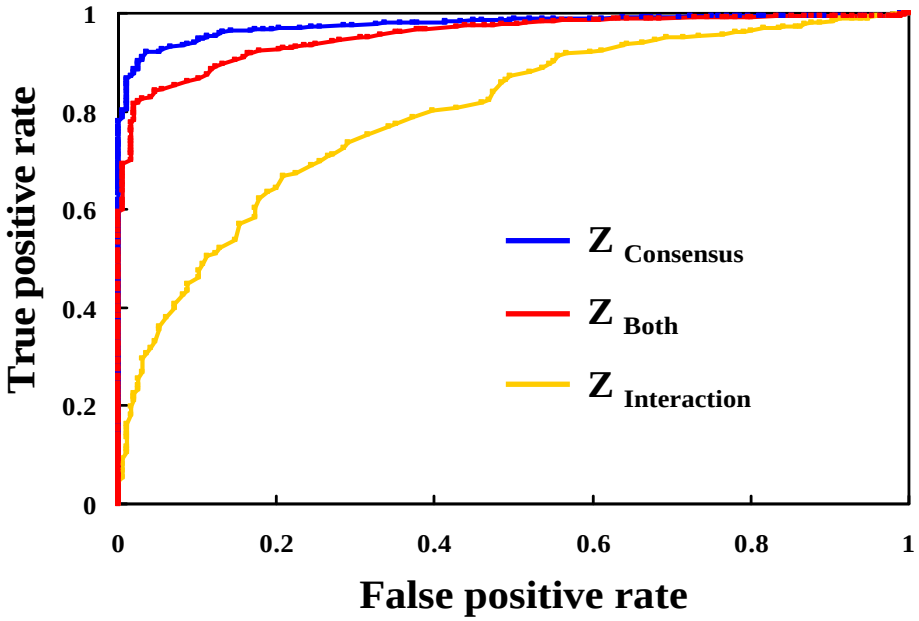

Figure 2.5.2.2. The ROC curves of *Z*-score on our scoring methods.

We further test the capability of the scoring methods to identify DNA-binding domains. The family members of all representative domains are used to be the positive set and the total number is 811 based on the SCOP database. The negative set was obtained by following steps. All representative domains are aligned to 250 public non-DNA-binding proteins [134] by using CE. An aligned domain of non-DNA-binding proteins is used to be a negative domain if the returned *z* value (defined by CE) of the alignment exceeds 3.7 and the misalign rate of contact residues is less than 20%. We totally obtained 196 negative cases. The overall performance is shown in Figure 2.5.2.2 by using ROC curves. The results show that the consensus score is the best and the interaction score is the worst to identify DNA-binding domains. It showed that the combination of the consensus and the interaction scoring function is acceptable to identify DNA-binding domains.

Free energy prediction between protein and DNAs

The hot spots of a protein are a set of individual residues which contributed the most binding free energy when interacting with other molecules [135] . The alanine scanning mutagenesis [136], which mutates a residue to alanine and measures the free energy change ($\Delta\Delta G$) of the mutation, is usually used to detect hot spots. To evaluate the capability of the scoring function on modeling the $\Delta\Delta G$, we obtained the point-mutation data of residues from the alanine scanning energetics database (ASEdb) [137]. Unfortunately, there are only two protein-DNA complexes (PDB code 1MNM and 1BDT) in ASEdb. We gathered all mutation data, which consists of 23 mutations of residues (non-DNA-contact residues are filtered out), from the two complexes.

For applying the scoring functions to these 23 mutation data, the protein of wild type complex is used as the template. We firstly align a protein sequence which is identical with the template to yield the wild type score S_{wt}. The one-point mutation protein sequence is then aligned to the template and obtain the mutation score S_{mt}. We used the energy gap (ΔS) between the wild-type (S_{wt}.) score and the mutation-type score (S_{mt}) to model the $\Delta\Delta G$ (i.e. S_{mt}-S_{wt}).

Figure 2.5.2.3 shows the correlation between experimental energies ($\Delta\Delta G$) and predicted energies (ΔS). The correlations of consensus and combination scoring methods are 0.38 and 0.6. This result indicates that the evolutionary conservation on DNA-contact residues is not sufficient to model the binding free energy between proteins and their binding DNAs. The interacting preference significantly improved the performance of only conservation score. These experimental results show that both evolutionary conservation and binding preference of DNA-contact residues play the key role for interactions between proteins and DNAs.

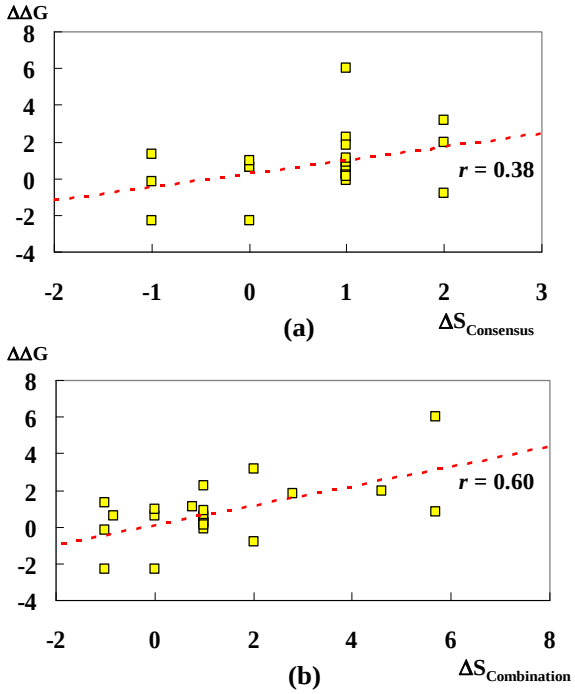

Figure 2.5.2.3. The correlation between ΔΔG and ΔS **(A)** Consensus **(B)** Combination.

2.6 Discussion

2.6.1 Compare the same SCOP family of homeodomain 1B8I-A

Figure 2.6.1.1 shows an example, which is the ultrabithorax homeodomain (Ubx) from *Drosophila melanogaster* (PDB entry 1B8I-A [138]) selected from 66 representative domains to described the characteristics of our method. The DNA is represented in green. 18

DNA-contact residues are presented as yellow stick and other residues are denoted as blue. The protein sequence is also presented and a contact residue is marked with an asterisk. For the alignment of the representative domain (1B8I-A) to the domains of its member, Figure 2.6.1.1 presents a nice case (PDB entry 1PUF-A), which is a homeobox protein hox-a9 from mouse [139]. We found that the contact residues is highly conserved in the aligned amino acids of the two domains and our scoring method shows this high z-score (z-score = 11.92). On the other hand, if we align 1B8I-A to 250 non-DNA-binding proteins, our method is able to discard the similar protein structures whose contact residues are not conserved (z-score = 0.58). Figure 2.6.1.1 shows an example of aligning 1B8I-A to 1BOB, which is histone acetyltransferase hat1 from *S. cerevisiae* in complex with acetyl coenzyme [140].

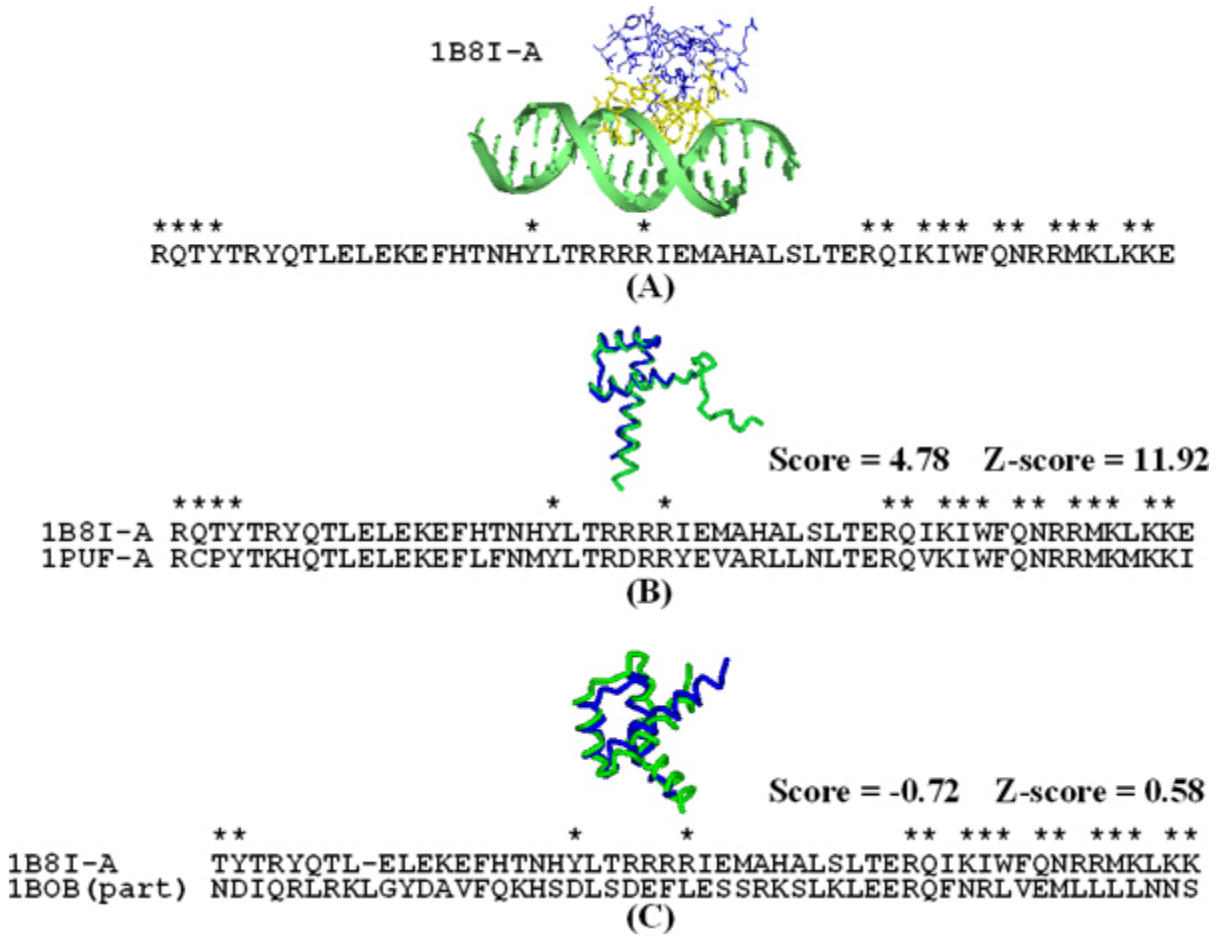

Figure 2.6.1.1. Searching results of the ultrabithorax homeodomain protein. Searching results using the homeotic Ubx/Exd/DNA ternary complex (PDB entry 1B8I-A) from Drosophila melanogaster as the query. **(A)** The contact residues of 1B8I-A complex are presented as stick (yellow). The sequence of 1B8I-A is shown and contact residues are marked with asterisks. **(B)** Structure alignment of 1B8I-A (blue) and 1PUF-A (green). The score is 4.78 and Z-score is 11.92 by our scoring method. **(C)** Structure alignment of 1B8I-A (blue) and non-DNA-binding protein 1BOB (green). Only the aligned structure/sequence of 1B8I-A and 1BOB are shown. We obtained score = -0.72 and Z-score = 0.58.

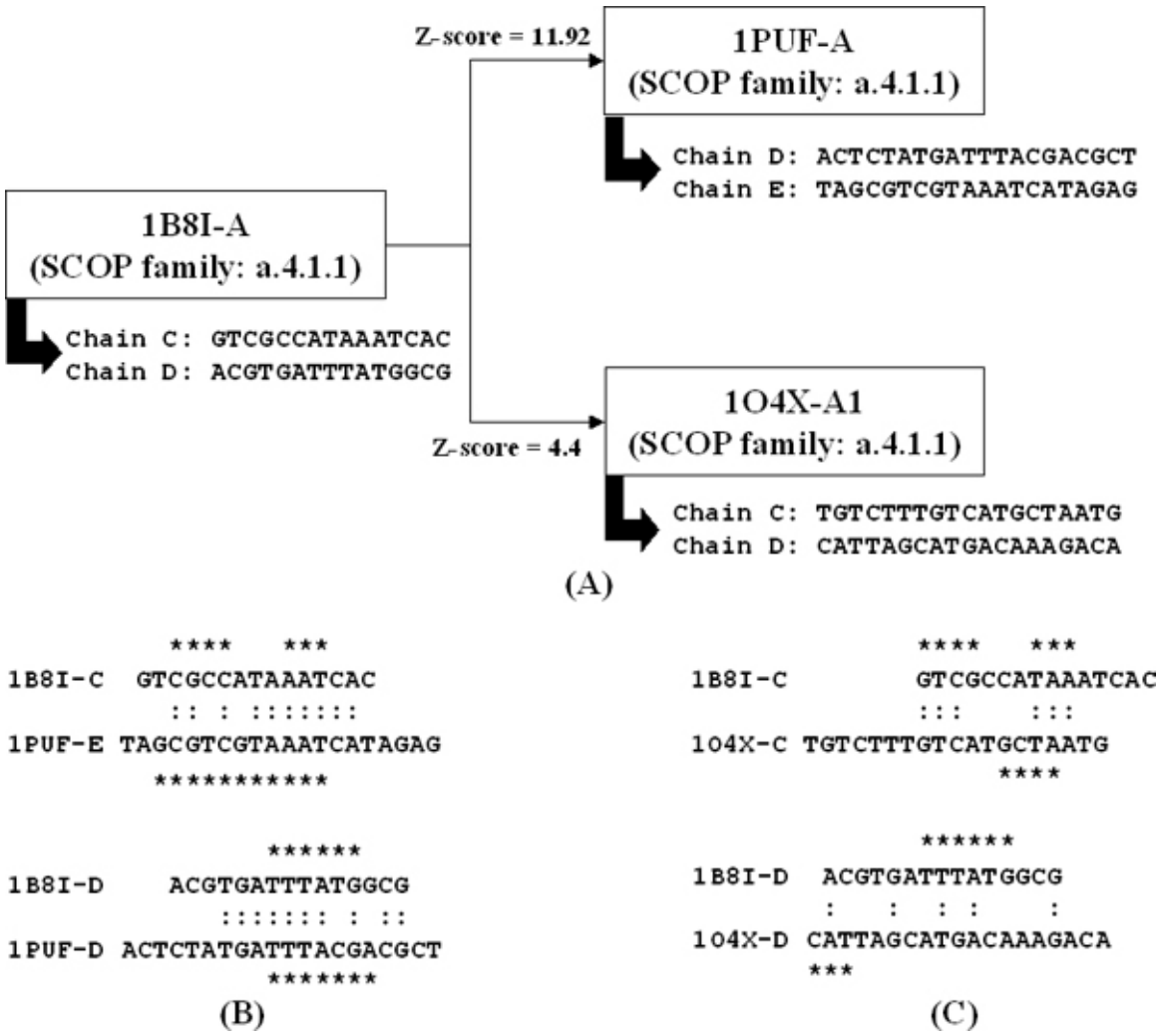

Figure 2.6.1.2. Comparison of bound DNA sequences of homologous proteins. The alignments of the bound DNA sequences of homologous proteins by using the homeotic ubx/exd/DNA ternary complex (PDB entry 1B8I-A) as the query. **(A)** The z-score values and the bound DNA sequences of the complex 1B8I (PDB entry 1B8I-C and 1B8I-D), 1PUF (PDB entry 1PUF-D and 1PUF-E), and 1O4X (PDB entry 1O4X-C and 1O4X-D). All sequences are from 5' to 3'. **(B)** Alignments of bound DNA sequences of the complexes 1B8I and 1PUF. A colon denotes an identical pair and an asterisk denotes a contact nucleotide (asterisks are marked above/below alphabets on the upper/lower sequence of the alignment, respectively). **(C)** Alignments of bound DNA sequences of the complexes 1B8I and 1O4X.

The z-score of DNA-binding domains in the same SCOP family may be variable for several representative domains (Figure 2.6.1.2(A)). The 1PUF-A and 1O4X-A1 (Oct-1 POU homeodomains from *Homo sapiens* [141]) are the members of the 1B8I-A representative domain. The *z*-scores are 11.92 (1PUF-A) and 4.4 (1O4X-A1) when 1B8I-A was used as the query (Figure 2.6.1.2(A)). The z-scores indicated that the contact residues between 1PUF-A and 1B8I-A are more conserved than the ones between 1O4X-A1 and 1B8I-A on contact residues interacting to the bases of the core binding site in the DNA.

To investigate variation of contact residues of DNA-binding domain in the same SCOP family, we compared the bound DNA sequences of two DNA-binding domains by aligning the double-strand sequences to each other. 1B8I-A binds two DNA sequences (i.e. PDB entry 1B8I-C and 1B8I-D) and 1O4X-A1 binds another two DNA sequences (PDB entry 1O4X-C and 1O4X-D). First we generated four pairing alignments: 1B8I-C and 1O4X-C; 1B8I-C and 1O4X-D; 1B8I-D and 1O4X-C; and 1B8I-D and 1O4X-D. We do not allow any gap insertion when aligning a-pairing DNA sequences. The alignments are obtained by sliding two sequences against each other until the best match is found. The alignment with the maximum number of identical aligned pairs is chosen, and as a result the alignment between 1B8I-C and 1O4X-C is the one chosen (Figure 2.6.1.2(C)). Then we adjust the alignment of the other DNA strand pairs (i.e. 1B8I-D and 1O4X-D) according to this best alignment (1B8I-C and 1O4X-C).

Figure 2.6.1.2(B) and Figure 2.6.1.2(C) show that the number of identical nucleotides between 1B8I-C and 1PUF-E [142] as well as 1B8I-D and 1PUF-D [142] is much higher than those of 1B8I-C and 1O4X-C [143] as well as 1B8I-D and 1O4X-D [144] for whole DNA sequences. At the same time, 11 identical contact nucleotides are obtained from the alignments of 1B8I-C and 1PUF-E as well as 1B8I-D and 1PUF-D; but two identical contact nucleotides are yielded from the alignments of 1B8I-C and 1O4X-C as well as 1B8I-D and 1O4X-D (the contact nucleotides are the nucleotides that interact with contact residues of protein). With respect to 1B8I-A, 1PUF-A and 1O4X-A1 are different not only in the DNA sequences they bind to but also in their DNA-binding sites. These results show that the members in the same SCOP family may have different DNA-binding models and that our method is able to detect the different Protein-DNA interactions based on the evolutionary conservation of DNA-contact residues.

We produced multiple protein sequence alignments of 13 homeodomains (Figure 2.6.1.3) selected from SCOP 1.71 using a multiple structure alignment tool, MUSTANG [145]. These domains were ranked by z-scores calculated by using our scoring method and the sequence of 1B8I-A as the query. According to z-scores, these 13 domains can be roughly divided into two groups, including the Ubx-like homeodomain colored in blue (e.g. PDB entry 9ANT-A (12.77), 1AHD-P (12.19), and 1SAN (11.96)) and the Oct-1 POU homeodomain colored in red (e.g. PDB entry 1E3O-C1 (6.40), 1GT0-C1 (6.38), and 1O4X-A1 (4.40)). Figure 2.6.1.3 shows that all Ubx-like homeodomains are significantly more conserved than Oct-1 POU homeodomains on contact residues (green). The Ubx homeodomain binds together with the extradenticle homeodomain (Exd) to recognize four DNA bases (ATAA) [138] based on four residues that are Ile47, Gln50, Asn51, and Met54, locating at α3 helix in the Ubx (gray columns in Figure 2.6.1.3). The z-scores of the domains are higher if they are conserved on these four residues, such as three antennapedia homeodomains and two homeobox protein

hox. These results show that contact residues interacting with bases in the DNA sequences are often conserved. This result is consistent to previous results [146] in which the homeodomain family was considered as a multi-specific family that consists of some subfamilies. This work concluded that members in the same subfamily bind DNA specifically but the members in different subfamilies recognize different DNA targets. In summary, we demonstrated the conservation of DNA-contact residues in DNA-binding domains.

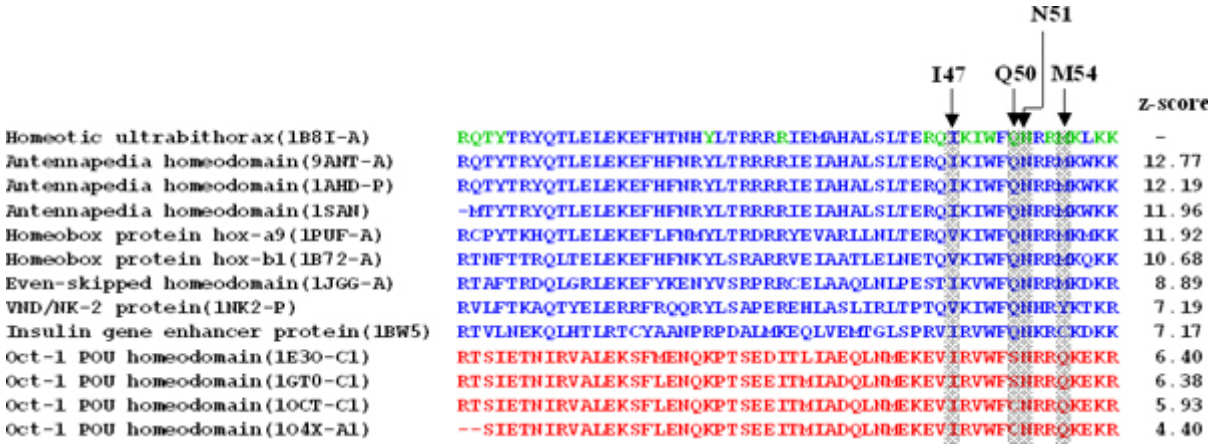

Figure 2.6.1.3. Multiple structure alignment of 13 homeodomain structures. The domains with similar DNA-binding specificities with 1B8I-A are shown in blue and others are red. The contact residues of 1B8I-A are marked green. The contact residues interacting to the bases of the core binding site in the DNA (ATAA) major groove are indicated gray.

2.6.2 Hormone receptor family

A hormone receptor is a receptor protein that binds a specific hormone and modulates numerous regulatory pathways [147, 148]. Based on the DNA-binding specificity of a protein, the hormone receptor family is classified into multi-specific families, which contain several subfamilies, by Luscombe *et al*. [146]. The members of a subfamily bind to specific DNA sequences; conversely, the members of different subfamilies target different DNA sequences. As shown in Figure 2.6.2.1, the hormone receptor family has two subfamilies that 50 members of Subfamily-1 target the sequence AGGTCA and 8 members of Subfamily-2 target the sequence AGAACA.

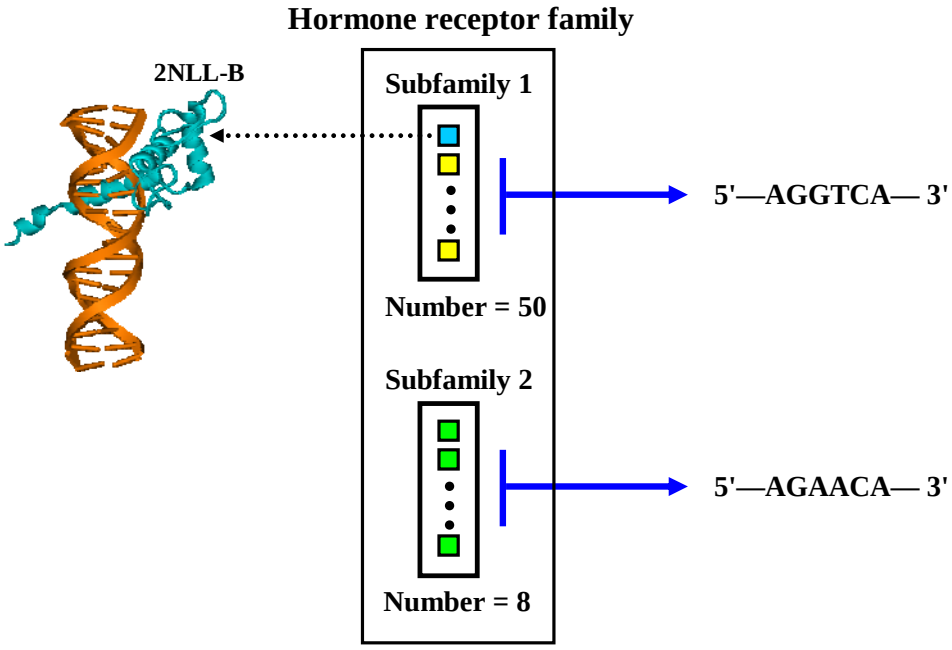

Figure 2.6.2.1. The target DNA sequences of two subfamilies in the hormone receptor family. The complex (2NLL-B) is selected to be the template.

All members of both subfamilies were obtained from PDB and SWISSPROT through homology searching by a representative protein (the detail was described in Luscombe *et al* [146]). To see how the contact residues affect the binding specificity of the hormone receptor family, we applied our combined scoring function to the family. First, we used the protein-DNA complex of thyroid hormone receptor β (PDB code 2NLL chain B) [149] from the Subfamily-1 to be a template. The members of the two subfamilies were then aligned to the template. Our combined scoring function is used to score each aligned contact residue. Finally, for each position of contact residues in the two subfamilies, Figure 2.6.2.2 shows the average scores of the Subfamily-1 (blue) and Subfamily-2 (red). The x-axis presents the contact residue with its residue number (in PDB) of the template.

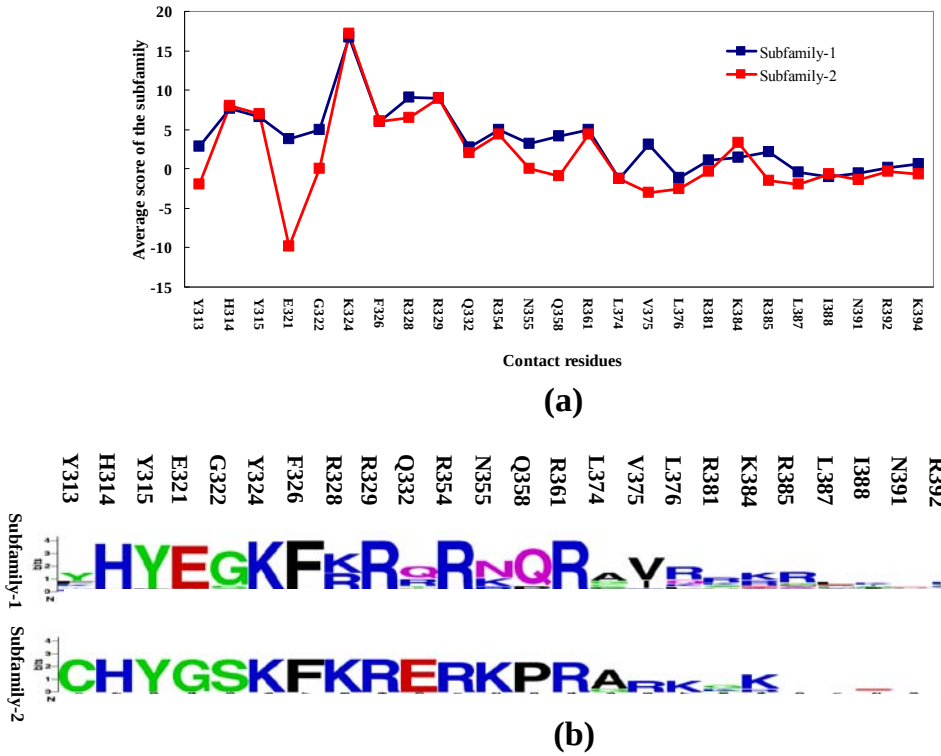

Figure 2.6.2.2. (A) The average score of each contact residues in two subfamilies. **(B)** The conservation of each contact residue in two subfamilies. The logo was created by WebLogo [150].

We observed three things. First, Subfamily-1 obtained higher overall score (by summing all scores of each contact residue) than Subfamily-2, indicating that the template is more similar with the members of Subfamily-1 than the members of Subfamily-2. Second, the scores of more than half contact residues are roughly equal. We found that these DNA-contact residues are conserved in both subfamilies of the hormone receptor family. Third, the score of Subfamily-1 and Subfamily-2 is obviously different at the contact residue Glu321. As shown in Figure 2.6.2.1 the Glu321 accepts a hydrogen bond from cytosine which is base-paring with the third base in the target sequence (AG**G**TCA) (Figure 2.6.2.3). However, the residue 321 in the members of Subfamily-2 is glycine which do not interact with any bases [146]. The target DNA sequences of these two subfamilies (Subfamily-1: AG**G**TCA; Subfamily-2: AG**A**ACA) are different. These results demonstrate that our combination scoring function is able to reflect binding specificity of the hormone receptor to discriminate these two subfamilies.

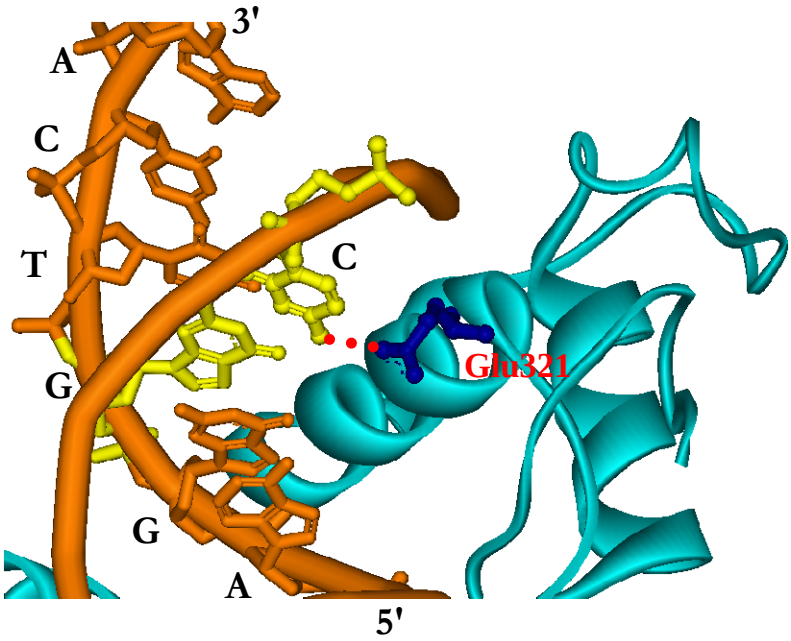

Figure 2.6.2.3. The hydrogen bond between Glu321 and DNA

2.7 Conclusion

3D-regulog based approach is proposed for modeling protein-DNA interactions and binding model. We proposed a structure template-based method which used a novel scoring function to identify potential protein-DNA interactions, such method has the advantage of increasing crystal structures of protein-DNA complexes. Furthermore, the method also reveals the structure information of identified protein-DNA binding partners, we found that the feature, evolutionary conservation of DNA-contact residues, is helpful to identify DNA-binding domains. By using the scoring function based on such a feature, we successfully identified 66 DNA-binding domain families, also identify the different DNA-binding behaviors of proteins in the same SCOP family.

The proposed scoring method which combined consensus information of DNA-contact residues and the preference of amino acid and DNA-bases is showed good performance in modeling protein-DNA interactions and good correlations between the scores and the binding free energy of protein-DNA complexes. The proposed method which is a residue-based approach has more potential than other atom-based approach for homology modeling of proteins.

2.8 Future work

For measuring the evolutionary conservation of DNA-contact residues, a position

specific scoring matrix (PSSM) which obtained by homological proteins of a template proteins should be used to improve the accuracy. The information of water-mediated bond and electrostatic interactions between amino acids and nucleotides will be incorporated into the knowledge-based scoring matrix. For detecting possible transcription factor binding sites, more transcription factors which have crystal structures of protein-DNA complexes will be used to as the template. The high-score region predicted by our scoring method in promoter regions will be further verified. The proposed method could be applied to predict the position weight matrix (PWM) of a template protein, selecting high-score DNA sequences using the scoring function, the PWM could be constructed by selected sequences.

Chapter 3:
Protein-RNA docking

3.1 Introduction

RNA is often used as an efficient drug target for some pathogenic therapies. For example, some antibiotics target RNA in the bacterial ribosome while treating bacterial diseases clinically and experimentally [151]. In addition, there are at least three advantages [152] to target RNA instead of proteins: 1) Inhibitors of RNA often have less side effects. Proteins that have similar substrates are difficult to be inhibited specifically (e.g. ATP); 2) RNA has more accessible sites for interacting with inhibitors; 3) Inhibitors of RNA usually have less drug resistance, because functional domains of RNA are often more highly conserved than active sites of proteins.

Various scoring functions have been developed for finding inhibitors of protein targets, including knowledge-based [153, 154], empirical [155, 156], physics-based [157, 158], and solvent-based scoring functions [159]. Most of scoring functions are designed for protein targets and do not consider properties of RNA. It is difficult to predict reliable conformations of ligands if we dock ligands into RNA targets by using these scoring functions. Therefore, a reliable scoring function is required to find novel inhibitors of RNA targets.

GEMDOCK is a docking/screening tool which achieved high accuracies on some benchmarks [160-162] and successfully identified novel substrates or inhibitors for some targets [163, 164]. The GEMDOCK used a soft energy function and a generic evolutionary method for flexible docking. The GEMDOCK energy function consists of electrostatic, steric, and hydrogen-bonding potentials. The latter two terms use a linear model that is simple and recognizes potential inhibitors rapidly. Based on these advantages, we selected GEMDOCK as the program for docking ligand into RNA targets. In this study, we added the atom types that are specific to nucleotides in the scoring function of GEMDOCK. The new scoring function is termed as “GemRNA”. We tested the performance of GemRNA on the public set (38 RNA-ligand complexes). The results show that GemRNA could model RNA-ligand binding reliably.

3.2 Method

3.2.1 GEMDOCK Parameters

Table 3.2.1.1 indicates the setting of GEMDOCK parameters, such as initial step sizes, family competition length (L = 2), population size (N = 300), and recombination probability (pc = 0.3) in this work. The GEMDOCK optimization stops when either the convergence is below certain threshold value or the iterations exceed a maximal preset value which was set to 70. Therefore, GEMDOCK generated 1200 solutions in one generation and terminated after it exhausted 84000 solutions in the worse case. These parameters were decided after experiments conducted to recognize complexes of test docking systems with various values.

Table 3.2.1.1. Parameters of GEMDOCK

Parameter	Value of parameters
Initial step sizes	$\sigma = 0.8$, $\psi = 0.2$ (in radius)
Family competition length	$L = 2$
Population size	$N = 300$
Recombination rate	$p_c = 0.3$
# of the maximum generation	70

3.2.2 Scoring Function for RNA-ligand Docking

In this work, we used an empirical scoring function given as

$$E_{tot} = E_{inter} + E_{intra} + E_{penal} \qquad \text{(Eq. 3.2.2.1)}$$

(3.2.2.1)

where E_{inter} and E_{intra} are the intermolecular and intramolecular energy, respectively, E_{penal} is a large penalty value if the ligand is out of range of the search box. E_{penal} is set to 10000.

The intermolecular energy is defined as

$$E_{inter} = \sum_{i=1}^{lig} \sum_{j=1}^{RNA} [F(r_{ij}^{B_{ij}}) + 332.0 \frac{q_i q_j}{4 r_{ij}}] \qquad \text{(Eq. 3.2.2.2)}$$

where r_{ij} is the distance between the atoms i and j; q_i and q_j are the formal charges and 332.0 is a factor that converts the electrostatic energy into kilocalories per mole. The lig and RNA denote the numbers of the heavy atoms in the ligand and RNA, respectively. The formal charge of a receptor and ligand atom is indicated in Table 3.2.2.1. $F(r_{ij}^{B_{ij}})$ is a simple atomic pair-wise potential function (Figure 3.2.2.1) modified from previous works

[155, 165] and given as

$$F(r_{ij}^{B_{ij}}) = \begin{cases} V_6 - \dfrac{V_6 r_{ij}^{B_{ij}}}{V_1} & \text{if } r_{ij}^{B_{ij}} \le V_1 \\ \dfrac{V_5(r_{ij}^{B_{ij}} - V_1)}{V_2 - V_1} & \text{if } V_1 < r_{ij}^{B_{ij}} \le V_2 \\ V_5 & \text{if } V_2 < r_{ij}^{B_{ij}} \le V_3 \\ V_5 - \dfrac{V_5(r_{ij}^{B_{ij}} - V_3)}{V_4 - V_3} & \text{if } V_3 < r_{ij}^{B_{ij}} \le V_4 \\ 0 & \text{if } r_{ij}^{B_{ij}} > V_4 \end{cases} . \qquad \text{(Eq. 3.2.2.3)}$$

$r_{ij}^{B_{ij}}$ is the distance between the atoms *i* and *j* with the interaction type B_{ij} forming by the pairwise heavy atoms between ligands and RNAs where B_{ij} is either a hydrogen bond or a steric state. In this atomic pair-wise model, these two potentials are calculated by the same function form but with different parameters, $V_1,\ldots,V_6$ given in Figure 3.2.2.1. The energy value of a hydrogen bond should be larger than the one of the steric potential. In this model, the atom is divided into four different atom types (Table 3.2.2.1): donor, acceptor, both, and nonpolar. A hydrogen bond can be formed by the following atom-pair types: donor-acceptor (or acceptor-donor), donor-both (or both-donor), acceptor-both (or both-acceptor), and both-both. Other atom-pair combinations are to form the steric state.

The intramolecular energy of a ligand is

$$E_{intra} = \sum_{i=1}^{lig} \sum_{j=i+2}^{lig} F\left(r_{ij}^{B_{ij}}\right) + \sum_{k=1}^{dihed} A[1 - \cos(m\theta_k - \theta_0)] \qquad \text{(Eq. 3.2.2.4)}$$

where $F\left(r_{ij}^{B_{ij}}\right)$ is defined as Equation 4 except that the value is 1000 to discard unreasonable conformations when $r_{ij}^{B_{ij}} <$ 2.0 Å and *dihed* is the number of rotatable bonds. We followed the work of Gehlhaar et al. [155] to set the values of A, m, and θ_0. For the $sp^3 - sp^3$ bond A, m, and θ_0 are set to 3.0, 3, and π; and A = 1.5, m = 6, and θ_0 = 0 for the $sp^3 - sp^2$ bond

Table 3.2.2.1. Atom formal charge of GEMDOCK

Formal charge	Atom name
Receptor:	
0.5	N atom in His (ND1 & NE2) and Arg (NH1 & NH2)
0.5	O atom in RNA (OP1&OP2)
-0.5	O atom in Asp (OD1 & OD2) and Glu (OE1 & OE2)
1.0	N atom in Lys (NZ)
2.0	metal ions (MG, MN, CA, ZN, FE, and CU)
0	other atoms
Ligand:	
0.5	N atom in $-C(NH_2)_2^+$
-0.5	O atom in $-COO^-$, $-PO_2^-$, $-PO_3^-$, $-SO_3^-$, and $-SO_4^-$
1.0	N atom in $-NH_3^+$ and $-N^+(CH_3)_3$
0	other atoms

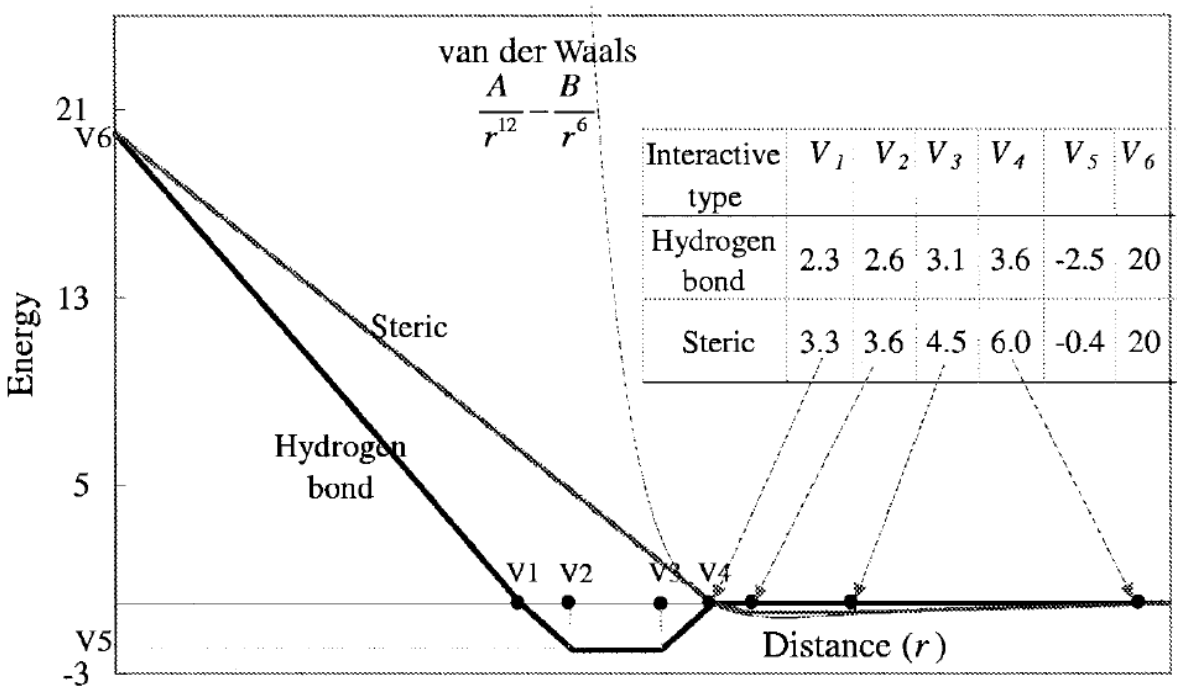

Figure 3.2.2.1. The linear energy function of the pair-wise atoms for the steric interactions and hydrogen bonds in GEMDOCK (bold line) with a standard Lennard-Jones potential (light line).

3.2.3 GEMDOCK algorithm details

In the following subsections, we present the details of our approach for molecular docking (Figure 3.2.3.1). The core idea of our evolutionary approach was to design multiple operators that cooperate using the family competition model which is similar to a local search procedure. We designed a new rotamer-based mutation operator for reducing the search space of ligand structure conformations, and used a differential evolution operator[166] for reducing the disadvantages of Gaussian and Cauchy mutations. GEMDOCK is a nearly automatic docking tool for generating all experimental variables, and may serve as a flexible or hybrid docking program. First we specified the coordinates of ligand and RNA atoms, the ligand binding area, atom formal charge (Table 3.2.2.1), and atom types (Table 3.2.3.1). Crystal coordinates of the ligand and RNA atoms were taken from the Protein Data Bank, and were separated into different files. GEMDOCK then automatically determined the center of the receptor and the search cube of a binding site according to the maximum and minimum of coordinates of these selected RNA atoms.

Table 3.2.3.1. Atom types of GEMDOCK

Atom type	Atom name of PDB
Donor	N atoms in U(N3) and G(N1)
Acceptor	O atoms in all RNA ribose (OP1&OP2&O5&O4&O3),C(O2), U(O2&O4),G(O6) N atoms in A(N1&N3&N7), C(N3),G(N7&N3)
Both	O atoms in all RNA ribose(O2) and N atoms in A(N6), C(N4),G(N2)
Nonpolar	other atoms (such as carbon and phosphorus)

After GEMDOCK prepares the ligand and RNA, GEMDOCK works as follows: Randomly generate a starting population with N solutions by initializing the orientation and conformation of the ligand relating to the center of the receptor. Each solution is represented as a set of three *n*-dimensional vectors $\left(x^i, \sigma^i, \psi^i\right)$, where n is the number of adjustable variables of a docking system and $i = 1,\ldots,N$ where N is the population size. The vector x represents the adjustable variables to be optimized in which x_1, x_2, and x_3 are the 3-dimensional location of the ligand; x_4, x_5, and x_6 are the rotational angles; and from x_7 to x_n are the twisting angles of the rotatable bonds inside the ligand. σ and ψ are the step-size vectors of decreasing-based Gaussian mutation and self-adaptive Cauchy mutation. In other words, each solution x is associated with some parameters for stepsize control. The initial values of x_1, x_2, and x_3 are randomly chosen from the feasible box, and the others, from x_4 to x_n, are randomly chosen from 0 to 2π in radians. The initial step sizes σ is 0.8 and ψ is 0.2. After GEMDOCK initializes the solutions, it enters the main evolutionary

loop which consists of two stages in every iteration: decreasing-based Gaussian mutation and self-adaptive Cauchy mutation. Each stage is realized by generating a new quasi-population (with N solutions) as the parent of the next stage. As shown in Figure 14, these stages apply a general procedure "FC_adaptive" with only different working population and the mutation operator.

GEMDOCK can be a flexible docking method or a hybrid docking method which evolves simultaneously both flexible and rigid conformation solutions of a ligand. GEMDOCK is a flexible docking tool if it evolves the conformation variables ($x_7,\ldots,x_n$) of each solution in a population. On the other hand, GEMDOCK is a hybrid approach if the conformation variables of part of solutions (e.g., ηN solutions) are fixed and set to the values of a native binding state. In this work, η is 0.2 when GEMDOCK is a hybrid method which simultaneously evolves fix and flexible ligand conformations by the recombination operators.

The FC_adaptive procedure (Figure 3.2.3.1) employs two parameters, namely, the working population (P, with N solutions) and mutation operator (M), to generate a new quasi-population. The main work of FC_adaptive is to produce offspring and then conduct the family competition. Each individual in the population sequentially becomes the "family father." With a probability pc, this family father and another solution that is randomly chosen from the rest of the parent population are used as parents for a recombination operation. Then the new offspring or the family father (if the recombination is not conducted) is operated by the rotamer mutation or by differential evolution to generate a quasi offspring. Finally, the working mutation is operates on the quasi offspring to generate a new offspring. For each family father, such a procedure is repeated L times called the family competition length. Among these L offspring and the family father, only the one with the lowest scoring function value survives. Since we create L children from one "family father" and perform a selection, this is a family competition strategy. This method avoids the population prematureness but also keeps the spirit of local searches. Finally, the FC_adaptive procedure generates N solutions because it forces each solution of the working population to have one final offspring.

In the following, genetic operators are briefly described. We use $a=\left(x^a,\sigma^a,\psi^a\right)$ to represent the "family father" and $b=\left(x^b,\sigma^b,\psi^b\right)$ as another parent. The offspring of each operation is represented as $c=\left(x^c,\sigma^c,\psi^c\right)$. The symbol x_j^s is used to denote the j th adjustable optimization variable of a solution s, $\forall j\in\{1,...,n\}$.

1. Initial the protein and the ligand as follows:
   (a) Determining the size and location of the ligand binding site and removing the structure water molecules.
   (b) Assigning the atom type (Table 12) and the atom formal charge (Table 11) of a ligand and a protein.
2. Fix the location of the receptor and Let g = 1. Randomly generate initial population, $P(g)$, with N solutions by initializing the orientation and conformation of a ligand related to the receptor.
3. Evaluate the scoring fitness of each solution in the population $P(g)$.
4. Generate a new quasi-population, $P1(g)$, with N solutions by applying FC_Adaptive with $P(g)$ and *decreasing-based Gaussian mutation* (*Mdg*).
5. Generate a new quasi-population, *Pnext*, with N solutions by applying FC Adaptive with $P1(g)$ and *self-adaptive Cauchy mutation* (*Mc*). Let g =

Figure 3.2.3.1. The main steps of GEMDOCK for molecular docking.

3.2.4 Recombination Operators

GEMDOCK implemented modified discrete recombination and intermediate recombination. [167] A recombination operator selected the "family father (a)" and another solution (b) randomly selected from the working population. The former generates a child as follows:

$$x_j^c = \begin{cases} x_j^a \text{ with probability } 0.8 \\ x_j^b \text{ with probability } 0.2 \end{cases} \qquad \text{(Eq. 3.2.4.1)}$$

The generated child inherits genes from the "family father" with a higher probability 0.8. Intermediate recombination works as:

$$\omega_j^c = \omega_j^a + \beta\left(\omega_j^b - \omega_j^a\right)/2, \qquad \text{(Eq. 3.2.4.2)}$$

where ω is σ or ψ based on the mutation operator applied in the FC_adaptive procedure. The intermediate recombination only operated on step-size vectors and the modified discrete recombination was used for adjustable vectors (x).

3.2.5 Mutation Operators

After the recombination, a mutation operator, the main operator of GEMDOCK, is applied to mutate adjustable variables (x).

Gaussian and Cauchy Mutations: Gaussian and Cauchy Mutations are accomplished by first mutating the step size (ω) and then mutating the adjustable variable x:

$$\omega'_j = \omega'_j A(\cdot), \qquad \text{(Eq. 3.2.5.1)}$$

$$x'_j = x_j + \omega'_j D(\cdot), \qquad \text{(Eq. 3.2.5.2)}$$

where ω_j and x_j are the i th component of ω and x, respectively, and ω_j is the respective step size of the x_j where ω is σ or ψ. If the mutation is a self-adaptive mutation, $A(\cdot)$ is evaluated as $\exp\left[\tau' N(0,1) + \tau N_j(0,1)\right]$ where $N(0,1)$ is the standard normal distribution, $N_j(0,1)$ is a new value with distribution $N(0,1)$ that must be regenerated for each index j. When the mutation is a decreasing-based mutation $A(\cdot)$ is defined as a fixed decreasing rate $\gamma = 0.95$. $D(\cdot)$ is evaluated as $N(0,1)$ or $C(1)$ if the mutation is, respectively, Gaussian mutation or Cauchy mutation. For example, the self-adaptive Cauchy mutation is defined as

$$\psi_j^c = \psi_j^a \exp\left[\tau' N(0,1) + \tau N_j(0,1)\right], \qquad \text{(Eq. 3.2.5.3)}$$

$$x_j^c = x_j^a + \psi_j^c C_j(t). \qquad \text{(Eq. 3.2.5.4)}$$

We set τ and τ' to $\left(\sqrt{2n}\right)^{-1}$ and $\left(\sqrt{2\sqrt{n}}\right)^{-1}$, respectively, according to the suggestion of evolution strategies. [167] A random variable is said to have the Cauchy distribution $(C(t))$ if it has the density function: $f(y;t) = \dfrac{t/\pi}{t^2 + y^2}$, $-\infty < y < \infty$. In this paper t is set to 1. Our decreasing-based Gaussian mutation uses the step-size vector σ with a fixed decreasing rate γ = 0.95 and works as

$$\sigma^c = \gamma \sigma^a, \qquad \text{(Eq. 3.2.5.5)}$$

$$x_j^c = x_j^a + \sigma^c N_j(0,1). \qquad \text{(Eq.3.2.5.6)}$$

Differential Evolution: An offspring of differential evolution is generated as

$$x_j^c = \begin{cases} u_j^m & \text{if rand}[0,1) \le CR \\ x_j^a & \text{otherwise} \end{cases} \qquad \text{(Eq. 3.2.5.7)}$$

and

$$u_j^m = x_j^a + K\left(x_j^b - x_j^c\right), \qquad \text{(Eq. 3.2.5.8)}$$

where a is the "family father"; b and c are two solutions randomly selected from the working population subjected to $a \neq b \neq c$. In this work, K and CR are set to 0.5 and 0.9, respectively.

Rotamer-Mutation: This operator is only used for x7 to xn to find the conformations of the rotatable bonds inside the ligand. For each ligand, this operator mutates all of the rotatable angles according to the rotamer distribution and works as:

$$x_j = \gamma_{ki} \text{ with probability } p_{ki}, \qquad \text{(Eq. 3.2.5.9)}$$

where γ_{ki} and p_{ki} are the angle value and the probability, respectively, of *i*th rotamer of *k*th bond type including $sp^3 - sp^3$ and $sp^3 - sp^2$ bond. The values of γ_{ki} and p_{ki} are based on the energy distributions of these two bond types.

3.3 Results and Discussion

3.3.1Test data set and docking protocols

To evaluate the strengths and limitations of GEMDOCK, we tested the program on a highly diverse dataset of 38 RNA-ligand complexes (Tables 3.3.1.1 and 3.3.1.2) Detering and Varani [168]. 14 of them are NMR structures, and the remaining complexes are crystallographic structures. Crystal coordinates of the ligands and RNA atoms were taken from the Protein Data Bank (PDB), and were separated into different files. The ligands consist of small, flexible, and cyclic molecules (30-40 heavy atoms, 3-4 rings). For the NMR structures, we selected the ligands of the first model as the native conformation unless a different structure was specified as the minimum energy structure in the PDB file.

Our program then assigned the atom formal charge and atom type (i.e., donor, acceptor, both, or nonpolar) for each atom of both the ligand and RNA. The bond type (sp3－sp3, sp3－sp2, or others) of a rotatable bond inside a ligand was also assigned. These variables were used in Equation 3.3.2.1 to calculate the scoring value of a docked conformation (see Materials and Methods).

When preparing the RNA receptors, the size and location of the ligand binding site was determined by considering the RNA atoms located ＜ 10 Å from each ligand atom. The metal atoms in the active site were also retained. We duplicated Jones' work [169] in that all structure water molecules were removed. GEMDOCK then automatically decided the search cube of a binding site based on the maximum and minimum values of coordinates among these selected RNA atoms.

Table 3.3.1.1. PDB codes with ligand names of the 38 test complexes

1AJU(ARG), 1AM0(AMP), 1BYJ(GET), 1EHT(TEP), 1EI2(NMY), 1FIT(ROS), 1F27(BTN), 1FJG(PAR), 1FJG(SCM), 1FJG(SRY), 1FMN(FMN), 1HNW(TAC), 1HNX(PCY), 1HNZ(HYG), 1J7T(PAR), 1JZX(GLY), 1JZY(ERY), 1JZZ(ROX), 1K01(CLM), 1K8A(CAI), 1K9M(TYK), 1KD1(SPR), 1KOC(ARG), 1KOD(CIR), 1LC4(TOY), 1LVJ(PMZ), 1M90(SPS), 1MWL(GET), 1NEM(BDG_NEB_BDR_IDG), 1NJM(SPS), 1NJN(SPS), 1NJO(PPU), 1NWY(ZIT), 1OND(TAO), 1PBR(PA1_PA2_PA3_IDG), 1QD3(RIB_IDG_BDG_CYY), 1TOB(TOA_TOC_TOB), 2TOB(TOA_TOC_2TB)

Table 3.3.1.2. GEMDOCK results of 38 complexes

RMSD (Å)	PDB codes with ligand names
≤ 0.5	1FIT(ROS), 1FJG(SCM)
> 0.5,≤ 1.0	1BYJ(GET), 1FJG(SRY), 1F27(BTN) 1QD3(RIB_IDG_BDG_CYY) 2TOB(TOA_TOC_2TB)
> 1.0 ≤ 1.5	1FJG(PAR), 1LC4(TOY) 1KOD(CIR) 1MWL(GET) 1J7T(PAR) 1NJO(PPU) 1AM0(AMP) 1JZX(GLY)
> 1.5,≤ 2.0	1HNZ(HYG) 1HNW(TAC)
> 2.0,≤ 2.5	1PBR(PA1_PA2_PA3_IDG) 1FMN(FMN)
> 2.5,≤ 3.0	1EHT(TEP)
> 3.0	1K01(CLM) 1K8A(CAI) 1LVJ(PMZ) 1KD1(SPR) 1NWY(ZIT) 1HNX(PCY) 1NEM(BDG_NEB_BDR_IDG) 1JZY(ERY) 1TOB(TOA_TOC_TOB) 1EI2(NMY) 1JZZ(ROX) 1OND(TAO) 1K9M(TYK) 1M90(SPS) 1NJM(SPS) 1NJN(SPS) 1AJU(ARG)

The root mean square deviation (RMSD) of heavy atom positions between the docked conformation and the crystal structure was used to assess the accuracy of docking predictions. The successful percentage (the proportion of docking experiments that found a solution within 2.5 Å RMSD) was determined to evaluate the robustness of a docking method. The RMSD commonly used in previous studies [169, 170] is defined as

$$\left\{\sum_{i=1}^{M}\left[(X_i - x_i)^2 + (Y_i - y_i)^2 + (Z_i - z_i)^2\right]/M\right\}^{1/2} \qquad \text{(Eq. 3.3.3.1)}$$

where M is the heavy atom number of a ligand; (X_i, Y_i, Z_i) and (x_i, y_i, z_i) are the coordinates of the ith atom of X-ray crystal and docked structures, respectively. And an arbitrary value of a 2.5Å rmsd from the experimental structure was chosen to separate successful and unsuccessful docking poses [168].

3.3.2 Overall accuracy on 38 complexes

The overall accuracy of GEMDOCK in predicting the docked ligand conformations of 38 test complexes is shown in Table 3.3.2.1. All results are derived from 20 independent docking runs, and the docked lowest-energy structure was considered for each test case. On average, GEMDOCK took 305 seconds for a docking run on a Pentium 1.4 GHz personal computer with a single processor.

In the test set, we found the docking poses of 20 compounds were near that of native ligands (≦ 2.5 Å). GemRNA achieved 53% success in identifying the experimental binding model (Table 3.3.2.1) The successful rates of the successful cases were shown in Figure

3.3.2.1, 2 of them (1F27 and 1FJGscm) exceed 50%, 3 of them are between 20% and 40%, and the others are less than 20%. The case that has the highest successful rate was shown in Figure 3.3.2.2.

Table 3.3.2.1. Features describing the properties of RNA, ligand, and interactions between RNA and ligands

Feature
Number of metal ions around native ligand within 4.5 Å
Number of water atoms around native ligand within 4.5 Å
Number of atoms of native ligand
Number of rotatable bonds of native ligand
Number of hydrogen-bonds between RNA and native ligand
Number of atoms around native ligand within 4.5 Å

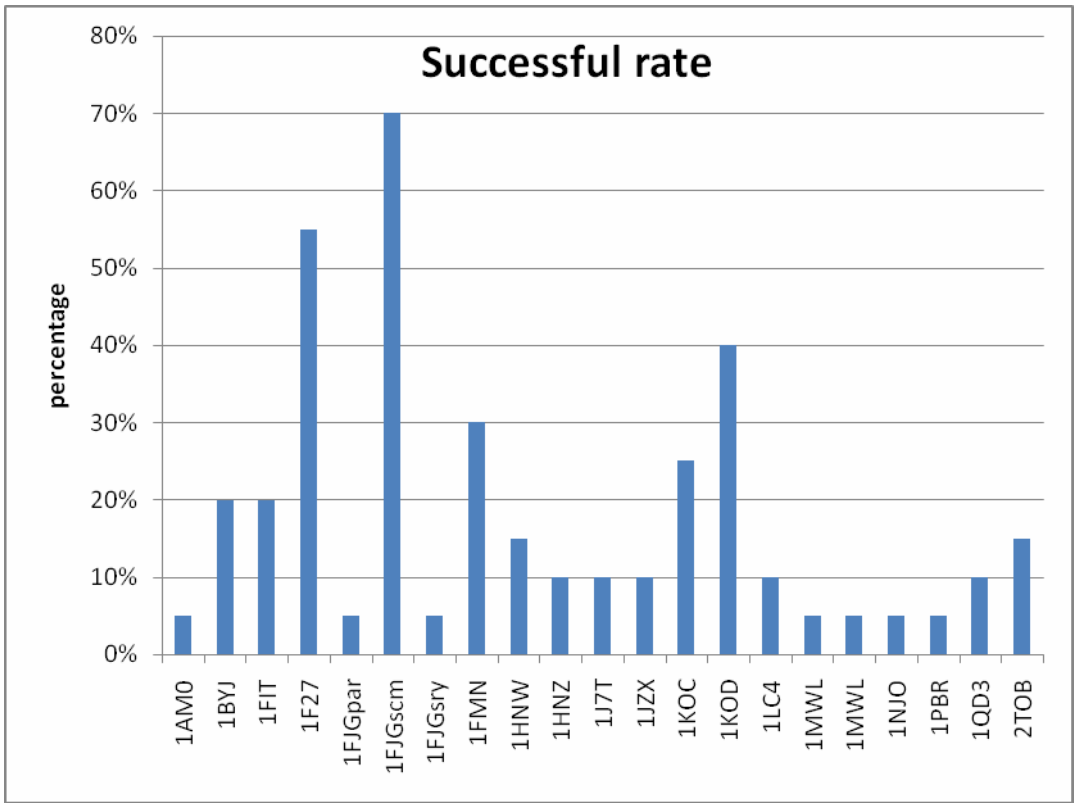

Figure 3.3.2.1. The successful rates of the successful cases. 2 of them (1F27 and 1FJGscm) exceed 50%, 3 of them are between 20% and 40%, and the others are less than 20%.

For analyzing what factors affect the docking results, we generated 6 features describing the properties of RNA, ligand, and interactions between RNA and ligands (shown in Table

3.3.2.1). 38 complexes were divided into 2 classes, successful cases and unsuccessful cases, according to the 2.5 Å cutoff. The decision tree method [171] was applied to select the most discriminative feature for discriminating successful cases and unsuccessful cases (Figure 3.3.2.3). In Figure 3.3.2.3, the decision model shows that the most discriminative feature is the number of atoms around the native ligand within 4.5 Å and the threshold is 110. The distribution of this feature of the 38 complexes was shown in Figure 3.3.2.4 The cases whose number of this feature is less than 110 failed in docking (i.e. 1NJN, 1NJM, 1AJU, 1JZY, 1NWY, 1OND, and 1K01). Based on the observations, we could find that the native ligands that bind RNA weakly and have few interactions with RNA often failed in docking. An example was shown in Figure 3.3.2.5. In these cases, the docking poses often preferred the conformations that formed stable interactions with RNA.

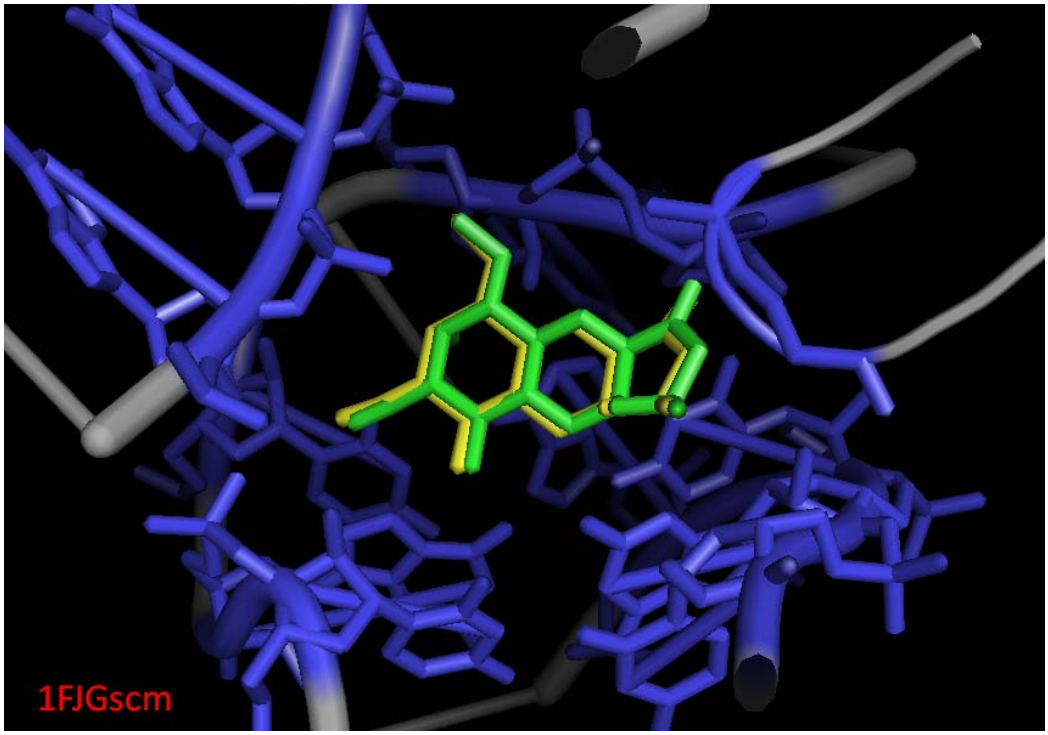

Figure 3.3.2.2. A successful case. The RMSD value between the native conformation (green) and the docked conformation (yellow) is 0.39Å. The ligand contains 23 heavy atoms, 6 single bonds, and 172 atoms around the native ligand within 4.5 Å, and has the highest successful rate of docking (78%).

Except for these weak binders, GemRNA performance was somewhat influenced by ligand parameters such as size and flexibility. In Figure 3.3.2.3, 4 cases whose numbers of atoms of native ligands exceed 50 failed in docking due even if their native ligands have stable interactions with RNAs. For the large and flexible ligands, GemRNA failed to identify correct conformations (i.e. 1JZZ, 1K8A, 1KD1, and 1K9M). All of these complexes have more than 27 rotable bonds, and an example was shown in Figure 3.3.2.6.

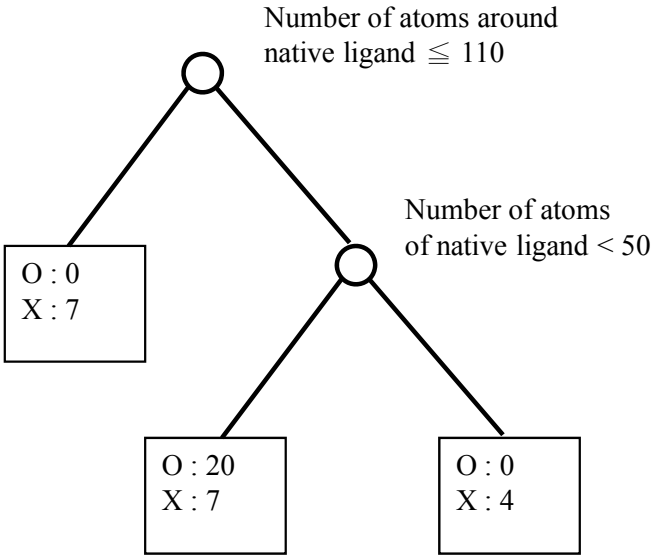

Figure 3.3.2.3. The decision tree model. O and X indicate the successful cases and unsuccessful cases, respectively. The fist rule for discriminating successful and unsuccessful cases is the number of atoms around native ligand, and the second rule is the number of atoms of native ligand.

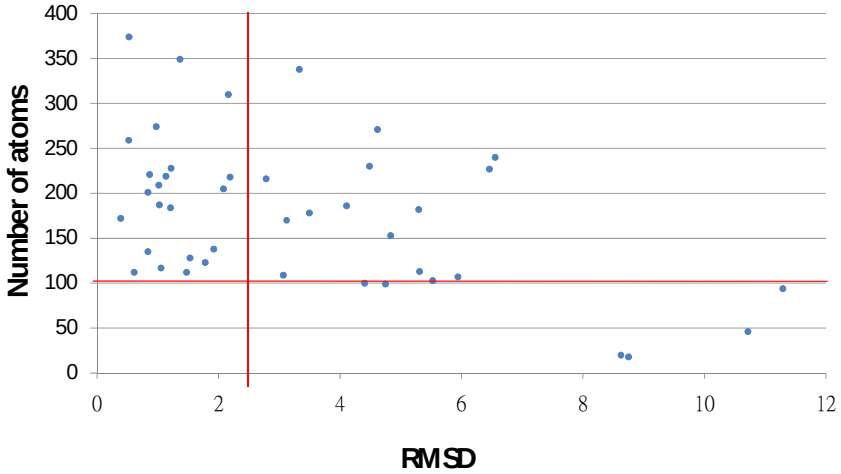

Figure 3.3.2.4. The distribution of the number of atoms around native ligand. The cases whose number of this feature is less than 110 failed in docking.

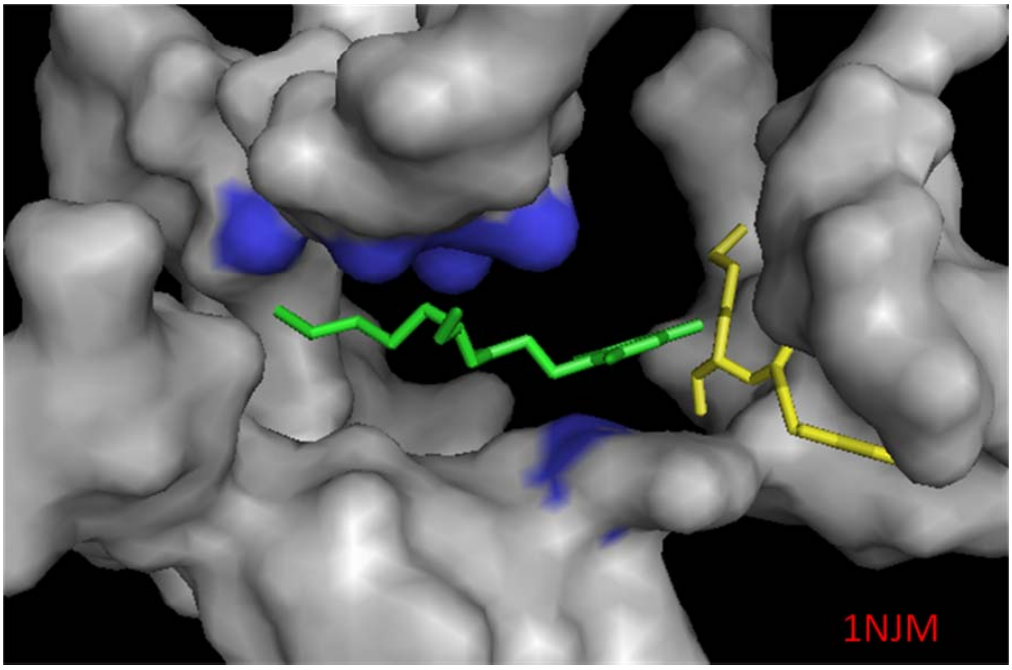

Figure 3.3.2.5. A case that has few interactions with RNA. The RMSD value between the native conformation (yellow) and the docked conformation (green) is 11.63Å. The native ligand has only 20 atoms within 4.5 Å and none of the 20 runs succeed in docking.

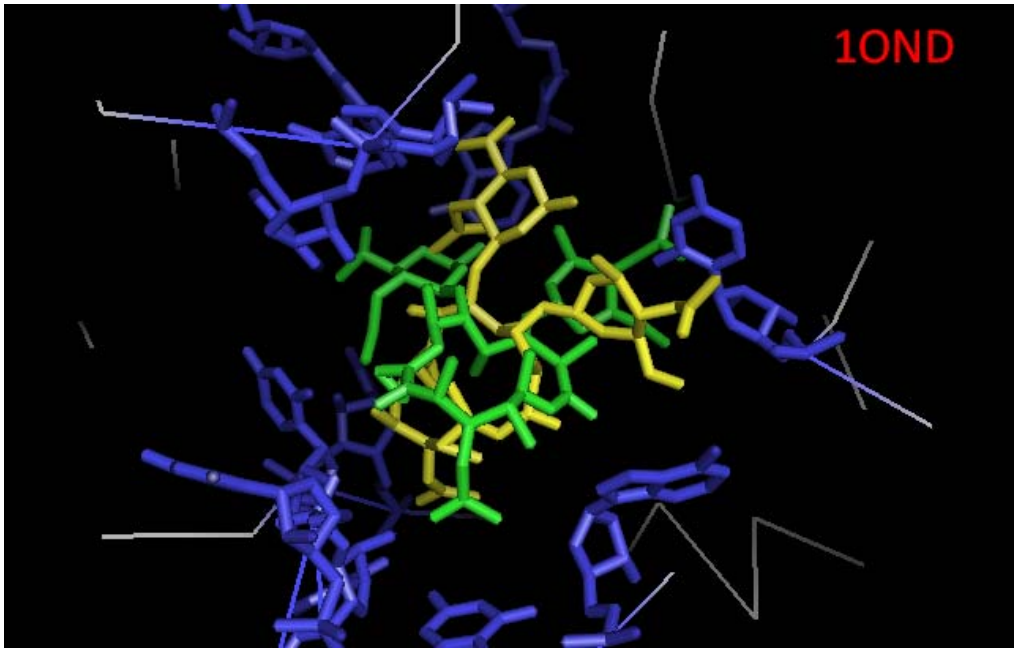

Figure 3.3.2.6. A example of the large ligands. The RMSD value between the native conformation (yellow) and the docked conformation (green) is 5.62Å. The native ligand has 57 heavy atoms and 26 rotable bonds, and none of the 20 runs succeed in docking.

3.3.4 Conclusion

In this study, we added the atom types that are specific to nucleotides in the scoring function of GEMDOCK, and the new scoring function is termed as “GemRNA”. GemRNA was applied to 38 RNA-ligand complexes which are proposed by Detering and Varani [168], and the successful rate of the 38 complexes is 53%. GemRNA shows good performances in RNA-ligand docking except for those ligands that bind RNA weakly and are large. In addition, GEMDOCK generated docked poses of ligands rapidly. On average, GEMDOCK took 305 seconds for a docking run on a Pentium 1.4 GHz personal computer with a single processor. These results demonstrate that the GemRNA is useful to predict conformations of ligands and fast. We believe that the GemRNA is useful for molecular recognition and virtual screening in large compound databases.

Chapter 4:
Self-evaluation of the project achievements

The goal of this project is to understand protein, RNA and DNA interaction based on sequence and structure information. In the past three years, we have studied protein/RNA structure prediction, clustering, protein-DNA interaction and protein-RNA docking. We have published five papers for the project and still work hard in studying interaction of protein, RNA and DNA. The five published papers are as below:

1. **Y. Hu**, "RNA Clustering and Secondary Structure Prediction", International Conference on Mathematics and Engineering Techniques in Medicine and Biological Science, 2005.
2. S. Ku and **Y. Hu**, "A Multistrategy Approach to Protein Structural Alphabet Design", Biocomp 2006.
3. K. Chen and **Y. Hu** "Bicluster Analysis of Genome-wide Gene Expression", IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology, 2006
4. C. Huang and **Y. Hu** "A Two-stage Approach to Finding Common Structure Elements in Unaligned RNA Sequences", Biocomp 2007
5. Y.-L. Chang, H.-K. Tsai, C.-Y. Kao, Y.-C. Chen, **Y.-J. Hu**, and J.-M. Yang*, "Evolutionary conservation of DNA-contact residues in DNA-binding domains," *BMC Bioinformatics*, vol. 9 (S6), pp. S3.1~S3.9, 2008

In the first year (2005), we have proposed a new adaptive method that conducts structure prediction and clustering simultaneously, since some current approaches can now identify common structure motifs from a set of RNAs, they typically assume the given set forms a single family, which is not necessarily correct. The performance of this study is demonstrated on several real RNA families, and showed very promising results. In the other hand, we have proposed a structure template-based method which used a novel scoring function to identify potential protein-DNA interactions, such method has the advantage of increasing crystal structures of protein-DNA complexes. Furthermore, the method also reveals the structure information of identified protein-DNA binding partners, we found that the feature, evolutionary conservation of DNA-contact residues, is helpful to identify DNA-binding domains. By using the scoring function based on such a feature, we successfully identified 66 DNA-binding domain families, also identify the different DNA-binding behaviors of proteins in the same SCOP family.

In the second year (2006), we demonstrated how the structural alphabet can be used with conventional 1D sequence alignment algorithms and presented its results. A comparative study of our alphabet with one of recently developed structural alphabets also showed a competitive result. Moreover, we proposed a new biclustering method based on the framework of market basket analysis in which a bicluster is described as a frequent itemset. As a feasibility test, we compared it with several standard clustering algorithms on a genome-wide yeast microarray dataset, and it showed very promising results. In the other

hand, we have proposed template-based alignment with a new scoring function which combined the evolutionary conservation and protein-DNA interacting scores of DNA-contact residues. We have showed that the combined scoring function is better to model the protein-DNA interactions than applying only one. Our method achieved high accuracy in identifying DNA-binding domains of 69 representative families and with the correlation 0.6 in predicting the binding free energy of the alanine scanning data.

In the third year (2007), unlike some methods that find consensus structures from a multiple sequence alignment if available or others that align sequences and structures simultaneously, we have developed an approach which separates consensus motif finding from sequence folding. After applying RNA folding algorithms to each sequence of given RNAs as a preprocess, we then combine structure decomposition and Gibbs sampling techniques to identify common structure motifs in unaligned RNA sequences. To demonstrate the performance, we tested it on several RNA families in Rfam. The experimental results show our new approach is competitive with other current prediction systems. Moreover, we have selected GEMDOCK as the program for docking ligand into RNA targets. We added the atom types that are specific to nucleotides in the scoring function of GEMDOCK. The new scoring function is termed as "GemRNA". We tested the performance of GemRNA on the public set (38 RNA-ligand complexes). The results show that GemRNA could model RNA-ligand binding reliably.

In summary, we believe that we have achieved fruitful results in this project. This project covers research areas from molecular interactions to regulatory networks of a biological system. We consider that the achievements in this project will be advantageous and valuable to researchers to study sequence-structure-function relationships and molecular interactions.

Reference

1. Hofacker I, Fontana, W., Stadler, P., Bonhoeffer, L., Tacker, M. And Schuster, P.: **Fast folding and comparison of RNA secondary structures**. *Monatshefte fur Chemie* 1994, **125**:167-188.
2. Zuker M: **On finding all suboptimal foldings of an RNA molecule**. *Science* 1989, **244**(4900):48-52.
3. Zuker M, Stiegler P: **Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information**. *Nucleic Acids Res* 1981, **9**(1):133-148.
4. Chiu DK, Kolodziejczak T: **Inferring consensus structure from nucleic acid sequences**. *Comput Appl Biosci* 1991, **7**(3):347-352.
5. Gutell RR: **Evolutionary characteristics of RNA: inferring higher-order structure from patterns of sequence variation**. *Curr Opin Struct Biol,* 1993, **3**:313-322.
6. Eddy SR, Durbin R: **RNA sequence analysis using covariance models**. *Nucleic Acids Res* 1994, **22**(11):2079-2088.
7. Knudsen B, Hein J: **Pfold: RNA secondary structure prediction using stochastic context-free grammars**. *Nucleic Acids Res* 2003, **31**(13):3423-3428.
8. Sakakibara Y, Brown M, Hughey R, Mian IS, Sjolander K, Underwood RC, Haussler D: **Stochastic context-free grammars for tRNA modeling**. *Nucleic Acids Res* 1994, **22**(23):5112-5120.
9. Gultyaev AP, van Batenburg FH, Pleij CW: **The computer simulation of RNA folding pathways using a genetic algorithm**. *J Mol Biol* 1995, **250**(1):37-51.
10. Hu YJ: **Prediction of consensus structural motifs in a family of coregulated RNA sequences**. *Nucleic Acids Res* 2002, **30**(17):3886-3893.
11. van Batenburg FH, Gultyaev AP, Pleij CW: **An APL-programmed genetic algorithm for the prediction of RNA secondary structure**. *J Theor Biol* 1995, **174**(3):269-280.
12. Hamada M, Tsuda K, Kudo T, Kin T, Asai K: **Mining frequent stem patterns from unaligned RNA sequences**. *Bioinformatics* 2006, **22**(20):2480-2487.
13. Ji Y, Xu X, Stormo GD: **A graph theoretical approach for predicting common RNA secondary structure motifs including pseudoknots in unaligned sequences**. *Bioinformatics* 2004, **20**(10):1591-1602.
14. Yao Z, Weinberg Z, Ruzzo WL: **CMfinder--a covariance model based RNA motif finding algorithm**. *Bioinformatics* 2006, **22**(4):445-452.
15. Siebert SaB, R.: **MARNA: A server for multiple alignment of RNAs**. *Proc of the German Conference on Bioinformatics,* 2003:35-40.
16. Hochsmann M, Toller T, Giegerich R, Kurtz S: **Local similarity in RNA secondary structures**. *Proc IEEE Comput Soc Bioinform Conf* 2003, **2**:159-168.
17. Juan V, Wilson C: **RNA secondary structure prediction based on free energy and**

phylogenetic analysis. *J Mol Biol* 1999, **289**(4):935-947.

18. Storz G: **An expanding universe of noncoding RNAs**. *Science* 2002, **296**(5571):1260-1263.
19. Lai EC: **RNA sensors and riboswitches: self-regulating messages**. *Curr Biol* 2003, **13**(7):R285-291.
20. Nudler E, Mironov AS: **The riboswitch control of bacterial metabolism**. *Trends Biochem Sci* 2004, **29**(1):11-17.
21. Hofacker IL, Priwitzer B, Stadler PF: **Prediction of locally stable RNA secondary structures for genome-wide surveys**. *Bioinformatics* 2004, **20**(2):186-190.
22. Furtig B, Richter C, Wohnert J, Schwalbe H: **NMR spectroscopy of RNA**. *Chembiochem* 2003, **4**(10):936-962.
23. Hu YJ: **GPRM: A genetic programming approach to finding common RNA secondary structure elements**. *Nucleic Acids Res* 2003, **31**(13):3446-3449.
24. Clark P, Boswell R: **Rule Induction with CN2: some recent improvements**. *In Proceedings of the Fifth European Working Session on Learning* 1991:p151-163.
25. Baker D, Sali A: **Protein structure prediction and structural genomics**. *Science* 2001, **294**(5540):93-96.
26. Bystroff C, Baker D: **Prediction of local structure in proteins using a library of sequencestructure motif**. *Journal of Molecular Biology* 1998, **281**:565-577.
27. Simons KT, I. R. CK, Fox BA, Bystroff C, Baker D: **Improved recognition of native-like protein structures using a combination of sequence-dependent and sequence-independent features of proteins**. *Proteins* 1999, **34**:82-95.
28. de Brevern AG, Etchebest C, Hazout S: **Bayesian probabilistic approach for predicting backbone structures in terms of protein blocks**. *Proteins* 2000, **41**(3):271-287.
29. Kohonen T: **Self-organizing Maps**. *Berlin/Heidelberg, Germany; Springer* 1995, **30**.
30. Vesanto J, Alhoniemi E: **Cluster of the selforganizing map**. *IEEE trans Neural Networks* 2000, **11**:586-600.
31. Tamayo P, Slonim D, Mesirov J, Zhu Q, Kitareewan S, Dmitrovsky E, Lander ES, Golub TR: **Interpreting patterns of gene expression with self-organizing maps: Methods and applications to hematopoietic differentiation**. *PNAS* 1999, **96**:2907-2912.
32. Iivarinen J, Kohonen T, Kangas J, Kaski S: **Visualizing the clusters on the self-organizing map**. *In Proc Conf Artificial Intelligence Research Finland* 1994:122-126.
33. Ermolaeva O, Rastogi M, Pruitt KD, Schuler GD, Bittner ML, Chen Y: **Data manegement and analysis for gene expression arrays**. *Nature Genetics* 1998, **20**:19-23.
34. Gaasterland T, Bekiranov S: **Making the most of microarray data**. *Nature Genetics* 2000, **24**:204-206.

35. Kim S, Dougherty ER, Bittner ML, Chen Y, Sivakumar K, Meltzer P, Trent JM: **General nonlinear framework for the analysis of gene interaction via multivariate expression arrays**. *J Biomed Opt* 2000, **5**(4):411-424.
36. Qian J, Dolled-Filhart M, Lin J, Yu H, Gerstein M: **Beyond synexpression relationships: local clustering of time-shifted and inverted gene expression profiles identifies new, biologically relevant interactions**. *J Mol Biol* 2001, **314**(5):1053-1066.
37. Eisen MB, Spellman PT, Brown PO, Botstein D: **Cluster analysis and display of genome-wide expression patterns**. *Proc Natl Acad Sci U S A* 1998, **95**(25):14863-14868.
38. Hartigan JA, Wong MA: **A k-means clustering algorithm**. *Applied Statistics* 1979, **28**:100-108.
39. Ben-Dor A, Chor B, Karp R, Yakhini Z: **Discovering local structure in gene expression data: the order-preserving submatrix problem**. *J Comput Biol* 2003, **10**(3-4):373-384.
40. Cheng Y, Church GM: **Biclustering of expression data**. *Proc Int Conf Intell Syst Mol Biol* 2000, **8**:93-103.
41. Hartigan JA: **Direct clustering of a data matrix**. *J Am Statistical Assoc* 1972, **67**:123-129.
42. Mirkin B: **Nonconcex optimization and its applications**. *Math Classification and Clustering* 1996.
43. Madeira SC, Oliveira AL: **Biclustering algorithms for biological data analysis: a survey**. *IEEE/ACM Trans Comput Biol Bioinform* 2004, **1**(1):24-45.
44. Tanay A, Sharan R, Shamir R: **Discovering statistically significant biclusters in gene expression data**. *Bioinformatics* 2002, **18 Suppl 1**:S136-144.
45. Lazzeroni L, Owen A: **Plaid models for gene expression data**. *techical report, Stanford Univ* 2000.
46. Han J, Pei J: **Mining frequent patterns by pattern-growth: methodology and implications**. *SIGKDD Explorations 2* 2000:14-20.
47. Prelic A, Bleuler S, Zimmermann P, Wille A, Buhlmann P, Gruissem W, Hennig L, Thiele L, Zitzler E: **A systematic comparison and evaluation of biclustering methods for gene expression data**. *Bioinformatics* 2006, **22**(9):1122-1129.
48. Jolliffe IT: **Principle component analysis**. *J Educational Psychology* 1986, **24**:417-441.
49. Ihmels J, Bergmann S, Barkai N: **Defining transcription modules using large-scale gene expression data**. *Bioinformatics* 2004, **20**:1993-2003.
50. Murali TM, Kasif S: **Extracting conserved gene expression motifs from gene expression data**. *Pac Symp Biocomput* 2003:77-88.
51. Quackenbush J: **Microarray data normalization and transformation**. *Nat Genet* 2002, **32 Suppl**:496-501.

52. Qiu X, Brooks AI, Klebanov L, Yakovlev N: **The effects of normalization on the correlation structure of microarray data**. *BMC Bioinformatics* 2005, **6**:120.
53. Spellman PT, Sherlock G, Zhang MQ, Iyer VR, Anders K, Eisen MB, Brown PO, Botstein D, Futcher B: **Comprehensive identification of cell cycle-regulated genes of the yeast Saccharomyces cerevisiae by microarray hybridization**. *Mol Biol Cell* 1998, **9**(12):3273-3297.
54. Becquet C, Blachon S, Jeudy B, Boulicaut JF, Gandrillon O: **Strong-association-rule mining for large-scale gene-expression data analysis: a case study on human SAGE data**. *Genome Biol* 2002, **3**(12):RESEARCH0067.
55. Creighton C, Hanash S: **Mining gene expression databases for association rules**. *Bioinformatics* 2003, **19**(1):79-86.
56. Agrawal R, Srikant R: **Fast algorithms for mining association rules in large databases**. *Proc International Conf Very Large Data Bases* 1994:478-499.
57. Gardner PP, Giegerich R: **A comprehensive comparison of comparative RNA structure prediction approaches**. *BMC Bioinformatics* 2004, **5**:140.
58. Thompson J, Higgins D, Gibson T: **CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice**. *Nucleic Acids Research* 1994, **22**:4673-4680.
59. Gutell RR: **Evolutionary characteristics of RNA: inferring higher-order structure from patterns of sequence variation**. *Curr Opin Struct Biol* 1993, **3**:313-322.
60. Hofacker IL, Fekete M, Stadler PF: **Secondary structure prediction for aligned RNA sequences**. *J Mol Biol* 2002, **319**(5):1059-1066.
61. Gorodkin J, Heyer L, Stormo G: **Finding the most significant common sequence and structure motifs in a set of RNA sequences**. *Nucleic Acids Research* 1997, **25**:3724-3732.
62. Sankoff D: **Simultaneous solution of the RNA folding, alignment and protosequence problems**. *SIAM J Applied Math* 1985, **45**:810-825.
63. Touzet H, Perriquet O: **CARNAC: folding families of related RNAs**. *Nucleic Acids Res* 2004, **32**(Web Server issue):W142-145.
64. Giegerich R, Voss B, Rehmsmeier M: **Abstract shapes of RNA**. *Nucleic Acids Res* 2004, **32**(16):4843-4851.
65. Siebert S, Backofen R: **MARNA: A server for multiple alignment of RNAs**. *Proc of the German Conference on Bioinformatics* 2003:35-40.
66. Needleman SB, Wunsch CD: **A general method applicable to the search for similarities in the amino acid sequence of two proteins**. *J Mol Biol* 1970, **48**(3):443-453.
67. Liu J, Wang JT, Hu J, Tian B: **A method for aligning RNA secondary structures and its application to RNA motif detection**. *BMC Bioinformatics* 2005, **6**:89.
68. Matthews BW: **Comparison of the predicted and observed secondary structure of**

T4 phage lysozyme. *Biochim Biophys Acta* 1975, **405**(2):442-451.

69. Gorodkin J, Stricklin SL, Stormo GD: **Discovering common stem-loop motifs in unaligned RNA sequences**. *Nucleic Acids Res* 2001, **29**(10):2135-2144.
70. Hu YJ: **The NCTU BioInfo Archive of biological data sets for bioinformatics research and experimentation**. *Bioinformatics* 2002, **18**(8):1145-1146.
71. van Batenburg FH, Gultyaev AP, Pleij CW: **PseudoBase: structural information on RNA pseudoknots**. *Nucleic Acids Res* 2001, **29**(1):194-195.
72. Brevern AGd, Valadie H, Hazout SA, Etchebest C: **Extension of a local backbone description using a structural alphabet: A new approach to the sequence-structure relationship**. *Protein Science* 2002, **11**:2871-2886.
73. Cho RJ, Campbell MJ, Winzeler EA, Steinmetz L, Conway A, Wodicka L, Wolfsberg TG, Gabrielian AE, Landsman D, Lockhart DJ *et al*: **A genome-wide transcriptional analysis of the mitotic cell cycle**. *Mol Cell* 1998, **2**(1):65-73.
74. Chu S, DeRisi J, Eisen M, Mulholland J, Botstein D, Brown PO, Herskowitz I: **The transcriptional program of sporulation in budding yeast**. *Science* 1998, **282**(5389):699-705.
75. DeRisi JL, Iyer VR, Brown PO: **Exploring the metabolic and genetic control of gene expression on a genomic scale**. *Science* 1997, **278**(5338):680-686.
76. Holstege FC, Jennings EG, Wyrick JJ, Lee TI, Hengartner CJ, Green MR, Golub TR, Lander ES, Young RA: **Dissecting the regulatory circuitry of a eukaryotic genome**. *Cell* 1998, **95**(5):717-728.
77. Jelinsky SA, Samson LD: **Global response of Saccharomyces cerevisiae to an alkylating agent**. *Proc Natl Acad Sci U S A* 1999, **96**(4):1486-1491.
78. Lashkari DA, DeRisi JL, McCusker JH, Namath AF, Gentile C, Hwang SY, Brown PO, Davis RW: **Yeast microarray for genome wide parallel genetic and gene expression analysis**. *PNAS* 1997, **94**:13057-13062.
79. Hughes TR, Marton MJ, Jones AR, Roberts CJ, Stoughton R, Armour CD, Bennett HA, Coffey E, Dai H, He YD *et al*: **Functional discovery via a compendium of expression profiles**. *Cell* 2000, **102**(1):109-126.
80. Dembele D, Kastner P: **Fuzzy C-means method for clustering microarray data**. *Bioinformatics* 2003, **19**(8):973-980.
81. Ihmels JH, Bergmann S: **Challenges and prospects in the analysis of large-scale gene expression data**. *Brief Bioinform* 2004, **5**(4):313-327.
82. Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, Davis AP, Dolinski K, Dwight SS, Eppig JT *et al*: **Gene ontology: tool for the unification of biology. The Gene Ontology Consortium**. *Nat Genet* 2000, **25**(1):25-29.
83. Berriz GF, King OD, Bryant B, Sander C, Roth FP: **Characterizing gene sets with FuncAssociate**. *Bioinformatics* 2003, **19**(18):2502-2504.
84. Steffen P, Voss B, Rehmsmeier M, Reeder J, Giegerich R: **RNAshapes: an integrated RNA analysis package based on abstract shapes**. *Bioinformatics* 2006,

22(4):500-503.

85. Michael Gromiha M, Siebers JG, Selvaraj S, Kono H, Sarai A: **Intermolecular and intramolecular readout mechanisms in protein-DNA recognition**. *J Mol Biol* 2004, **337**(2):285-294.
86. Harrison SC: **A structural taxonomy of DNA-binding domains**. *Nature* 1991, **353**(6346):715-719.
87. Luscombe NM, Austin SE, Berman HM, Thornton JM: **An overview of the structures of protein-DNA complexes**. *Genome Biol* 2000, **1**(1):REVIEWS001.
88. Vinson CR, Sigler PB, McKnight SL: **Scissors-grip model for DNA recognition by a family of leucine zipper proteins**. *Science* 1989, **246**(4932):911-916.
89. Johnson PF, McKnight SL: **Eukaryotic transcriptional regulatory proteins**. *Annu Rev Biochem* 1989, **58**:799-839.
90. Berman HM, Olson WK, Beveridge DL, Westbrook J, Gelbin A, Demeny T, Hsieh SH, Srinivasan AR, Schneider B: **The nucleic acid database. A comprehensive relational database of three-dimensional structures of nucleic acids**. *Biophys J* 1992, **63**(3):751-759.
91. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, Shindyalov IN, Bourne PE: **The Protein Data Bank**. *Nucleic Acids Res* 2000, **28**(1):235-242.
92. Henikoff S, Henikoff JG: **Amino acid substitution matrices from protein blocks**. *Proc Natl Acad Sci U S A* 1992, **89**(22):10915-10919.
93. Chang YL, Tsai HK, Kao CY, Chen YC, Hu YJ, Yang JM: **Evolutionary conservation of DNA-contact residues in DNA-binding domains**. *BMC Bioinformatics* 2008, **9 Suppl 6**:S3.
94. Ahmad S, Gromiha MM, Sarai A: **Analysis and prediction of DNA-binding proteins and their binding residues based on composition, sequence and structural information**. *Bioinformatics* 2004, **20**(4):477-486.
95. Ahmad S, Sarai A: **Moment-based prediction of DNA-binding proteins**. *J Mol Biol* 2004, **341**(1):65-71.
96. Bhardwaj N, Langlois RE, Zhao G, Lu H: **Kernel-based machine learning protocol for predicting DNA-binding proteins**. *Nucleic Acids Res* 2005, **33**(20):6486-6493.
97. Bhardwaj N, Lu H: **Residue-level prediction of DNA-binding sites and its application on DNA-binding protein predictions**. *FEBS Lett* 2007, **581**(5):1058-1066.
98. Szilagyi A, Skolnick J: **Efficient prediction of nucleic acid binding function from low-resolution protein structures**. *J Mol Biol* 2006, **358**(3):922-933.
99. Tsuchiya Y, Kinoshita K, Nakamura H: **Structure-based prediction of DNA-binding sites on proteins using the empirical preference of electrostatic potential and the shape of molecular surfaces**. *Proteins* 2004, **55**(4):885-894.
100. Yu X, Cao J, Cai Y, Shi T, Li Y: **Predicting rRNA-, RNA-, and DNA-binding proteins from primary structure with support vector machines**. *J Theor Biol* 2006,

240(2):175-184.

101. Ahmad S, Sarai A: **PSSM-based prediction of DNA binding sites in proteins**. *BMC Bioinformatics* 2005, **6**:33.
102. Kuznetsov IB, Gou Z, Li R, Hwang S: **Using evolutionary and structural information to predict DNA-binding sites on DNA-binding proteins**. *Proteins* 2006, **64**(1):19-27.
103. Tjong H, Zhou HX: **DISPLAR: an accurate method for predicting DNA-binding sites on protein surfaces**. *Nucleic Acids Res* 2007, **35**(5):1465-1477.
104. Luscombe NM, Thornton JM: **Protein-DNA interactions: amino acid conservation and the effects of mutations on binding specificity**. *J Mol Biol* 2002, **320**(5):991-1009.
105. Stawiski EW, Gregoret LM, Mandel-Gutfreund Y: **Annotating nucleic acid-binding function based on protein structure**. *Journal of Molecular Biology* 2003, **326**(4):1065-1079.
106. Klug SJ, Famulok M: **All You Wanted to Know About Selex**. *Molecular Biology Reports* 1994, **20**(2):97-107.
107. Riechmann L, Winter G: **Novel folded protein domains generated by combinatorial shuffling of polypeptide segments**. *Proceedings of the National Academy of Sciences of the United States of America* 2000, **97**(18):10068-10073.
108. Winter G, Milstein C: **Man-Made Antibodies**. *Nature* 1991, **349**(6307):293-299.
109. Choo Y, Klug A: **Selection of DNA-Binding Sites for Zinc Fingers Using Rationally Randomized DNA Reveals Coded Interactions**. *Proceedings of the National Academy of Sciences of the United States of America* 1994, **91**(23):11168-11172.
110. Choo Y, Klug A: **Toward a Code for the Interactions of Zinc Fingers with DNA - Selection of Randomized Fingers Displayed on Phage (Vol 91, Pg 11163, 1994)**. *P Natl Acad Sci USA* 1995, **92**(2):646-646.
111. Iyer VR, Horak CE, Scafe CS, Botstein D, Snyder M, Brown PO: **Genomic binding sites of the yeast cell-cycle transcription factors SBF and MBF**. *Nature* 2001, **409**(6819):533-538.
112. Lieb JD, Liu XL, Botstein D, Brown PO: **Promoter-specific binding of Rap1 revealed by genome-wide maps of protein-DNA association**. *Nature Genetics* 2001, **28**(4):327-334.
113. Ren B, Robert F, Wyrick JJ, Aparicio O, Jennings EG, Simon I, Zeitlinger J, Schreiber J, Hannett N, Kanin E *et al*: **Genome-wide location and function of DNA binding proteins**. *Science* 2000, **290**(5500):2306-+.
114. N.C. Seeman JMR, and A. Rich: **Sequence specific recognition of double helical nucleic acids by proteins.** *Proc Natl Acad Sci U S A* 1976, **73**:804-808.
115. Matthews BW: **Protein-DNA Interaction - No Code for Recognition**. *Nature* 1988, **335**(6188):294-295.

116. Mandel-Gutfreund Y, Margalit H: **Quantitative parameters for amino acid-base interaction: implications for prediction of protein-DNA binding sites**. *Nucleic Acids Research* 1998, **26**(10):2306-2312.
117. Kono H, Sarai A: **Structure-based prediction of DNA target sites by regulatory proteins**. *Proteins-Structure Function and Genetics* 1999, **35**(1):114-131.
118. Liu ZJ, Mao FL, Guo JT, Yan B, Wang P, Qu YX, Xu Y: **Quantitative evaluation of protein-DNA interactions using an optimized knowledge-based potential**. *Nucleic Acids Research* 2005, **33**(2):546-558.
119. Havranek JJ, Duarte CM, Baker D: **A simple physical model for the prediction and design of protein-DNA interactions**. *Journal of Molecular Biology* 2004, **344**(1):59-70.
120. Morozov AV, Havranek JJ, Baker D, Siggia ED: **Protein-DNA binding specificity predictions with structural models**. *Nucleic Acids Research* 2005, **33**(18):5781-5798.
121. Murzin AG, Brenner SE, Hubbard T, Chothia C: **Scop - a Structural Classification of Proteins Database for the Investigation of Sequences and Structures**. *Journal of Molecular Biology* 1995, **247**(4):536-540.
122. Luscombe NM, Laskowski RA, Thornton JM: **Amino acid-base interactions: a three-dimensional analysis of protein-DNA interactions at an atomic level**. *Nucleic Acids Research* 2001, **29**(13):2860-2874.
123. Holm L, Sander C: **Protein-Structure Comparison by Alignment of Distance Matrices**. *Journal of Molecular Biology* 1993, **233**(1):123-138.
124. Pearson WR: **Effective protein sequence comparison**. *Computer Methods for Macromolecular Sequence Analysis* 1996, **266**:227-258.
125. Pearson WR: **Flexible sequence similarity searching with the FASTA3 program package**. *Methods Mol Biol* 2000, **132**:185-219.
126. Pearson WR, Lipman DJ: **Improved Tools for Biological Sequence Comparison**. *Proceedings of the National Academy of Sciences of the United States of America* 1988, **85**(8):2444-2448.
127. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ: **Basic Local Alignment Search Tool**. *Journal of Molecular Biology* 1990, **215**(3):403-410.
128. Karplus K, Barrett C, Hughey R: **Hidden Markov models for detecting remote protein homologies**. *Bioinformatics* 1998, **14**(10):846-856.
129. Skolnick J, Fetrow JS: **From genes to protein structure and function: novel applications of computational approaches in the genomic era**. *Trends in Biotechnology* 2000, **18**(1):34-39.
130. Smith TF: **The art of matchmaking: sequence alignment methods and their structural implications**. *Structure* 1999, **7**(1):R7-R12.
131. Smith TF, Waterman MS: **Identification of Common Molecular Subsequences**. *Journal of Molecular Biology* 1981, **147**(1):195-197.

132. Shindyalov IN, Bourne PE: **Protein structure alignment by incremental combinatorial extension (CE) of the optimal path**. *Protein Engineering* 1998, **11**(9):739-747.
133. Hobohm U, Sander C: **Enlarged Representative Set of Protein Structures**. *Protein Sci* 1994, **3**(3):522-524.
134. Stawiski EW, Gregoret LM, Mandel-Gutfreund Y: **Annotating nucleic acid-binding function based on protein structure**. *J Mol Biol* 2003, **326**(4):1065-1079.
135. Clackson T, Wells JA: **A hot spot of binding energy in a hormone-receptor interface**. *Science* 1995, **267**(5196):383-386.
136. Cunningham BC, Wells JA: **Rational design of receptor-specific variants of human growth hormone**. *Proc Natl Acad Sci U S A* 1991, **88**(8):3407-3411.
137. Thorn KS, Bogan AA: **ASEdb: a database of alanine mutations and their effects on the free energy of binding in protein interactions**. *Bioinformatics* 2001, **17**(3):284-285.
138. Passner JM, Ryoo HD, Shen LY, Mann RS, Aggarwal AK: **Structure of a DNA-bound Ultrabithorax-Extradenticle homeodomain complex**. *Nature* 1999, **397**(6721):714-719.
139. LaRronde-LeBlanc NA, Wolberger C: **Structure of HoxA9 and Pbx1 bound to DNA: Hox hexapeptide and DNA recognition anterior to posterior**. *Gene Dev* 2003, **17**(16):2060-2072.
140. Dutnall RN, Tafrov ST, Sternglanz R, Ramakrishnan V: **Structure of the histone acetyltransferase Hat1: A paradigm for the GCN5-related N-acetyltransferase superfamily**. *Cell* 1998, **94**(4):427-438.
141. Clore GM, Williams D: **Molecular basis for synergistic transcriptional activation by Oct1 and Sox2 revealed from the solution structure of the 42 kDa Oct1-Sox2-Hoxb1-DNA ternary transcription factor complex**. *Faseb J* 2004, **18**(8):C81-C81.
142. Bhardwaj N, Lu H: **Residue-level prediction of DNA-binding sites and its application on DNA-binding protein predictions**. *Febs Lett* 2007, **581**(5):1058-1066.
143. Ahmad S, Sarai A: **Moment-based prediction of DNA-binding proteins**. *J Mol Biol* 2004, **341**(1):65-71.
144. Johnson PF, Mcknight SL: **Eukaryotic Transcriptional Regulatory Proteins**. *Annu Rev Biochem* 1989, **58**:799-839.
145. Konagurthu AS, Whisstock JC, Stuckey PJ, Lesk AM: **MUSTANG: A multiple structural alignment algorithm**. *Proteins* 2006, **64**(3):559-574.
146. Luscombe NM, Thornton JM: **Protein-DNA interactions: Amino acid conservation and the effects of mutations on binding specificity**. *Journal of Molecular Biology* 2002, **320**(5):991-1009.
147. Tsai MJ, O'Malley BW: **Molecular mechanisms of action of steroid/thyroid**

receptor superfamily members. *Annu Rev Biochem* 1994, **63**:451-486.

148. Brent GA: **The molecular basis of thyroid hormone action**. *N Engl J Med* 1994, **331**(13):847-853.
149. Rastinejad F, Perlmann T, Evans RM, Sigler PB: **Structural determinants of nuclear receptor assembly on DNA direct repeats**. *Nature* 1995, **375**(6528):203-211.
150. Crooks GE, Hon G, Chandonia JM, Brenner SE: **WebLogo: a sequence logo generator**. *Genome Res* 2004, **14**(6):1188-1190.
151. Hermann T: **Drugs targeting the ribosome**. *Curr Opin Struct Biol* 2005, **15**(3):355-366.
152. Sucheck SJ, Wong CH: **RNA as a target for small molecules**. *Current Opinion in Chemical Biology* 2000, **4**:678–686.
153. Gohlke H, Hendlich M, Klebe G: **Knowledge-based scoring function to predict protein-ligand interactions**. *Journal of Molecular Biology* 2000, **295**:337-356.
154. Verdonk L, Cole JC, Watson P, Gillet V, Willett P: **SuperStar: improved knowledge-based interaction fields for protein binding sites**. *Journal of Molecular Biology* 2001, **307**:841-859.
155. Gehlhaar DK, Verkhivker GM, Rejto P, Sherman CJ, Fogel DB, Fogel LJ, Freer ST: **Molecular recognition of the inhibitor AG-1343 by HIV-1 protease: conformationally flexible docking by evolutionary programming**. *Chemistry and Biology* 1995, **2**:317-324.
156. Verkhivker GM, Bouzida D, Gehlhaar DK, Rejto PA, Arthurs S, Colson AB, Freer ST, Larson V, Luty BA, Marrone T *et al*: **Deciphering common failures in molecular docking of ligand-protein complexes**. *Journal of Computer-Aided Molecular Design* 2000, **14**:531-551.
157. Taylor JS, Burnett RM: **DARWIN: A program for docking flexible molecules**. *Proteins: Structure, Function, and Genetics* 2000, **41**:173-191.
158. Weiner SJ, Kollman PA, Case DA, Singh UC, Ghio C, Alagona G, Profeta SJ, Weiner P: **A new force field for molecular mechanical simulation of nucleic acids and proteins**. *Journal of the American Chemical Society* 1984, **106**:765-784.
159. Shoichet BK, LeachA R, Kuntz ID: **Ligand solvation in molecular docking**. *Proteins: Structure, Function, and Genetics* 1999, **34**:4-16.
160. Yang J-M, Chen C-C: **GEMDOCK: a generic evolutionary method for molecular docking**. *Proteins: Structure, Function, and Bioinformatics* 2004, **55**:288-304.
161. Yang J-M, Shen T-W: **A pharmacophore-based evolutionary approach for screening selective estrogen receptor modulators**. *Proteins: Structure, Function, and Bioinformatics* 2005, **59**:205-220.
162. Yang J-M, Chen Y-F, Shen T-W, Kristal BS, Hsu DF: **Consensus scoring sriteria for Improving enrichment in virtual screening**. *Journal of Chemical Information and Modeling* 2005(45):1134-1146.

163. Yang JM, Chen YF, Tu YY, Yen KR, Yang YL: **Combinatorial computational approaches to identify tetracycline derivatives as flavivirus inhibitors**. *PLoS ONE* 2007:e428.
164. Lin ES, Yang JM, Yang YS: **Modeling the binding and inhibition mechanism of nucleotide and sulfotransferase using molecular docking**. *Journal of the Chinese Chemical Society* 2003, **50**:655-663.
165. Knegtel RMA, Antoon J, Rullmann C, Boelens R, Kaptein R: **MONTY: a Monte Carlo approach to protein-DNA recognition**. *Journal of Molecular Biology* 1994, **235**:318-324.
166. Storn R, Price KV: **Differential evolution: a simple and efficient heuristic for global optimization over continuous spaces**. *Journal of Global Optimization* 1997, **11**:341-369.
167. Back T: **Evolutionary Algorithms in Theory and Practice**. In. New York, USA: Oxford University Press; 1996.
168. Detering C, Varani G: **Validation of automated docking programs for docking and database screening against RNA drug targets**. *Journal of Medicinal Chemistry* 2004, **47**:4188-4201.
169. Jones G, Willett P, Glen RC, Leach AR, Taylor R: **Development and validation of a genetic algorithm for flexible docking**. *Journal of Molecular Biology* 1997, **267**:727-748.
170. Kramer B, Rarey M, Lengauer T: **Evaluation of the FlexX incremental construction algorithm for protein-ligand docking**. *Proteins: Structure, Function, and Genetics* 1999, **37**:228-241.
171. Quinlan JR: **C4.5: Programs for Machine Learning**. 1993.

行政院國家科學委員會補助國內專家學者出席國際學術會議報告

97 年 9 月 15 日

附件一

報告人姓名	胡 毓 志	服務機構及職稱	交通大學資訊工程系 副教授
會議時間 會議地點	07/14/2008-07/17/2008 Las Vegas, U.S.A.	本會核定補助文號	NSC 96-2221-E-009-042-
會議名稱	（中文）Biocomp 生物資訊暨計算生物學國際研討會 （英文）Biocomp 2008		
發表論文題目	1.（中文）利用蛋白質結構字元表述蛋白質區間特性 （英文）Using Protein Structural Alphabet to Characterize Local Structure Features		

	一、參加會議經過 於 07/13 辦理註冊報到，隔日隨即參加開幕演說，於 07/14-07/17 期間，參加與會學者之論文發表，並與多位國外學者討論相關研究議題。會議中不乏中國大陸籍學者之論文，對於我國內生物資訊的發展，應可產生良性刺激，提供非常多的助益與新的發展方向。 二、與會心得 根據議程中部分美國研究學者所述，由於經濟壓力上升，美國 NIH 已將研究主軸放在 translational research，希望藉由在實驗室的研究成果實際應用於人類醫學。本次參加人數及國家眾多，其研究領域更包括計算機科學、醫學、生物學等之應用，藉由討論及論文發表，獲得寶貴經驗，對於未來研究提供了新的方向。其中更結識他國友人，經由研討，可明白其他國家的發展經驗。從這次與會學習的經驗，我們可以得知國外研究之重點，作為我國在生物科技的發展依據。 三、考察參觀活動(無是項活動者省略) 無 四、建議 生物科技是目前國內新興研究發展之重要產業，懇請國科會及相關單位，能多支持與獎勵國內學者多參與此類國際研討會，除了增加我國在國際相關領域的能見度，同時，提供相互學習之機會。此外，建議由國科會主導，召集國內各大學與民間企業支援，以召開國際性生物資訊與相關科技研討會，邀請國內外學者共同參與，這是直接提昇我國在生技發展地位的最有效做法。 五、攜回資料名稱及內容 The Proceedings of Biocomp2008

Using Protein Structural Alphabet to Characterize Local Structure Features

Shih-Yen Ku[12] **and Yuh-Jyh Hu**[13]
[1]Department of Computer Science, National Chiao Tung University, Hsinchu, Taiwan
[2]MIB Program at Institute of Statistical Science, Academia Sinica, Taipei, Taipei
[3]Institute of Biomedical Engineering, National Chiao Tung University, Hsinchu, Taiwan

Abstract - **As the number of available 3D protein structures increases rapidly, a wider variety of studies can be conducted more efficiently, among which is the design of protein structural alphabet. With the structural alphabet, not only can we describe the global folding structure of a protein as a 1D sequence, but we can also characterize local structures in proteins. Previously, we applied a combinatorial approach to protein structural alphabet design. In our previous work, we verified the usefulness of our structural alphabet by demonstrating the competitive accuracy in protein alignment, compared with alphabets. Here we took a further step by applying motif finding tools to our alphabet with the aim to characterize protein structure local features. Two structure domains, TIM and EGF, were used to evaluate the performance of our structural alphabet. Our method successfully recovered their sub-domains as common motifs in our structural alphabet.**

Keywords: protein structure, structural alphabet, motifs

Introduction

As all proteins have a certain degree of structural similarities to other proteins, and they probably share a common ancestor in evolution. Based on evolutionary relationships and the principles governing the 3D structures, a protein structure hierarchy, SCOP, was constructed mainly by visual inspection with the assistance of various automatic tools to compare protein structures. The original aim of SCOP was to serve as a tool for understanding protein evolution through the relationships between sequences and structures [1].

The conservation in local active sites may reflect biological meanings, and their structural patterns can be used to predict protein functions [2], e.g., the binding sites for metal-binding proteins [3]. The conserved local structural features can be identified in various ways and described in different representations. For example, some have attempted to investigate the relationships between local sequences and structures by identifying common structural motifs first, then characterizing amino acid preferences [4-6]. Others instead have adopted the inverse approach by examining structural correlates from recurring sequence patterns found to obtain sequence-structure motifs [7,8].

Unlike those works above on correlations between protein local structures and sequence patterns, we first convert protein 3D structures into 1D structural alphabet letters, and then identify and represent conserved local features as 1D structural alphabet sequence motifs. Besides, our goal is to mine the protein families for conserved local characteristics rather than to predict 3D structures of novel proteins as those studies mentioned above. There are several advantages of 1D structural alphabet over 3D co-ordinates representations. First, 1D representation of protein structures is more efficient in comparison and more economical in storage. Second, many previously designed and widely used 1D sequence alignment tools can be directly applied to protein structures as well as sequences. Third, conserved protein local structural features can be described as 1D sequence motifs and be identified by various well-developed sequence motif-finding tools. Four, this type of 1D-based approaches can serve as a pre-processor to filter out remotely related or irrelevant proteins before we apply other more accurate but more computationally intensive structure analysis tool.

Previous analysis of protein structures has shown the importance of repetitive secondary structures, in particular, α-helix and β-sheet. Together with variable coils, they constituted a basic standard 3-letter structural alphabet. In spite

of the increase in predictive accuracy, the approximation of 3D structures with only a 3-letter alphabet is apparently too crude for the more refined 3D reconstruction [9-13]. Various more complex structural alphabets have been developed by taking into account the heterogeneity of backbone protein structures through sets of small protein fragments frequently observed in different protein structure databases [14-21]. Unlike most other works, we developed a multi-strategy method for structural alphabet design, which combined self-organizing maps, minimum spanning tree algorithm and k-means algorithm [22]. The performance of our alphabet was demonstrated by the competitive accuracy in all-alpha protein search within SCOP using the standard 1D sequence alignment tool, FASTA [23].

In this paper, we introduced an improved version of our alphabet design pipeline, to which we added a substitution matrix self-trainer. The substitution matrix used in aligning proteins represented by structural alphabets affects the accuracy of alignment. In our earlier work, we applied the identity matrix in the alignment [22]. Though the preliminary results successfully demonstrated the feasibility of our alphabet, yet a more appropriate matrix will further improve its applicability. The substitution matrix is a crucial factor in the successful application of 1D sequence alignment tools to search for similar 3D structures. We thus developed an automatic matrix training framework that can generate appropriate substitution matrices for new alphabets when applied in standard 1D sequence alignment methods, e.g. FASTA. Based on the alphabet we constructed, we can transform proteins into 1D structural alphabet representations. To identify protein local structure features, we applied the motif-finding tool MEME [24] to detect the common motifs. We tested two protein families in SCOP, TIM and EGF. The results showed our method successfully recovered their structure domains.

Materials and Methods

The simplest substitution matrix to use is the identity matrix, but it ignores possible acceptable alphabet letter substitutions, which significantly limits its applicability. Some authors applied HMM approach to define the matrix [25], while others adopted a similar approach in the development of BLOSUM matrices [26,27]. Most of these approaches to constructing substitution matrices required the alignments of known proteins [27-29]. As the alignments may be unavailable or even questionable, we took a self-training strategy to build a substitution matrix for our new structural alphabet. This training framework is a flexible and modular design, and it does not rely on any pre-alignment of protein sequences or structures. This matrix training procedure can be applied regardless of how the alphabet is derived. Different training data or alignment tools available can be incorporated in this framework to generate appropriate matrices under various circumstances.

There are three components in the matrix training framework, an alignment tool with a substitution matrix, training data, and a matrix trainer. We used FASTA as the alignment tool, and the non-redundant proteins in SCOP1.69 with sequence similarity less than 40%, excluding the families of size smaller than 5 proteins, as the training dataset. We started by using the identity matrix as the initial substitution matrix where the score is 1 for a match, 0 for a mismatch. Each protein in the training dataset was iteratively used as a query for FASTA to search the rest of the dataset for similar proteins. If a protein returned by FASTA belonged to the same family as the query, we considered the case as a positive hit; otherwise, a negative hit. Those proteins not returned by FASTA but in the same family as the query were considered as misses. For all positive hits and misses, we gathered their alignments with the query produced by FASTA. Based on the alignments, we computed the log-odd ratios defined in the same way as in the BLOSUM matrices [28] to build the *positive matrix*. Similarly, with the alignments of negative hits, we constructed the *negative matrix*. The matrix trainer updated the current substitution matrix $S^{(t)}$ to $S^{(t+1)}$ as the following.

$$S^{(t+1)} = S^{(t)} + M$$

$$M = [W_p \cdot (P - S^{(t)}) - W_n \cdot (N - S^{(t)})] \cdot \tau$$

$$W_p = (|positive\_hits| + |misses|) / |training\_data|$$

$$W_n = |negative\_hits| / |training\_data|$$

where P and N are the positive and the negative matrix respectively, τ is the learning rate (similar to the learning rate in neural networks), and W_p and W_n are the weights. They were defined as the proportion of the total number of positive hits and misses to the training data size and the ratio of the number of negative hits to the training data size, respectively. We repeated the update process to train the substitution matrix until there was no change in the matrix, i.e. the number of both the positive and the negative hits remain constant. The converged matrix was our final substitution matrix which we combined with FASTA as a new alignment tool to demonstrate the applicability of our new alphabet and matrix. We compared our alignment tool with other similar ones on database-scale search tasks. The results were detailed in the next section. The matrix training framework was presented in Figure 1.

Currently, we used the non-redundant proteins in SCOP1.69 with sequence similarity less than 40% for training. We defined the positive hit rate of a query as the ratio of the number of positive hits to the size of the family the query belonged to. As we iterated over each training protein (as a query), we refined the matrix till we could no longer increase the average positive hit rate of all the proteins. One learning example was presented in Figure 2. We tried different learning rates from 0.25 to 1.00. The final average positive hit rates under different learning rates were similar, between 0.9112 and 0.9153. We selected the converged matrix with the maximum positive hit rate when learning rate set 0.50. We named this matrix TRISUM-169 (TRained Iteratively for SUbstitution Matrix-SCOP1.69) as shown in Figure 3.

Experimental Results

Several protein structure search tools based on 1D alignment algorithms have been developed, including SA-Search [25], YAKUSA [30], 3D-BLAST [27], but few were evaluated on the performance of database-scale search. To keep the consistency, we used the same 50 proteins selected from SCOP95-1.69 as used in Yang & Tung's experiment to compare our alignment tool with 3D-BLAST, PSI-BLAST, YAKUSA MAMMOTH and CE in search time, predictive accuracy and precision. There are some other search tools, e.g. PBE [31], SA-Search [30], Vorolign [32] and so on. Because they either could not be tested on the SCOP database directly (e.g. only PDB available in SA-Search) or the version of their databases provided was older (e.g. ASTRAL in PBE derived from SCOP-1.65, Vorolign server only scans SCOP40-1.69), these tools were not chosen for comparison. We summarized the results in Table 1. It showed that our tool outperformed the other two BLAST-based search tools (i.e. 3D-BLAST and PSI-BLAST) and another structure search tool that also described structures as 1D sequences (i.e. YAKUSA) in predictive accuracy and precision. Compared with the structural alignment tools (i.e. MAMMOTH and CE), our tool obtained a bit worse but comparable accuracy as well as precision. As for search time (using one Intel Pentium 2.8GHz processor and 512Mbytes of memory), Table 1 clearly indicated that our alignment tool was far more efficient than the structural alignment tools, MAMMOTH and CE.

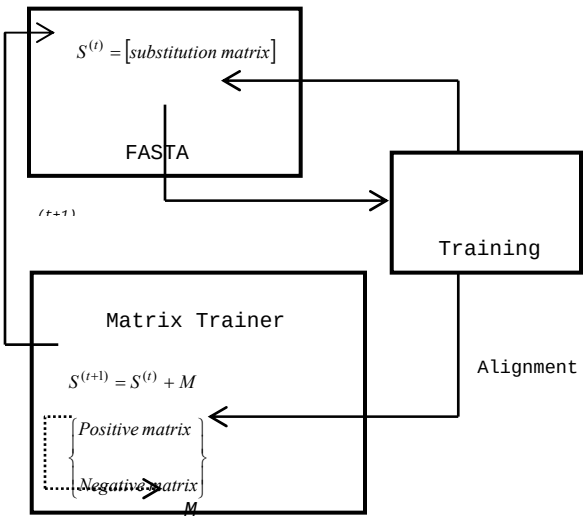

Fig 1. System architecture of the matrix training framework.

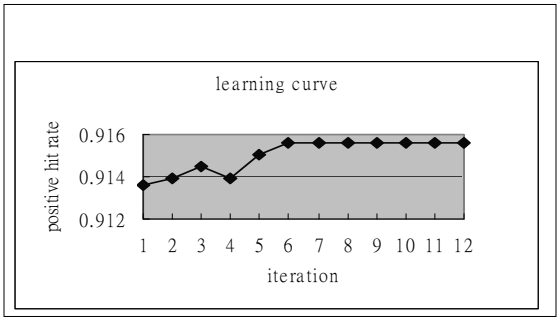

Fig 2. An example of the learning curve of matrix training. The average positive hit rate converged at 0.9153 with the learning rate set 0.5.

To demonstrate the ability of our structural alphabet to describe protein local structure features, we used MEME [24] to detect common motifs in the top 100 hits found by our alignment tool. These motifs could be well mapped to the eight β/α barrel strands of TIM barrel domains. Figure 4(a) showed the structure of archaeon pyrococcus woesei (PDB 1hg3a). In Figure 4(b), we highlighted the identified motif in PDB 1hg3a, and Figure 4(c) illustrated the motif structure. The structural alphabet letter sequence of this motif and the corresponding amino acids were shown in Figure 4(d). In addition to TIM barrel structures, we also used the EGF/EGF-like domain as another study case. Epidermal growth factor (EGF) domains are extracellular protein modules typically described by 30-40 amino acids primarily stabilized by three disulfide bonds. Compared with TIM barrel structures, EGF are much smaller domains. We used it to evaluate how well a structural alphabet could define the 3D structures of small proteins. Many proteins contain the regions of homology to EGF, and the cysteine residues at similar positions. The homologies and available functional data suggest that these domains share some common functional features. If we number the cysteine residues as Cys1 to Cys6, where Cys1 is the closest to the N-terminus, the regularity of cysteine spacing defines three regions, A, B and C. Based on the conservation in sequence and length of these regions, the homologies have been classified into three different categories [33]. We described the 227 proteins in the EGF-type module family of SCOP 1.69 in our alphabet, Yang & Tung's [27] and de Brevern *et al.*'s [15,26,31], respectively. We then used MEME to identify the common motifs corresponding to the sub-domains, A, B and C. According to InterPro [34], 24 of these proteins were exclusively of *EGF Type-1*, 74 were of *EGF-like Type-2*, and 117 belonged to *EGF-like Type-3* only. We classified the remaining 12 proteins as *Others*.

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W
A	5	-3	-4	-3	-2	-4	-4	-4	-4	-3	-4	-4	-3	-3	-4	-3	-7	-3
R	-3	8	-4	-4	-3	-3	-5	-3	-4	-2	-4	-4	-4	-3	-3	-3	-6	-3
N	-4	-4	6	-3	-2	-4	-3	-4	-3	-3	-5	-4	-3	-4	-5	-3	-8	-3
D	-3	-4	-3	10	-3	-3	-4	-2	-3	-2	-5	-4	-2	-4	-6	-4	-8	-2
C	-2	-3	-2	-3	8	-3	-3	-3	-3	-2	-4	-4	-3	-3	-5	-3	-8	-2
Q	-4	-3	-4	-3	-3	8	-6	-4	-4	-1	-3	-4	-4	-3	-2	-3	-5	-4
E	-4	-5	-3	-4	-3	-6	3	-6	-5	-6	-7	-6	-4	-5	-6	-5	-10	-3
G	-4	-3	-4	-2	-3	-4	-6	10	-3	-2	-4	-4	-4	-3	-4	-3	-7	-4
H	-4	-4	-3	-3	-3	-4	-5	-3	9	-2	-4	-4	-3	-4	-4	-3	-7	-2
I	-3	-2	-3	-1	-2	-1	-6	-2	-2	16	-1	0	-2	-1	-1	-1	-3	-2
L	-4	-4	-5	-5	-4	-3	-7	-4	-4	-1	11	-4	-5	-3	-3	-3	-5	-5
K	-4	-4	-4	-4	-4	-4	-6	-4	-4	0	-4	11	-4	-4	-4	-3	-6	-4
M	-3	-4	-3	-2	-3	-4	-4	-4	-3	-2	-5	-4	10	-4	-6	-4	-10	-3
F	-3	-3	-4	-4	-3	-3	-5	-3	-4	-1	-3	-4	-4	10	-3	-2	-5	-3
P	-4	-3	-5	-6	-5	-2	-6	-4	-4	-1	-3	-4	-6	-3	9	-2	-4	-4
S	-3	-3	-3	-4	-3	-3	-5	-3	-3	-1	-3	-3	-4	-2	-2	9	-5	-4
T	-7	-6	-8	-8	-8	-5	-10	-7	-7	-3	-5	-6	-10	-5	-4	-5	3	-8
W	-3	-3	-3	-2	-2	-4	-3	-4	-1	-2	-5	-4	-3	-3	-4	-4	-8	8

Fig 3. Substitution matrix TRISUM-169.

Despite that the sub-domains are less conserved in EGF-like Type-3, sub-domain A is typically composed of five to six residues in

Type-1 and 2, sub-domain B usually contains 10-11 residues in Type-1, but consistently three residues shorter than in Type-1, sub-domain C is conserved in length with four or five specific residues in Type-1 and 2 [33]. We used 8, 10 and 15 respectively as the motif width and ran MEME to find motifs. A motif found was considered as corresponding to a sub-domain correctly if more than half of the residues in the sub-domain were included in the motif. If any single motif of width 8, 10 or 15 alphabet letters correctly corresponded to a sub-domain, we claimed this sub-domain was recovered successfully (i.e. a hit). We summarized the results of the motifs found in Table 2. It showed that with our structural alphabet MEME was able to identify more EGF sub-domains than using Yang & Tung's or de Brevern *et al*.'s alphabets.

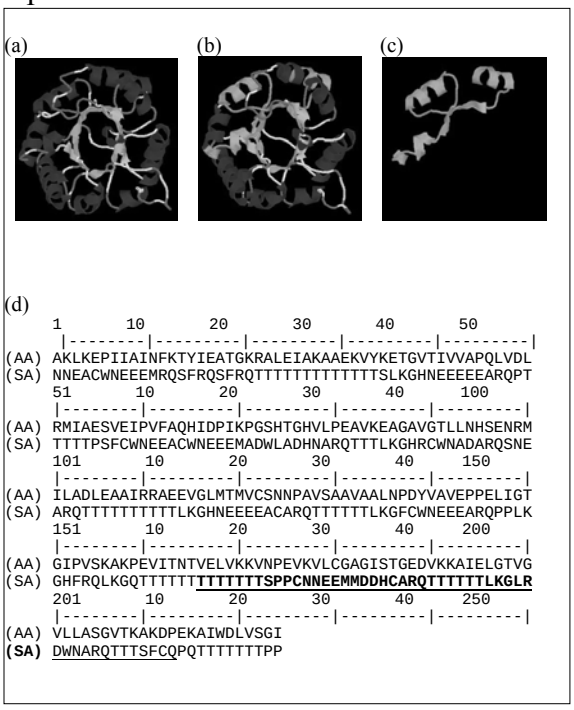

Fig. 4. Common motif found by MEME in PDB 1hg3a. (a) TIM barrel structure of PDB 1hg3a (b) motif highlighted in green (c) motif structure (d) PDB 1hg3a described in amino acids (AA) and structural alphabet (SA), respectively, where motif underlined. (Note. Images are shown in grey scale.)

4 Discussion

The protein structure data we used to build the alphabet were from the non-redundant PDB database instead of some specialized databases, e.g. Pair Database [27] and PDB-SELECT [29], with the aim to ensure the generality of our alphabet. We also proposed an automatic matrix training framework to construct an appropriate substitution matrix for the alphabet. This training strategy did not need any information of known alignments that most previous works required. Using different training data and update rules, the self-training methodology can be applied to various alphabets.

To demonstrate the performance of our alignment tool, we systematically compared it with other search tools. The results showed that our new tool was very competitive in predictive accuracy and alignment efficiency for database-scale search. We further evaluated the potential of using motif-finding tools, e.g. MEME, to detect structure domains/sub-domains represented in our structural alphabet. Two examples of different protein classes, TIM in α/β and EGF in small proteins, have been tested. The results indicated that the identified motifs mapped well to the known structure sub-domains.

We can extend the work in several directions. First, we can use a more complete datasets for substitution matrix training to increase sensitivity and selectivity in database search. Second, besides FASTA, we can combine other alignment tools with our substitution matrix, and evaluate the performance of different combinations. Third, currently we use MEME to detect motifs, and we have demonstrated it is able to recover some structure sub-domains described in our structural alphabet. MEME was originally designed to find motifs in amino acid and nucleic acid sequences. To increase the performance in structural motif detection, we can either modify MEME or develop a new motif-finding tool specifically for our structural alphabet. Finally, several structural alphabets have been developed based on different protein structural characteristics. It is worthwhile to conduct a thorough comparative study and evaluate the feasibility of combining different alphabets. The combination of structural alphabets that complement each other will increase their overall applicability and characterize 3D protein structures more completely.

5 References

[1] A.G. Murzin, S.E. Brenner, T. Hubbard and C. Chothia "SCOP: A Structural Classification of Proteins Database for the Investigation of Sequences and Structures", *J. Mol. Biol*., 1995, 536-540.

[2] R. Unger, D. Harel, S. Wherland and J.L. Sussman "A 3D building blocks approach to analyzing and predicting structure of proteins", *Proteins*, 1989, 355-373.

[3] M. Dudev and C. Lim "Discovering structural motifs using a structural alphabet: Applications to magnesium-binding sites", *BMC Bioinformtics*, 2007, 106.

[4] R. Aurora, R. Srinivasan and G.D. Rose "Rules for alpha-helix termination by glycine", *Science*, 1994, 1126-1130.

[5] R. Unger and J.L. Sussman "The importance of short structural motifs in protein structure analysis", *J. Comput. Aided Mol. Des*., 1993, 457-472.

[6] Z.Y. Zhu and T.L. Blundell "The use of amino acid patterns of classified helices and strands in secondary structure prediction", *J. Mol. Biol.*, 1996, 261-276.

[7] K.F. Han and D. Baker "Recurring local sequence motifs in proteins", *J. Mol. Biol*., 1995, 176-187.

[8] K.T. Simons, C. Kooperberg, E. Huang and D. Baker "Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and Bayesian scoring functions", *J Mol Biol*., 1997, 209 – 225.

[9] J. Garnier, D. Osguthorpe and B. Bobson "Analysis of the accuracy and implications of simple methods for predicting the secondary structure of globular protein", *Journal of Molecular Biology*, vol. 120, 1978, pp. 97-120.

[10] B. Rost and C. Snader, "Prediction of protein secondary structure at better than 70% accuracy", *Journal of Molecular Biology*, vol. 232, 1993, pp. 584-599.

[11] A. Salamov and V. Solovyev, "Protein secondary structure prediction using local alignments", *Journal of Molecular Biology*, vol. 268, 1997, pp. 31-36.

[12] T.N. Petersen, C. Lundegaard, M. Nielsen, H. Bohr, J. Bohr, S. Brunak, G.P. Gippert and O. Lund, "Prediction of protein secondary structure at 80% accuracy", *Proteins*, vol. 41, 2000, pp. 17-20.

[13] B. Rost, "Review: Protein secondary structure prediction continues to rise," *Journal of Structural Biology*, vol. 134, 2001, pp. 204-218.

[14] A.G. de Brevern and S.A. Hazout, "Hybrid Protein Model(HPM): a method to compact protein 3D-structure

information and physicochemical properties", *IEEE Comp. Soc*. S1, 2000, pp. 49-54.

[15] A.G. de Brevern, H. Valadie, S.A. Hazout and C. Etchebest, "Extension of a local backbone description using a structural alphabet: A new approach to the sequence-structure relationship," *Protein Science*, vol. 11, 2002, pp. 2871-2886.

[16] R. Unger, D. Harel, S. Wherland and J.L. Sussman, "A 3D building blocks approach to analyzing and predicting structure of proteins", *Proteins*, vol. 5, 1989, pp. 355-373.

[17] J. Schuchhardt, G. Schneider, J. Reichelt, D. Schomburg and P. Wrede, "Local structural motifs of protein backbones are classified by self-organizing neural networks", *Protein Engineering*, vol. 9, 1996, pp. 833-842.

[18] M.J. Rooman, J. Rodriguez and S.J. Wodak, "Automatic definition of recurrent local structure motifs in proteins", *Journal of Molecular Biology*, vol. 213, 1990, pp. 327-336.

[19] J.S. Fetrow, M.J. Palumbo and G. Berg, "Patterns, structures, and amino acid frequencies in structural building blocks, a protein secondary structure classification scheme", *Proteins*, vol. 27, 1997, pp. 249-271.

[20] C. Bystroff and D. Baker, "Prediction of local structure in proteins using a library of sequence-structure motif", *Journal of Molecular Biology*, vol. 281, 1998, pp. 565-577.

[21] A.C. Camproux, R. Gautier and P. Tuffery, "A hidden Markov model derived structural alphabet for proteins", *Journal of Molecular Biology*, doi: 10.1016/j.jmb.2004.04.005.

[22] S. Ku and Y. Hu "A Multi-strategy Approach to Protein Structural Alphabet Design", *Biocomp* 2006.

[23] W.R. Pearson "Flexible sequence similarity searching with the FASTA3 program package", *Methods Mol. Biol.*, 2000, 185-219.

[24] T.L. Bailey and C. Elkan "Unsupervised learning of multiple motifs in biopolymers using EM", *Machine Learning*, 1995, 51-80.

[25] F. Guyon, A.C. Camproux, J. Hochez and P. Tuffery "SA-Search: a web tool for protein structure mining based a structural alphabet", *Nucleic Acids Res*., 2004, W545–W548.

[26] M. Tyagi, V.S. Gowri, N. Srinivasan, A.G. de Brevern and B. Offmann "A substitution matrix for structural alphabet based on structural alignment of homologous proteins and its applications", *Proteins: Structure, Function and Bioinformatics*, 2006, 32-39.

[27] J.M. Yang and C.H. Tung "Protein structure databases search and evolutionary classification", *Nucleic Acids Research*, 2006, 3646-3659.

[28] S. Henikoff and J.G. Henikoff "Amino acid substitution matrices from protein blocks", *PNAS*, 1992, 10915-10919.

[29] W.M. Zheng and X. Liu "A protein structural alphabet and its substitution matrix CLESUM", *LNCS*, 2005, 59-67.

[30] M. Carpentier, S. Brouillet and J. Pothier "YAKUSA: a fast structural database scanning method", *Proteins: Structure, Function and Genetics*, 2005, 137-151.

[31] M. Tyagi, P. Sharma, C.S. Swamy, F. Cadet, N. Srinivasan, A.G. de Brevern and B. Offmann "Protein Block Expert (PBE): a web-based protein structure analysis server using structural alphabet", *Nucleic Acids Research*, 2006, W119-123.

[32] F. Birzele, J.E. Gewehr, G. Csaba and R. Zimmer "Vorolign- fast structural alignment using Voronoi contacts", *Bioinformatics*, 2007, e205-211.

[33] E. Appella, I.T. Weber and F. Blasi "Structure and function of epidermal growth factor-like regions in proteins", *FEBS Letters*, 1988, 1-4.

[34] N.J. Mulder *et al.* "New developments in the InterPro database", *Nucleic Acids Research*, 2007, D224-228.

Table 1. Comparison between our alignment tool, 3D-BLAST, PSI-BLAST, YAKUSA, MAMMOTH and CE on 50 proteins selected fromSCOP95-1.69.

Search tool	Average time required for a query (sec)	Relative to SA-FAST	Accuracy (%)	Average precision (%)
Our Tool	1.15	1.00	96	90.80
3D-BLAST	1.30	1.13	94	85.20
PSI-BLAST	0.48	0.42	84	68.16
YAKUSA	8.88	7.72	90	74.86
MAMMOTH	1834.18	1594.94	100	94.01
CE	22053.32	19176.80	98	90.78

Table 2. Comparison between our structural alphabet, Yang & Tung's and de Brevern *et al.*'s in describing motifs found by MEME within EGF family.

(a) Number of motifs found by MEME, using different structural alphabets to describe EGF (EGF-like) proteins

		Our SA						Yang & Tung's						de Brevern *et al.*'s					
Sub-domain Type		**A**		**B**		**C**		**A**		**B**		**C**		**A**		**B**		**C**	
EGF proteins	**No.[a]**	**Hits[b]**	**Cov[c]**	**Hits**	**Cov**	**Hits**	**Cov**	**Hits**	**Cov**	**Hits**	**Cov**	**Hits**	**Cov**	**Hits**	**Cov**	**Hits**	**Cov**	**Hits**	**Cov**
Type 1	24	23	95.8	22	91.7	23	95.8	11	45.8	21	87.5	19	79.2	18	75.0	14	58.3	18	75.0
Type 2	74	73	98.6	71	95.9	74	100.0	62	83.8	73	98.6	60	81.1	68	91.9	62	83.8	70	94.6
Type 3	117	116	99.1	106	90.6	61	52.1	54	46.2	102	87.2	25	21.4	109	93.2	112	95.7	48	41.0
Others	12	12	100.0	11	91.7	11	91.7	9	75.0	11	91.7	9	75.0	12	100.0	11	91.7	9	75.0
All	227	224	98.6	210	92.5	169	74.4	136	59.9	207	91.2	113	49.8	207	91.2	199	87.7	145	63.9

[a]The number of EGF proteins of a specific type, [b]We called it a hit for a sub-domain when more than half of the sub-domain residues were contained in a motif. We presented the count of hits of different types, [c]Cov(Coverage) was defined as the ratio of the count of hits to the number of EGF proteins, e.g., if No.=24 and Hits=22, then Cov=22/24=91.7%.

(b) Statistics of EGF (EGF-like) proteins whose sub-domains detected by MEME

	Structural Alphabet					
EGF proteins	**Our SA**		**Yang & Tung's**		**de Brevern *et al.*'s**	
	Count	**Percentage**	**Count**	**Percentage**	**Count**	**Percentage**
Found 3[a]	151	66.52	79	34.80	104	45.81
Found 2[b]	74	32.60	78	34.36	116	51.10
Found 1[c]	2	0.88	63	27.75	7	3.08
Found 0[d]	0	0.00	7	3.08	0	0.00
Total	227	100.00	227	100.00	227	100.00

[a]EGF (EGF-like) proteins in which all three sub-domains (A, B and C) were found by MEME, [b]EGF (EGF-like) proteins in which two out of three sub-domains were found by MEME, [c]EGF (EGF-like) proteins in which only one sub-domain was found by MEME, [d]EGF (EGF-like) proteins in which MEME failed to identify any sub-domain.