BIOINFORMATICS

Vol. 20 no. 16 2004, pages 2586–2596
doi:10.1093/bioinformatics/bth290

Incremental generation of summarized clustering hierarchy for protein family analysis

Chien-Yu Chen[1,*], Yen-Jen Oyang[1] and Hsueh-Fen Juan[2]

[1]Department of Computer Science and Information Engineering, National Taiwan University, Taipei 106, Taiwan, R.O.C. and [2]Institute of Biotechnology and Department of Chemical Engineering, National Taipei University of Technology, Taipei 106, Taiwan, R.O.C.

Received on November 18, 2003; revised on March 28, 2004; accepted on April 25, 2004
Advance Access publication May 6, 2004

ABSTRACT

Motivation: Protein sequence clustering has been widely exploited to facilitate in-depth analysis of protein functions and families. For some applications of protein sequence clustering, it is highly desirable that a hierarchical structure, also referred to as dendrogram, which shows how proteins are clustered at various levels, is generated. However, as the sizes of contemporary protein databases continue to grow at rapid rates, it is of great interest to develop some summarization mechanisms so that the users can browse the dendrogram and/or search for the desired information more effectively.

Results: In this paper, the design of a novel incremental clustering algorithm aimed at generating summarized dendrograms for analysis of protein databases is described. The proposed incremental clustering algorithm employs a statistics-based model to summarize the distributions of the similarity scores among the proteins in the database and to control formation of clusters. Experimental results reveal that, due to the summarization mechanism incorporated, the proposed incremental clustering algorithm offers the users highly concise dendrograms for analysis of protein clusters with biological significance. Another distinction of the proposed algorithm is its incremental nature. As the sizes of the contemporary protein databases continue to grow at fast rates, due to the concern of efficiency, it is desirable that cluster analysis of a protein database can be carried out incrementally, when the protein database is updated. Experimental results with the Swiss-Prot protein database reveal that the time complexity for carrying out incremental clustering with k new proteins added into the database containing n proteins is $O(n^{2\beta}\log n)$, where $\beta \cong 0.865$, provided that $k \ll n$.

Availability: The Linux executable is available on the following supplementary page.

Contact: Graduate School of Biotechnology and Bioinformatics, Yuan–Ze University, Chang–Li 320, Taiwan, ROC. Email: cychen@mars.csie.ntu.edu.tw

Supplementary information: http://mars.csie.ntu.edu.tw/~cychen/protein_clustering/psc.htm

*To whom correspondence should be addressed.

1 INTRODUCTION

Protein sequence clustering is a process that aims to identify sets of homologous proteins in a protein database (Lesk, 2002; Kriventseva *et al*., 2001a). The information derived from protein sequence clustering is then widely used for further analysis such as protein family discovery, function prediction and database compression (Abascal and Valencia, 2002; Apweiler *et al*., 2001; Enright *et al*., 2002; Li *et al*., 2001, 2002; Kriventseva *et al*., 2001a; Sasson *et al*., 2003; Yona *et al*., 1999). The general practice to carry out protein sequence clustering is based on pairwise sequence similarity/dissimilarity between two proteins computed by algorithms such as Smith–Waterman (Smith and Waterman, 1981), BLAST (Altschul *et al*., 1990, 1997) and FASTA (Pearson and Lipman, 1998). The widely adopted hypothesis is that a high degree of sequence similarity between two proteins implies that these two proteins also have similar structures and/or functions (Dayhoff, 1976; Hegyi and Gerstein, 1999; Lesk, 2002).

In latest studies of protein sequence clustering, the single-link clustering algorithm (Koonin *et al*., 1995; Kriventseva *et al*., 2001b; Watanabe and Otsuka, 1995) and alternative graph-based clustering algorithms (Bolten *et al*., 2001; Enright *et al*., 2002; Kawaji *et al*., 2001; Matsuda *et al*., 1996) are most commonly employed, due to the clustering quality that these two types of algorithms deliver. In protein sequence clustering, a popular measure of clustering quality is based on how well the clusters identified by the clustering algorithm match the protein families defined in some databases (Kawaji *et al*., 2001; Yona *et al*., 1999).

For some applications of protein sequence clustering, it is highly desirable that a hierarchical structure, also referred to as dendrogram, which shows how proteins are clustered at various levels, is generated (Lesk, 2002; Sasson *et al*., 2002; Kriventseva *et al*., 2001b). However, as the sizes of contemporary protein databases continue to grow at rapid

rates, the number of nodes and the depth of the dendrogram generated by the clustering algorithm have become too large for the users to effectively browse the dendrogram and/or to search for the desired information. As a result, the user may need to impose some thresholds to flatten the dendrogram so that the visualization and interpretative quality of the dendrogram is improved. Nevertheless, figuring out the proper threshold values may be a tedious and difficult task for the users, especially for naive users. Therefore, it is of great interest to develop some summarization mechanism for presenting the results of protein sequence clustering.

In this paper, the design of a novel incremental clustering algorithm aimed at generating summarized dendrograms for analysis of protein databases is described. The proposed incremental clustering algorithm employs a statistics-based model to summarize the distributions of the similarity scores among the proteins in the database and to control formation of clusters. Due to the summarization operations employed, the dendrogram generated by the proposed algorithm offers better visual and interpretative quality than the dendrogram generated by the conventional single-link algorithm. In this paper, the weighted average matching rate is employed to measure how well a clustering algorithm can cluster proteins in conformity with human's interpretation and no cutoff threshold is imposed to flatten the dendrogram generated by the single-link algorithm. The dendrogram generated by the proposed incremental clustering algorithm contains much fewer non-leaf nodes than that generated by the single-link algorithm. Furthermore, those clusters identified by the single-link algorithm that best match the protein families defined in the InterPro (Apweiler *et al*., 2000) are deeply embedded in the dendrogram. On the other hand, in the dendrogram generated by the proposed incremental clustering algorithm, most of those clusters that best match the protein families defined in the InterPro are located just one level down from the root. Due to the summarization mechanism incorporated, the proposed incremental clustering algorithm offers the users highly concise dendrograms for analysis of protein clusters with biological significance.

Another main property of the proposed incremental clustering algorithm is that, when the protein database is updated, there is no need to redo all the clustering analysis from scratch. Instead, the incremental clustering algorithm can refer to an abstraction generated by the previous run of the algorithm and carry out the analysis much more efficiently. This issue is of significance, as contemporary protein databases, such as Swiss-Prot (Bairoch and Apweiler, 2000) and PIR (Wu *et al*., 2002), keep growing rapidly. Experimental results with the Swiss-Prot protein database reveal that the time complexity for carrying out incremental clustering with k new proteins added into the database containing n proteins is $O(n^{2\beta}\log n)$, where $\beta \sim 0.865$, provided that $k \ll n$.

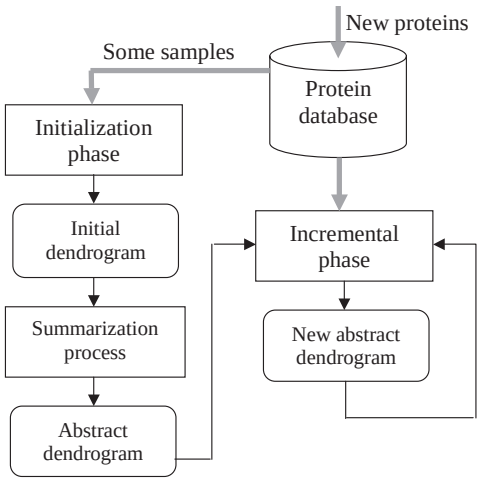

Fig. 1. A system diagram that summarizes the operations carried out by the proposed incremental clustering algorithm.

The remaining part of this paper is organized as follows. Section 2 elaborates the design of the proposed incremental clustering algorithm. Section 3 reports the experiments conducted to evaluate the performance of the proposed algorithm. Section 4 concludes the discussion of this paper.

2 METHODS AND ALGORITHMS

2.1 Overview of the incremental clustering algorithm

Figure 1 shows the major operations carried out by the incremental clustering algorithm presented in this paper. The algorithm consists of two phases of operations, namely, the initialization phase and the incremental phase. In the initialization phase, a set of proteins extracted from the protein database is taken to construct the initial dendrogram. The number of proteins extracted could range from a few hundreds to a few thousands. With the initial set of proteins, a conventional agglomerative hierarchical clustering algorithm, such as single-link or complete-link (Han and Kamber, 2000; Jain and Dubes, 1988), is invoked to construct the initial dendrogram. In this paper, the single-link algorithm is employed for carrying out the initial protein clustering.

With the initial dendrogram, a summarization process is then conducted to identify a set of representatives that collectively provide an abstract description of the distribution of the samples. With these representatives, an abstract dendrogram is generated. The abstract dendrogram provides the basis for the incremental phase of the clustering algorithm to proceed. In the incremental phase, all the remaining proteins in the database, i.e. those proteins that were not taken as samples, as well as the new proteins that are continuously added into the database are examined one by one and the abstract dendrogram is updated dynamically to reflect the evolution of the protein database.

In the following three subsections, how the representatives are identified and how clustering is carried out incrementally are elaborated.

2.2 The summarization process

In the summarization process, protein clusters that meet the following statistical criteria are identified as homogeneous clusters. The statistical criteria are imposed to guarantee that the distribution of the pairwise similarity scores among the protein sequences in a homogeneous cluster has a symmetrical unimodal distribution (Jobson, 1991). In the following discussion, we treat the pairwise similarity scores among the protein sequences in a cluster denoted by C as $|C|\times(|C|-1)/2$ random samples of random variable X_C. Let S_C denote the set of $|C| \times (|C| - 1)/2$ random samples. In statistics, the corresponding skewness and kurtosis, denoted by Skew(C) and Kurt(C), respectively in this paper, are defined as follows (Jobson, 1991):

$$\text{Skew}(C) = \frac{|S_C|}{(|S_C|-1)\times(|S_C|-2)} \sum_{x_i \in S_C} (x_i - \bar{x})^3/s^3,$$

$$\text{Kurt}(C) = \left[\frac{|S_C|\times(|S_C|+1)}{(|S_C|-1)\times(|S_C|-2)\times(|S_C|-3)} \times \sum_{x_i \in S_C} (x_i - \bar{x})^4/s^4\right] - 3\frac{(|S_C|-1)^2}{(|S_C|-2)\times(|S_C|-3)},$$

where

$$\bar{x} = \frac{1}{|S_C|}\sum_{x_i \in S_C} x_i, \text{ and}$$

$$s = \sqrt{\frac{1}{|S_C|-1}\sum_{x_i \in S_C}(x_i - \bar{x})^2}.$$

With skewness and kurtosis, thresholds are set to guarantee that each homogeneous cluster has a symmetric unimodal distribution of the pairwise similarities. By setting the lower bound of kurtosis, we can guarantee that a group of protein sequences with a multimodal distribution will be decomposed into a number of homogeneous clusters. By setting both the upper and lower bounds of skewness, we can guarantee that the distribution of the pairwise similarities among the proteins in a homogeneous cluster is symmetric. An asymmetric distribution implies that a small subgroup of proteins in the cluster has the pairwise similarities among the subgroup either much higher or much lower than the pairwise similarities among the remaining proteins in the cluster. Accordingly, we define the homogeneous cluster as follows.

DEFINITION 1 (*Homogeneous cluster*). *A protein cluster C is said to be homogeneous, if the corresponding skewness and kurtosis satisfy the following criterion:*

$$-\theta_s \le \textit{Skew}\ (C) \le \theta_s, \textit{ and}$$
$$\textit{Kurt}\ (C) \ge \theta_k,$$

where θ_s and θ_k are two thresholds to be set by the user. In this paper, θ_s and θ_k are set to 1 and 0, respectively, based on experiences learned with extensive experiments.

The criterion of homogeneous clusters presented above is only applied to clusters containing more than Min_Cardinality proteins. For clusters that contain less than Min_Cardinality proteins, a different criterion is applied. The reason why a different criterion is imposed is that a cluster must contain a sufficient number of proteins for the tests of skewness and kurtosis to be meaningful. The criterion applied to small clusters is that all the pairwise similarities between two proteins in a small cluster must be larger than a threshold denoted by Min_Similarity. Min_Similarity is imposed to guarantee that every homogeneous cluster containing less than Min_Cardinality proteins meets a certain quality criterion. By default, each leaf node in the dendrogram is regarded as a homogeneous cluster containing one single protein. For those clusters that continue to grow in size, the statistical tests that involve the skewness and kurtosis of the cluster will eventually be imposed.

DEFINITION 2 (*Leaf homogeneous cluster*). *A cluster C in a dendrogram is said to be a leaf homogeneous cluster, if C and all of its children are homogeneous clusters and the parent of C is not satisfied as a leaf homogeneous cluster.*

In the summarization process, one protein in a homogeneous cluster will be designated as the representative of the cluster. The representative of a homogeneous cluster C is the protein with the maximum lumped sum of the similarity scores to the other proteins in C. In this paper, the representative of protein cluster C is denoted by Rep(C).

With homogeneous clusters and their representatives identified, the summarization process then conducts a bottom-up flattening operation on the dendrogram generated by the agglomerative hierarchical clustering algorithm. The bottom-up flattening operation begins with the leaf homogeneous clusters. All the subclusters under a leaf homogeneous cluster will be removed and all the proteins contained in the leaf homogeneous cluster will become its children. Figure 2 illustrates the bottom-up flattening operation. Figure 2a shows the initial dendrogram constructed with 100 objects. In Figure 2a, those clusters marked by an asterisk are leaf homogenous clusters according to Definition 2. Figure 2b shows the dendrogram after the bottom-up flattening operation has been applied to the leaf homogeneous clusters.

The bottom-up flattening operation described above is conducted recursively with each of the leaf homogeneous clusters identified in one level of recursion being substituted by its representative protein. Figure 2c depicts the dendrogram

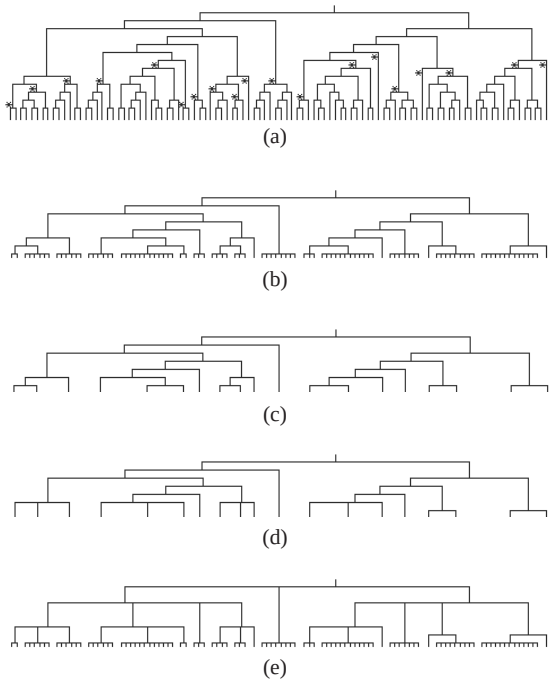

Fig. 2. An example illustrating the flattening operation of the summarization process. (**a**) A dendrogram with nodes passing the criterion of leaf homogeneous cluster marked. (**b**) The summarized dendrogram after flattening the leaf homogeneous clusters. (**c**) The dendrogram derived from that in (b) by substituting each leaf homogeneous cluster with its representative. (**d**) The dendrogram generated after the second recursion of the bottom-up flattening operation is applied. (**e**) The final dendrogram.

derived from substituting the leaf homogeneous clusters in Figure 2b with their respective representatives and Figure 2d shows the flattened dendrogram after the second level of recursion is conducted. Figure 2e shows the final dendrogram after the bottom-up flattening operation is completed and this dendrogram is referred to as the abstract dendrogram.

2.3 The incremental process

With the flattened dendrogram, the incremental phase of the clustering algorithm is then carried out to cluster the remaining proteins in the database, i.e. those proteins that are not taken to construct the initial dendrogram, as well as the proteins that may be added into the database later on. These proteins are examined one by one. For each protein, the incremental clustering algorithm examines whether the protein can be inserted into a leaf homogeneous cluster according to the following criteria. The criterion for inserting a protein p into a leaf homogeneous cluster C is as follows:

$$\left|\frac{\mu_C - \text{sim}[p, \text{Rep}(C)]}{\sigma_C}\right| \leq \theta_q \quad \text{and}$$

$$\text{sim}[p, \text{Rep}(C)] \geq \underset{C'}{\forall} \text{sim}[p, \text{Rep}(C')],$$

where

(1) μ_C and σ_C are the mean and standard deviation of the pairwise similarities in C;

(2) C' is a leaf homogeneous cluster of size larger than 2;

(3) sim[p, Rep(C)] denotes the similarity between protein p and protein Rep(C);

(4) θ_q is a parameter and is set to 1 in this paper for carrying out protein sequence clustering.

Each time a protein is inserted into a leaf homogeneous cluster, the skewness and kurtosis of the cluster will be recomputed to check whether the cluster still meets the criterion of being homogeneous. Should the cluster, with the protein inserted, fails to pass the test, a split operation will be conducted. The single-link algorithm and the flattening operation described in Section 2.2 will be invoked to construct a subdendrogram containing all the proteins in the cluster and the newly added protein. The split operation and the reconstruction operation described in the next paragraph are essential for avoiding order dependence, which is a common problem in many incremental clustering algorithms.

In case the protein being examined cannot be inserted into any of the existing leaf homogeneous clusters, then the protein is temporarily moved to a temporary buffer called TempBuffer and will be processed again later on. Every time TempBuffer becomes full, a reconstruction operation is conducted to generate a new abstract dendrogram containing all the proteins in the current dendrogram and in TempBuffer. In the reconstruction operation, the primitive objects are the representatives of the leaf homogeneous clusters and the proteins in TempBuffer. In this paper, the single-link algorithm is invoked. During the reconstruction process, two leaf homogenous clusters in the original dendrogram may merge due to inclusion of the proteins in TempBuffer. As mentioned earlier, merge along with the split operation described above are essential for avoiding order dependence, and alternative forms of these two operations have been widely employed in the design of incremental clustering algorithms (Fisher, 1987; Zhang *et al*., 1996).

2.4 Reducing the skewness of the dendrogram

In this paper, an optional top-down flattening operation is developed to reduce the skewness of the dendrogram, i.e. to make the dendrogram more balanced. The optional top-down operation creates a superroot that includes the nodes at the highest levels of the dendrogram. Figure 3a shows a dendrogram generated by the proposed incremental clustering algorithm after the bottom-up flattening operation has been applied. Figure 3b depicts the distribution of the heights of the homogeneous clusters in the dendrogram. The height of a cluster is defined to be the number of nodes on the longest path from the cluster to a leaf homogeneous cluster. In Figure 3b, all the vertical bars with horizontal coordination larger than or

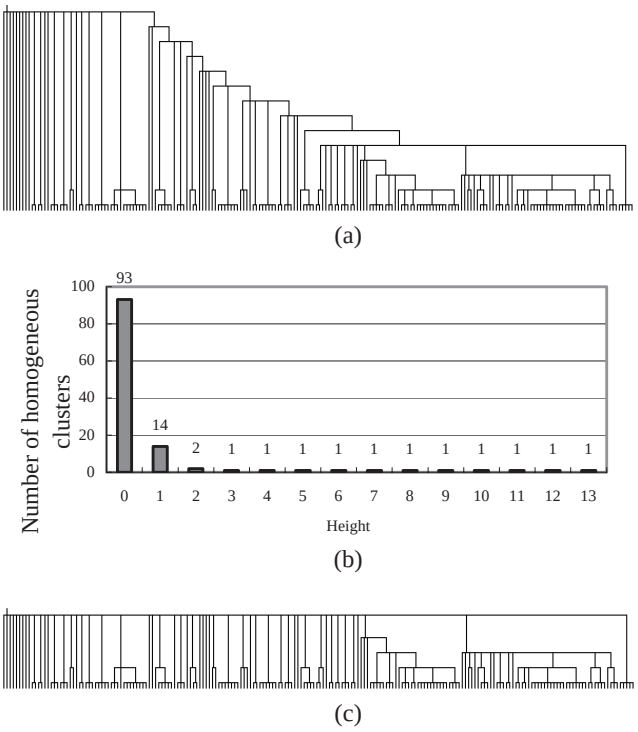

Fig. 3. An example that illustrates the top-down flattening operation to reduce the skewness of the dendrogram. (**a**) A dendrogram generated by the proposed incremental clustering algorithm after the bottom-up flattening operation has been applied. (**b**) Distribution of the heights of the clusters in the dendrogram shown in (a). (**c**) The final dendrogram after the top-down flattening operation has been applied.

equal to 3 have values equal to 1. This implies that all nodes with height larger than 3 have skewed child-dendrograms. In this case, the top-down flattening operation is invoked to create a superroot that contains all the nodes with height larger than 3 and the result is shown in Figure 3c. In fact, for each dendrogram, there exists a value θ_h such that the number of nodes in the dendrogram with height h is equal to 1, if $h \geq \theta_h$. The top-down flattening operation simply creates a superroot that contains all the nodes with height larger than θ_h.

In our implementation of the incremental phase, a heuristic mechanism that exploits the nature of the single-link algorithm is employed to accelerate the reconstruction operation described above. The detailed description of the accelerated reconstruction process can be found in (Chen, 2003, http://mars.csie.ntu.edu.tw/~cychen/PhDThesisChen2003.pdf).

2.5 Analysis of time complexity

The completed analysis of time and space complexities of the proposed incremental clustering algorithm can be found in (Chen, 2003, http://mars.csie.ntu.edu.tw/~cychen/PhDThesisChen2003.pdf). In summary, if given that the time complexity of single-link algorithm is $O(n^2 \log n)$, the time complexity of the proposed algorithm would be $O(km) + O(kq^2 \log q) + O((k/b\_s)m^2 \log m))$, where

(1) n: the number of the proteins in the current version of the protein database;
(2) k: the number of new proteins to be added to the protein database;
(3) m: the number of leaf homogeneous clusters in the current version of the abstract dendrogram;
(4) q: the number of proteins that the largest leaf homogeneous cluster contains;
(5) b_s: the size of the TempBuffer.

As the experiments reported in next section reveal, for cluster analysis of contemporary protein databases, we generally have $q \ll m$. Therefore, the dominant term of the time complexity for carrying out protein sequence clustering with the proposed incremental clustering algorithm is $O(km) + O((k/b\_s)m^2 \log m))$ or $O(km^2 \log m)$, if b_s is regarded as a constant.

The analysis presented above shows that the time complexity of the proposed incremental clustering algorithm is determined by how m increases as new proteins continue to be added into the protein database. As in no case m could exceed n, the upper bound of the time complexity is $O(kn^2 \log n)$. On the other hand, if m does not increase as new proteins continue to be added into the protein database beyond a certain point, then m can be treated as a constant and the time complexity is $O(k)$. In the experiments reported in this paper, it is observed that, if $n \gg k$, then in general we have the time complexity equal to $O(kn^{2\beta} \log n)$ or $O(n^{2\beta} \log n)$, if k is treated as constant, where $0 < \beta \leq 1$. In particular, in the experiment conducted to cluster all the proteins in Swiss-Prot, it is observed that $\beta = 0.865$. This paper also reports the results from several additional experiments conducted to study the correlation between β and the characteristics of the datasets. The general observation is that if the dataset contains a large number of highly similar pairs between the k new proteins and the n proteins that the database originally contains, then β tends to be smaller. Otherwise, β tends to be larger. In the additional experiments, β ranges from 0.830 to 0.979. In fact, 0 and 1 are the theoretical lower bound and upper bound of β, respectively, as the time complexity of generating a new dendrogram with k new proteins added into a protein database containing n proteins is bounded between $O(k)$ and $O(kn^2 \log n)$.

Another issue that deserves further analysis is the quantity of pairwise protein–protein similarity scores that must be computed. The total number of pairwise similarity scores that must be computed for including one new protein into the database could be as high as $m + q$. As mentioned earlier, we typically have $q \ll m$ and thus, for adding k new proteins into the

Table 1. Parameter settings employed in the experiments for the proposed incremental clustering algorithm

Parameter	Value
θ_s	1
θ_k	0
θ_q	1
Size of TempBuffer (b_s)	500 or 5000
Min_Cardinality	10 for leaf homogeneous clusters; 5 for non-leaf homogeneous clusters
Min_Similarity	90 (bit-score) for leaf homogeneous clusters; no constraint for non-leaf homogeneous clusters

Table 2. Characteristics of the four datasets used in the experiments

Dataset	Number of proteins	Number of cross-referenced InterPro Families	Number of proteins labeled with family identification
Mouse	4708	861	2563
Human	7471	1067	3796
Rat	2916	714	1902
SP-41	122 564	4212	82 194

database, the total number of pairwise similarity scores that must be computed is in the order of O(km).

3 EXPERIMENTAL RESULTS

This section reports the experiments conducted to evaluate the performance of the proposed incremental clustering algorithm. Section 3.1 addresses the clustering and summarization qualities delivered by the proposed incremental clustering algorithm. Section 3.2 analyzes the execution time of the proposed incremental clustering algorithm. Table 1 shows the parameter settings employed in these experiments for the proposed incremental clustering algorithm and Table 2 summarizes the characteristics of the four datasets used in this study. Datasets Mouse, Human and Rat contain the proteins belonging to mouse, human and rat, respectively, in Swiss-Prot (Release 40.0, October 2001). Dataset SP-41 contains all the proteins in Release 41.0 (2003) of Swiss-Prot. In the experiments, parameter b_s in Table 1 is set to 500 for the three smaller datasets Mouse, Human and Rat, and is set to 5000 for dataset SP-41 due to its size. In addition, as we have attempted to emulate the environment in which new protein sequences are continuously added into the database, the same numbers of protein sequences are taken from the beginning of the input datasets for construction of the initial dendrograms. In this paper, the bit-scores computed by the BLAST algorithm (Altschul *et al.*, 1990, 1997) with BLOSUM62 table are employed.

3.1 Evaluation of clustering and summarization qualities

This section reports the experiments conducted to evaluate the clustering and summarization qualities of the incremental clustering algorithm proposed in this paper. With respect to protein sequence clustering, a popular measure of clustering quality quantifies how well the clusters identified by the clustering algorithm match the protein families identified by biochemists (Kawaji *et al.*, 2001; Yona *et al.*, 1999). Let S denote the set of clusters outputted by the clustering algorithm, the matching rate of S with respect to a protein family F in the InterPro (Apweiler *et al.*, 2000) is defined as follows:

$$M(F,S) = \max_{C_i \in S} \frac{|C_i \cap (F \cap D)|}{|C_i \cup (F \cap D)|}, \quad (1)$$

where C_i is a cluster in S, D is the set of proteins on which clustering is conducted and $|D|$ denotes the number of proteins in D. Accordingly, the weighted average matching rate of S is defined as follows:

$$\bar{M}(S) = \frac{1}{\sum_{F_i} |F_i \cap D|} \sum_{F_i} \big[|F_i \cap D| \cdot M(F_i, S)\big], \quad (2)$$

where $F_1, F_2, F_3, \ldots, F_i$ are protein families in the InterPro. In this paper, evaluation of clustering quality is carried out by comparing the dendrograms generated by the proposed incremental clustering algorithm with those generated by the single-link algorithm.

Figure 4 shows the weighted average matching rates delivered by these two algorithms with the four benchmark datasets. In Figure 4, no cutoff threshold is imposed to flatten the dendrograms generated by the single-link algorithm. If a cutoff were imposed, then the weighted average matching rates delivered by the single-link algorithm would turn lower. As Figure 4 reveals, the proposed incremental clustering algorithm and the single-link algorithm deliver comparable performance in terms of weighted average matching rate. In this experiment, the results of the proposed incremental clustering algorithm with the three smaller datasets, Mouse, Human and Rat, are the averages of five independent runs with random order of input sequence.

With respect to summarization quality, Table 3 lists the numbers of non-leaf nodes in the dendrograms generated by the proposed algorithm and by the single-link algorithm with the four benchmark datasets. Again, for the three smaller datasets, the results of the proposed incremental clustering algorithm are the averages of five independent runs with random order of input sequence. Table 4 lists the average depth that a user needs to traverse in each dendrogram in order to find a cluster that matches one family in InterPro best. Here, a cluster C_i in a dendrogram H is said to match a family F defined in InterPro best, if

$$\frac{|C_i \cap (F \cap D)|}{|C_i \cup (F \cap D)|} = \max_{C_j \in H} \frac{|C_j \cap (F \cap D)|}{|C_j \cup (F \cap D)|},$$

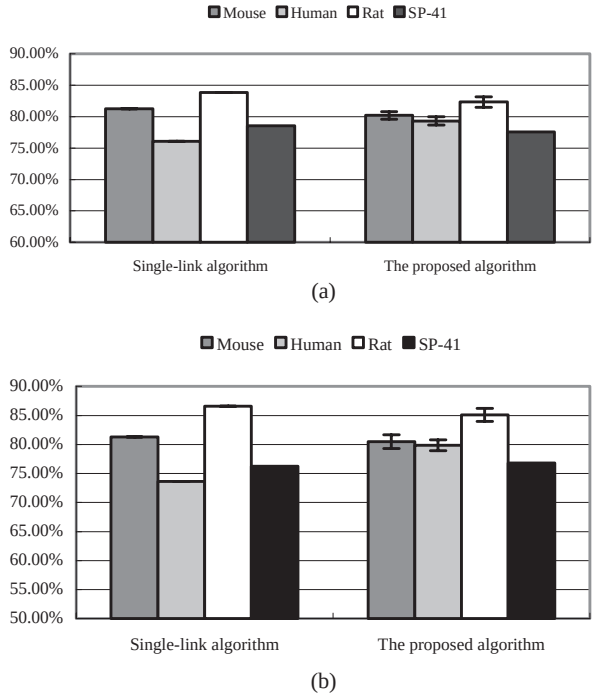

Fig. 4. Comparison of the weighted average matching rates delivered by the proposed incremental clustering algorithm and the single-link algorithm. (**a**) Comparison of the weighted average matching rate for protein families containing more than 10 proteins. (**b**) Comparison of the weighted average matching rate for protein families containing more than 30 proteins.

where C_j is a cluster in dendrogram H, and D denotes the set of proteins that H contains. Accordingly, given a dendrogram H, the numbers listed in Table 4 are computed as follows:

$$\frac{\sum_{j=1}^{f} Depth[\Psi(F_j, H)]}{f},$$

where

(1) $\Psi(F_j, H)$ denotes the cluster in H that matches F_j best;
(2) Depth of cluster $\Psi(F_j, H)$ in a dendrogram H is the number of edges on the path from $\Psi(F_j, H)$ to the root of the dendrogram;
(3) $F_1, F_2, \ldots, F_f$, are families defined in InterPro that contains proteins in H.

As shown in Table 3, the dendrograms generated by the proposed incremental clustering algorithm contain much fewer non-leaf nodes than the corresponding dendrograms generated by the single-link algorithm. Furthermore, as Table 4 shows, the clusters in the dendrograms generated by the single-link algorithm that best match the protein families in the InterPro are deeply embedded in the dendrogram. On the other hand, in the dendrograms generated by the proposed incremental clustering algorithm, most of those clusters that best match

Table 3. Comparison of the numbers of non-leaf nodes in the dendrograms generated by the proposed algorithm and the single-link algorithm

Dataset	Number of non-leaf nodes	
	Single-link	The proposed algorithm
Mouse	4707	1333.8 ± 6.14
Human	7470	1986.8 ± 12.28
Rat	2915	806 ± 6.54
SP-41	122 563	25 479

Table 4. Comparison of the average depths that a user needs to traverse in the dendrograms in order to find a cluster that matches one family in InterPro best

Dataset	Depth	
	Single-link	The proposed algorithm
Mouse	798.33	1.85 ± 0.037
Human	1414.24	2.54 ± 0.050
Rat	520.67	1.78 ± 0.029
SP-41	15312.95	4.48

the protein families in the InterPro are located no more than five levels down from the root. What Tables 3 and 4 combined imply is that, due to the summarization mechanism incorporated, the user can find protein clusters with biological meaning much more easily in the dendrograms generated by the proposed incremental clustering algorithm than in the dendrograms generated by the single-link algorithm. That is, the proposed incremental clustering algorithm offers the users highly concise dendrograms for analysis of protein clusters with biological significance.

In the discussion above, we have reported the overall clustering and summarization quality of the incremental clustering algorithm proposed in this paper. In the following, we will present more in-depth analyses. One issue that we have examined is the purities of the homogeneous clusters defined as follows:

$$purity(C) = \max_{i=1,\ldots,f} \left\{ \frac{|F_i \cap C|}{\left| \bigcup_{k=1,\ldots,f} (F_k \cap C) \right|} \right\},$$

where $F_1, F_2, \ldots, F_f$, are families defined in InterPro that contains proteins in cluster C.

The purity of the homogeneous cluster can be regarded as an inverse index of how likely proteins from different families are mixed in a homogeneous cluster. In the dendrogram generated by the proposed incremental clustering algorithm with the SP-41 dataset, there are 9017 homogeneous clusters with two or more proteins and with family identification in InterPro. Figure 5 shows a histogram of these 9017 homogeneous clusters. Among these 9017 homogeneous clusters,

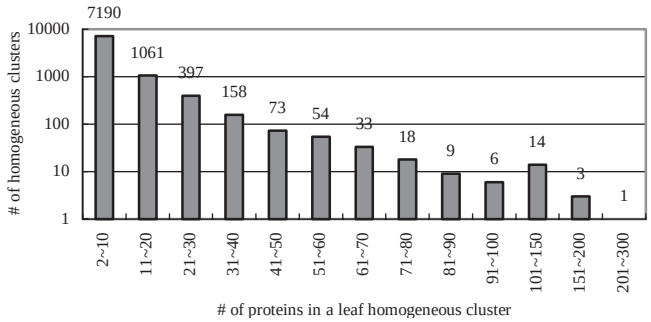

Fig. 5. The statistics for the homogeneous clusters with two or more proteins and with family identification.

there are 8653 clusters with purity equal to 1 and the weighted average purity of these 9017 homogeneous clusters is 98.50%.

Another issue that we have examined is how consistent the proposed incremental clustering algorithm is. We have compared how the human proteins in the Human dataset are clustered in the dendrograms generated with the Human dataset and with the SP-41 dataset. The analysis is based on the modified matching rate defined in the following:

$$M'(F,S) = \max_{C_i \in S} \frac{|P \cap C_i \cap (F \cap D)|}{|P \cap [C_i \cup (F \cap D)]|},$$

where C_i is a cluster in dendrogram S, D is the set of proteins on which clustering is conducted and P is the Human dataset set. Among all the families that contain the proteins in the Human dataset, 71.08% have exactly identical modified matching rates with the two dendrograms, the one generated with the SP-41 dataset and the one generated with the Human dataset. In addition, 20.5% have a higher matching rate with the dendrogram generated with the SP-41 dataset, and 8.42% have a higher matching rate with the dendrogram generated with the Human dataset. Overall, the experimental results reveal that the proposed incremental clustering algorithm performs quite consistently with datasets of different sizes and distributions.

Figure 6 presents a subtree of the dendrogram generated with the Human dataset to demonstrate the effects achieved with the proposed incremental clustering algorithm. In this example, the Glutathione *S*-transferase proteins with an identical subunit are clustered in the same leaf homogeneous cluster and each of these homogeneous clusters corresponds to a family defined in InterPro. The only exception is the leaf homogeneous cluster that contains Glutathione *S*-transferase theta 1 and Glutathione *S*-transferase theta 2, which do not belong to any family in the InterPro. In this subdendrogram, all proteins except the four in family IPR002946 contain domains 'Glutathione *S*-transferase, C-terminal' and 'Glutathione *S*-transferase, N-terminal'. The four proteins in family IPR002946 are present in the subdendrogram, because they contain domain 'Glutathione *S*-transferase, C-terminal'.

3.2 Evaluation of execution time

As elaborated in Section 2.5, the time complexity of the proposed incremental clustering algorithm for generating a new dendrogram with k new proteins added into a protein database is $O(km^2 \log m)$, where m is the number of leaf homogeneous clusters in the dendrogram corresponding to the current version of the database. Therefore, it is important to analyze how m grows as the number of proteins in the database, denoted by n in the following discussion, increases. Figure 7 shows the results from running the proposed incremental clustering algorithm to cluster all the proteins in Swiss-Prot (Release 41.0, 2003, 122 564 proteins). It is observed that the relation between $\log m$ and $\log n$ is governed by a linear equation with slope equal to 0.865. In other words, we have $m = cn^{\beta}$ and $\beta = 0.865$. We also conducted four additional experiments to get more insight about the relation between β and the characteristics of the dataset. The general observation is that if the dataset contains a large number of highly similar protein sequences, then β tends to be smaller. Otherwise, β tends to be larger. In the experiments conducted to cluster the human, rat and mouse proteins, the observed β values are 0.948, 0.979 and 0.888, respectively, when n is sufficiently large. On the other hand, if clustering is conducted on a dataset that contains proteins from two species, then the observed β values is smaller, 0.830 in this case. The reason behind this observation is that there exists a large number of highly similar protein sequences in the human, mouse and rat datasets. As mentioned earlier, in no case, could m exceed n. Therefore, the upper bound of β is 1.

Figure 8 shows the actual execution time of the proposed incremental clustering algorithm in one benchmark case. In this experiment, 7000 human proteins in Swiss-Prot are incrementally added into the dataset that contains 4708 mouse proteins in Swiss-Prot. Each time 500 proteins are added and the proposed algorithm resorts to an abstraction generated by the previous run of the algorithm to carry out clustering incrementally. In fact, the execution time of the proposed incremental clustering algorithm is dependent on the implementation of the single-link algorithm. In other words, if the single-link algorithm can run faster due to a more advanced implementation, the proposed incremental algorithm also benefits from this. The execution time reported in Figure 8a does not include the time taken to compute the pairwise similarity scores among proteins. Figure 8b compares the number of pairwise similarity scores that need to be computed with the proposed incremental clustering algorithm in this experiment, in comparison with that need to be computed, if clustering is carried out without the summarization process. It is observed that the proposed incremental clustering algorithm reduces the number of sequence alignment operations that need to be carried out by ~70%.

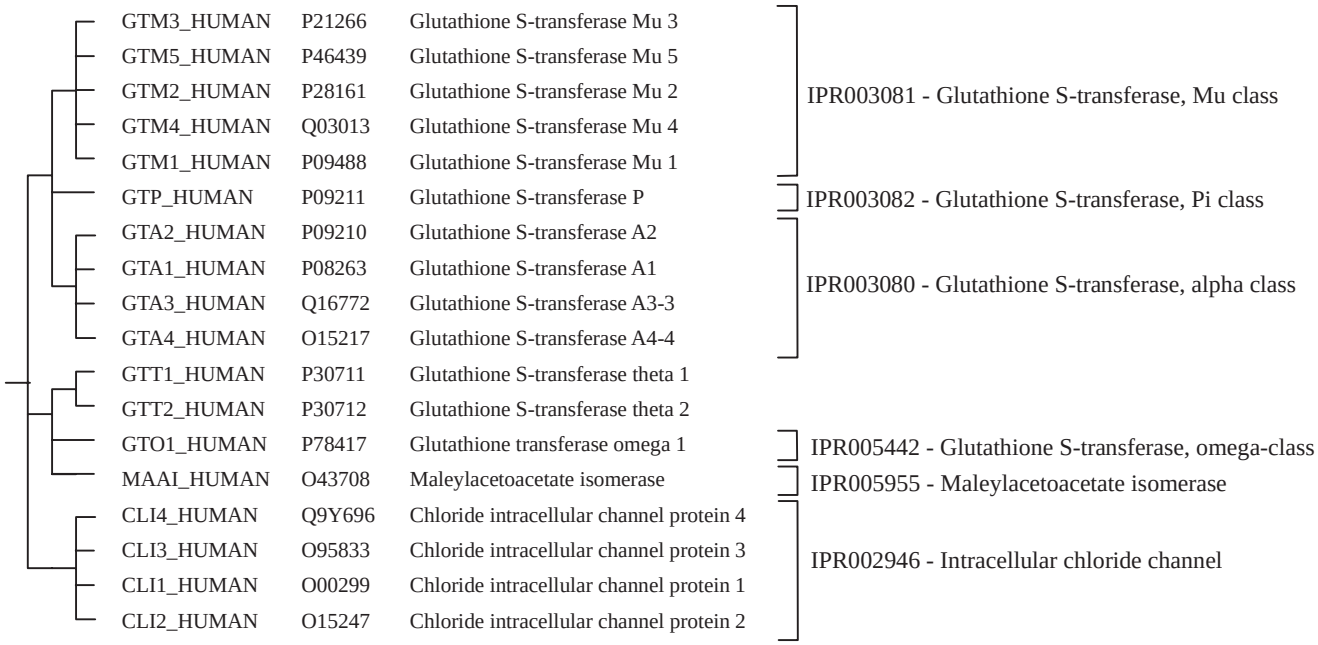

Fig. 6. An example that demonstrates the effects achieved with the proposed incremental clustering algorithm.

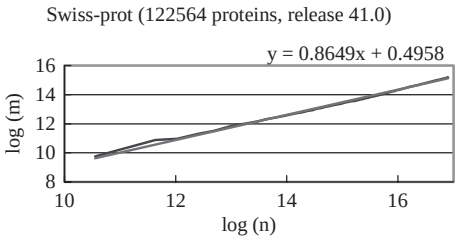

Fig. 7. Results from the experiments conducted to study the correlation between m (the number of leaf homogeneous clusters) and n (the number of proteins) on all the proteins in Swiss-Prot.

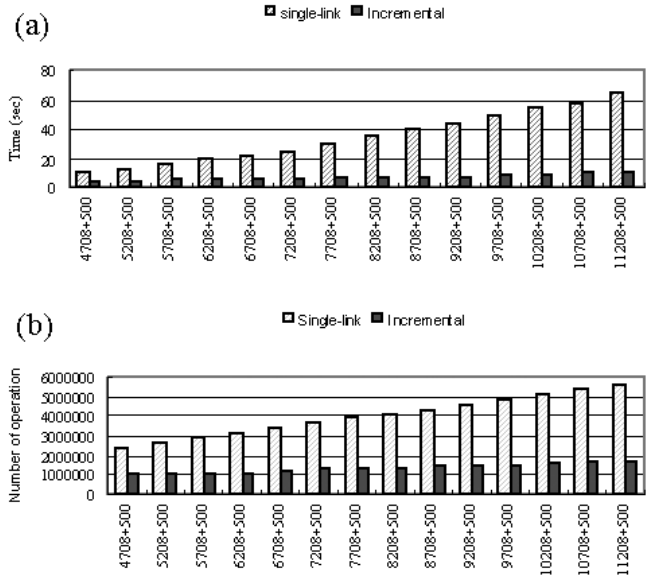

Fig. 8. Comparison of the execution times of the proposed incremental clustering algorithm and the single-link algorithm. (**a**) The execution times. (**b**) Number of sequence alignment operations executed.

4 CONCLUSION

This paper presents the design of a novel incremental hierarchical clustering algorithm aimed at generating high-quality dendrograms for analysis of protein databases. The proposed incremental clustering algorithm employs a statistics-based model to summarize the distributions of the similarity scores among the proteins in the database and to control formation of clusters. Experimental results reveal that, due to the summarization mechanism incorporated, the proposed incremental clustering algorithm offers the users highly concise dendrograms for analysis of protein clusters with biological significance.

Another distinction of the proposed algorithm is its incremental nature. This feature is essential for efficient handling of the contemporary protein databases, as the sizes of these databases continue to grow at fast rates. With incremental clustering, there is no need to carry out cluster analysis for a protein database starting from scratch, when the protein database is updated. Instead, the incremental clustering algorithm can refer to an abstraction generated by the previous run of the algorithm and carry out the analysis much more efficiently. The experiment conducted with the Swiss-Prot protein database shows that the time complexity of the

proposed incremental clustering algorithm for generating a new dendrogram with k new proteins added into a protein database containing n proteins is $O(n^{2\beta}\log n)$, where $\beta \sim 0.865$, provided that $k \ll n$.

As protein sequence clustering is so useful in analysis of protein functions and structures, continuous enhancements of clustering algorithms are essential for developing more effective approaches for discovery of new knowledge. Based on the results presented in this paper, there are several issues that deserve further investigation. The first issue concerns handling of multi-domain proteins. A multi-domain protein could belong to more than one protein families. Due to the nature of the hierarchical agglomerative clustering algorithm invoked in this paper, in the dendrogram generated by the proposed algorithm, a protein cannot be present in two sibling clusters. It is of interest to investigate whether approaches such as that proposed by GeneRAGE (Enright and Ouzounis, 2000) for detecting multi-domain proteins can be exploited to deal with this problem. Another possible improvement is the execution time of the proposed algorithm. As elaborated in Section 2.5, the time complexity of the proposed algorithm is a function of the number of leaf homogeneous clusters involved in the reconstruction process. Therefore, if the number of leaf homogeneous cluster involved is reduced, then the proposed algorithm can execute faster. One possibility in this regard is to detect outlier proteins and develop a different mechanism to process them, so that they will not be involved in the reconstruction process. Another issue that we have started to investigate is how to combine the statistical models based approach employed in this paper for controlling formation of clusters with other clustering algorithms.

ACKNOWLEDGEMENTS

We gratefully acknowledge financial support from the National Science Council of Taiwan grant NSC92–3112-B-027–001 and NSC92–2323-B-002–013.

REFERENCES

Abascal,F. and Valencia,A. (2002) Clustering of proximal sequence space for the identification of protein families. *Bioinformatics*, **18**, 908–921.

Altschul,S.F., Gish,W., Miller,W., Myers,E.W. and Lipman,D.J. (1990) Basic local alignment search tool. *J. Mol. Biol.*, **215**, 403–410.

Altschul,S.F., Madden,T.L., Schaffer,A.A., Zhang,J., Zhang,Z., Miller,W. and Lipman,D.J. (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 3389–3402.

Apweiler,R., Biswas,M., Fleischmann,W., Kanapin,A., Karavidopoulou,Y., Kersey,P., Kriventseva,E.V., Mittard,V., Mulder,N., Phan,I. and Zdobnov,E. (2001) Proteome Analysis Database: online application of InterPro and CluSTr for the functional classification of proteins in whole genomes. *Nucleic Acids Res.*, **29**, 44–48.

Apweiler,R., Attwood,T.K., Bairoch,A., Bateman,A., Birney,E., Biswas,M., Bucher,P., Cerutti,L., Corpet,F., Croning,M.D.R. *et al*. (2000) InterPro—an integrated documentation resource for protein families, domains and functional sites. *Bioinformatics*, **16**, 1145–1150.

Bairoch,A. and Apweiler,R. (2000) The Swiss-Prot protein sequence database and its supplement TrEmBL in 2000. *Nucleic Acids Res.*, **28**, 45–48.

Bolten,E., Schliep,A., Schneckener,S., Schomburg,D. and Schrader,R. (2001) Clusteirng protein sequences-structure prediction by transitive homology. *Bioinformatics*, **17**, 935–941.

Chen,C.-Y. (2003) Incremental hierarchical clustering algorithms based on statistical models. PhD Thesis, *National Taiwan University*, Taipei, Taiwan.

Dayhoff,M.O. (1976) The origin and evolution of protein superfamilies. *Fed. Proc.*, **35**, 2132–2138.

Enright,A.J., Dongen,S.V. and Ouzounis,C.A. (2002) An efficient algorithm for large-scale detection of protein families. *Nucleic Acids Res.*, **30**, 1575–1584.

Enright,A.J. and Ouzounis,C.A. (2000) GeneRAGE: A robust algorithm for sequence clustering and domain detection. *Bioinformatics*, **16**, 451–457.

Fisher,D. (1987) Improving inference through conceptual clustering. In *Proceedings of 6th National Conference on Artificial Intelligence (AAAI-87)*. Seattle, WA, pp. 461–465.

Han,J. and Kamber,M. (2000) *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers, New York.

Hegyi,H. and Gerstein,M. (1999) The relationship between protein structure and function: a comprehensive survey with application to the yeast genome. *J. Mol. Biol.*, **288**, 147–164.

Jain,A.K. and Dubes,R.C. (1988) *Algorithms for clustering data*. Prentice Hall, New Jersey.

Jobson,J.D. (1991) *Applied Multivariate Data Analysis*, Springer-Verlag, New Jersey.

Kawaji,H., Yamaguchi,Y., Matsuda,H. and Hashimoto,A. (2001) A graph-basd clustering method for a large set of sequences using a graph partitioning algorithm. *Genome Inform.*, **12**, 93–102.

Koonin,E.V., Tatusov,R.L. and Rudd,K.E. (1995) Sequence similarity analysis of *Escherichia coli* proteins—functional and evolutionary implications. *Proc. Natl Acad. Sci., USA*, **92**, 11921–11925.

Kriventseva,E.V., Biswas,M. and Apweiler,R. (2001a) Clustering and analysis of protein families. *Curr. Opin. Struct. Biol.*, **11**, 334–339.

Kriventseva,E.V., Fleischmann,W., Zdobnov,E.M. and Apweiler,R. (2001b) CluSTr: a database of clusters of Swiss-Prot+TrEMBL proteins. *Nucleic Acids Res.*, **29**, 33–36.

Lesk,A.M. (2002) *Introduction to Bioinformatics*. Oxford University Press, New York.

Li,W., Jaroszewski,L. and Godzik,A. (2001) Clustering of highly homologous sequences to reduce the size of large protein databases. *Bioinformatics*, **17**, 282–283.

Li,W., Jaroszewski,L. and Godzik,A. (2002) Sequence clustering strategies improve remote homology recognitions while reducing search times. *Protein Eng.*, **15**, 643–649.

Matsuda,H., Ishihara,T. and Hashimoto,A. (1996) A clustering method for molecular sequences based on pairwise similarity. *Genome Inform.*, **7**, 23–32.

Pearson,W.R. and Lipman,D.J. (1988) Improved tools for biological sequence comparison. *Proc. Natl Acad. Sci., USA*, **85**, 2444–2448.

Sasson,O., Linial,N. and Linial,M. (2002) The metric space of proteins: comparative study of clustering algorithms. *Bioinformatics*, **18** (Suppl. 1), s14–s21.

Sasson,O., Vaaknin,A., Fleischer,H., Portugaly,E., Bilu,Y., Linial,N. and Linial,M. (2003) ProtoNet: hierarchical classification of the protein space. *Nucleic Acids Res.*, **31**, 348–352.

Smith,T.F. and Waterman,M.S. (1981) Identification of common molecular subsequences. *J. Mol. Biol.*, **14**, 195–197.

Yona,G., Linial,N. and Linial,M. (1999) ProtoMap: automatic classification of protein sequences, a hierarchy of protein families, and local maps of the protein space. *Proteins*, **37**, 360–378.

Watanabe,H. and Otsuka,J. (1995) A comprehensive representation of extensive similarity linkage between large numbers of proteins, *Comput. Appl. Biosci.*, **11**, 159–166.

Wu,C.H., Huang,H., Arminski,L., Castro-Alvear,J., Chen,Y., Hu,Z.-Z., Ledley,R.S., Lewis,K.C., Mewes,H.-W., Orcutt,B.C. *et al*. (2002) The Protein Information Resource: an integrated public resource of functional annotation of proteins, *Nucleic Acids Res.*, **30**, 35–37.

Zhang,T., Ramakrishnan,R., Livny,M. (1996) BIRCH: an efficient data clustering method for very large databases. In *Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data (SIGMOD-96)*. Montreal, Canada, June 1996, pp. 103–114.