Graph-based clustering for finding distant relationships in a large set of protein sequences

Hideya Kawaji¹,², Yoichi Takenaka¹ and Hideo Matsuda¹,

¹Department of Bioinformatic Engineering, Graduate School of Information Science and Technology, Osaka University, 1-3 Machikaneyama, Toyonaka, Osaka 560-8531, Japan and ²NTT Software Corporation, 223-1 Yamashita-cho, Naka-ku, Yokohama, Kanagawa 231-8554, Japan

Received on May 26, 2003; revised on July 22, 2003; accepted on July 26, 2003

ABSTRACT
Motivation: Clustering of protein sequences is widely used for the functional characterization of proteins. However, it is still not easy to cluster distantly-related proteins, which have only regional similarity among their sequences. It is therefore necessary to develop an algorithm for clustering such distantly-related proteins.

Results: We have developed a time and space efficient clustering algorithm. It uses a graph representation where its vertices and edges denote proteins and their sequence similarities above a certain cutoff score, respectively. It repeatedly partitions the graph by removing edges that have small weights, which correspond to low sequence similarities. To find the appropriate partitions, we introduce a score combining the normalized cut and a locally minimal cut capacities. Our method is applied to the entire 40,703 human proteins in SWISS-PROT and TrEMBL. The resulting clusters show a 76% recall (20,529 proteins) of the 26,917 classified by InterPro. It also finds relationships not found by other clustering methods.

Availability: The complete result of our algorithm for all the human proteins in SWISS-PROT and TrEMBL, and other supplementary information are available at http://motif.ics.es.osaka-u.ac.jp/Ncut-KL/

Contact: matsuda@ist.osaka-u.ac.jp

INTRODUCTION
The number of molecular sequences in databases has been increasing exponentially with the progress of experimental technologies in molecular biology. The functional roles of many of the predicted proteins from their nucleotide sequences are still uncharacterized. It is one of the most important subjects in bioinformatics to provide some suggestions as to what the functional relationships between a number of poorly-characterized and well-characterized proteins are.

Several computational methods to measure sequence similarities between proteins have been proposed. For example, homology search algorithms, such as BLAST (Altschul et al., 1997) and FASTA (Pearson and Lipman, 1988), retrieve similar database sequences to a query from a database and calculate the statistical significance of their similarities. In most cases, proteins with a significant sequence similarity are related functionally. Such similarities are widely used to infer protein functions. However, retrieved proteins with apparent similarities are often uninformative, such as similarities to hypothetical proteins. Similarities between the queries and well-characterized proteins are often modest at best. It is not easy to predict its function from such modest similarities. To cope with this problem, it is important to incorporate alternative approaches, such as signature identification and sequence clustering (Kriventseva et al., 2001a,b).

Functionally or structurally-related proteins often have locally conserved regions, called functional sites, motifs or domains (hereafter referred to as signatures). Certain residues are well-conserved in all the regions corresponding to the same signature; the length of such a conserved region is, however, short compared to the length of the whole sequence. Proteins sharing the same signatures do not always exhibit apparent similarities between their sequences. Thus, we call them distantly-related proteins. In a protein family, an orthologous or paralogous protein group, the orders of signatures on protein sequences are conserved as well as the types of them. A group of distantly-related proteins is larger than a protein family.

A vast number of signature data has been accumulated in signature databases, such as PROSITE (Sigrist et al., 2002), Pfam (Bateman et al., 2002), ProDom (Servant et al., 2002), PRINTS (Attwood, 2002) and SMART (Letunic et al., 2002), to identify distant relationships. Some of them are integrated into InterPro (Mulder et al., 2002) that is widely used for protein functional characterization. These databases enable us to find proteins accurately that share signatures and their relative positions on the protein sequences; however, they still require some method for finding signatures. The increasing number of signatures stored in such databases

*To whom correspondence should be addressed.
indicates that there still remain many distant relationships that are as of yet unidentified.

Clustering protein sequences is another approach to find distant relationships. This method creates groups (or clusters) of possibly related proteins based on sequence similarities. Databases of such clusters are also constructed, such as CluSTr (Kriventseva et al., 2001a,b), SYSTERS (Krause et al., 2002), and ProtoMap (Yona et al., 2000). These databases are constructed by their own methods and classify proteins hierarchically.

The clustering approach enables us to use a transitive chain of similarities among many sequences. Distantly-related proteins may be grouped together into a cluster even though they do not show apparent similarities. It also can be performed without any pre-computation of the signatures, unlike methods which use the signature databases. Clustering proteins can find distant relationships without any pre-computation of the signatures even if the proteins do not show any apparent similarities with each other.

In this paper, we aim to find distant relationships, or to group distantly-related proteins, by clustering proteins. All proteins sharing the same signature should be classified into a group even if some of them share other more-conserved signatures. As the amount of sequence data has been increasing rapidly, it is necessary to design a sophisticated method with time and space efficiency.

Protein clustering methods

Several methods for clustering proteins have been proposed. The single linkage (SL) method is one of the simplest clustering methods and is widely used for the clustering of protein sequences (Koonin et al., 1995; Watanabe and Otsuka, 1995; Kriventseva et al., 2001a; Enright and Ouzounis, 2000; Krause et al., 2002). Although the method is simple, fast and widely used, it has difficulty in detecting an appropriate cutoff score for sequence similarity. Similarities between proteins sharing a signature vary greatly in terms of the length of the signature and the conservation of their residues. In addition, a protein is sometimes similar to another by chance even though they do not share any signatures.

Current protein-clustering applications use either a rigid cutoff score (Koonin et al., 1995; Enright and Ouzounis, 2000) or a set of cutoff scores with different levels (Kriventseva et al., 2001a). An SL-based clustering method, named SL-KL, classifies proteins with different levels that are selected automatically (Kawaji et al., 2001). It uses a graph representation of a given set of proteins, whose vertices denote proteins and edges denote pairs of proteins having sequence similarities above a certain cutoff score.

To find distant relationships, a $p$-quasi complete linkage algorithm (Matsuda et al., 1999) also uses the graph representation. It allows some proteins to overlap between two or more groups. This overlap is essential for clustering a protein set containing multi-domain structure proteins that have two or more signatures. However, its application to a large number of proteins is difficult due to its computational cost.

Two methods using the graph representation are proposed to identify protein families. TRIBE-MCL (Enright et al., 2002) adopts a Markov cluster (MCL) algorithm (Van Dongen, 2000). It may be applicable for finding distant relationships by changing a parameter affecting its cluster granularity, called inflation. The other method proposed by Abascal and Valencia (2002) adopts a normalized cut algorithm (Shi and Malik, 1997). The method differs from the other methods in that it focuses on proteins carefully retrieved by iterative homology searches to avoid spurious homologs.

Graph partitioning criteria

To find distantly-related proteins with clustering, we need to take following problems into consideration: (1) the degree of similarities between proteins sharing a signature varies greatly and (2) the numbers of proteins sharing a signature also vary greatly. For example, a signature of major histocompatibility complex (MHC) class I is shared by over 1500 human proteins although some minor signatures are shared only by a few human proteins. These indicate that sequence similarities and the size of a protein group do not provide any direct clues to clustering proteins. Thus, another criterion should be utilized. In addition, a protein set to be clustered contains multi-domain structure proteins. To resolve this problem completely, resulting clusters must overlap. But such clustering carries a high computational cost in such methods as the $p$-quasi complete linkage algorithm. Therefore, in order that our method is applicable to a large number of proteins, we focus on approximate distantly-related proteins without overlapping groups.

We use the graph representation of protein similarities to find distant relationships. An approach to clustering proteins is by iterative partitioning. In the remaining paragraphs, we consider a more appropriate criterion of graph partitioning for modeling distantly-related proteins. The key concept of graph partitioning is a ‘cut’, which is a set of edges between two distinct sets of nodes (or proteins). When the sum of edge weights in a cut, called capacity, is small, the two sets are considered as dissimilar.

Figure 1 shows an example of the graph representation of protein similarities, where all edges are weighted by one for simplicity. The proteins are intuitively grouped into the left proteins of P2 and the right ones. However, P1 is a partition with the lowest capacity of cut, called the minimum cut, due to the sparseness of the edges among the left proteins of P2. In graph partitioning with the minimum cut capacity, a protein group tends to be divided into small portion rather than to be discriminated from the other groups when similarities among its proteins are small. Although the criterion is adopted by the clustering of mRNA expression data (Sharan and Shamir, 2000), it is difficult for applications to find distant relationships in a protein set containing various kinds of signatures.
pairwise sequence similarities are pre-computed and that their graph representations are constructed before clustering.

**Cut and Ncut in a graph of proteins**

Protein sequences and their similarities are represented by a weighted graph $G = (V, E)$, where a node $v \in V$ represents a protein sequence and an edge $e(v_i, v_j) \in E$ exists only when its adjacent proteins $v_i, v_j \in V$ have a higher similarity than the specified cutoff score. The edge is weighted by the similarity $w(v_i, v_j)$. For a partition of the proteins $(A, V - A)$, the capacity of the cut is defined as:

$$\text{cut}(A, V - A) = \sum_{a \in A, \bar{a} \in V - A} w(a, \bar{a})$$

and Ncut is:

$$\text{Ncut}(A, V - A) = \frac{\text{cut}(A, V - A)}{\text{assoc}(A, V)} + \frac{\text{cut}(A, V - A)}{\text{assoc}(V - A, V)}$$

where $\text{assoc}(X, V) = \sum_{x \in X, v \in V} w(x, v)$. Note that the sum of $\text{assoc}(A, V)$ and $\text{assoc}(V - A, V)$ is invariable for the same graph. The balanced assoc values minimize the Ncut when cut$(A, V - A)$ is a constant. This means that Ncut is minimized when both the cut capacity is small and the sum of edge weights are balanced.

The graph is expressed as an adjacent matrix $W$, which is a $|V| \times |V|$ symmetrical matrix with $W(i, j) = w(v_i, v_j)$. The space of $W$ can be reduced to $O(|V| + |E|)$ by using a sparse matrix representation. Here, we let $D$ denote a $|V| \times |V|$ diagonal matrix with $d_{i,j} = \sum_{v_j \in V} w(v_i, v_j)$.

**Normalized cut partitioning**

Finding the cut with the smallest Ncut is known as a NP-hard problem, and its approximation algorithm was proposed by Shi and Malik (1997). Their algorithm uses the second smallest eigenvalue and its eigenvector for $D^{-1/2} (D - W) D^{-1/2} x = \lambda x$. A few extreme eigenvalues can be computed by an eigensolver, called the Lanczos method, in $O(m|V| + m|E|)$ time and $O(m|V| + |E|)$ space, where $m$ is a constant value of the Krylov subspace dimension (Golub and Van Loan, 1989; Chatelin, 1988). Components of the eigenvector are divided with an appropriate cutoff.

In order to apply the partitioning algorithm to the protein clustering problem, partitioning needs to be repeated until all clusters are obtained. Since the size and the number of clusters are not given in advance, we need to determine the appropriate conditions under which to stop partitioning. Abascal and Valencia (2002) introduced, in their clustering method of proximal proteins, the notions of connectivity as one of the stopping conditions; it is defined as:

$$\text{connectivity}(A, V - A) = \frac{|\{e(a, \bar{a}) | a \in A, \bar{a} \in V - A\}|}{|A| \times |V - A|}$$

Connectivity is a parameter similar to the completeness ratio $p$ in the $p$-quasi complete linkage algorithm (Matsuda *et al.*, 1984).
1999). In the $p$-quasi algorithm, $p$ is a ratio of the number of adjacent edges to the number of possible edges for each vertex in a cluster (the algorithm coincides with the complete linkage method if $p = 1$, and with SL if $p$ is virtually set to 0). Both the connectivity and the completeness ratio $p$ do not consider edge weights but consider only the number of edges. We adopt a constant value of connectivity as the stop condition, i.e. the recursion stops when the connectivity becomes higher than this constant value. This affects resulting cluster granularity as the cutoff in SL and the inflation in TRIBE-MCL do.

**Locally minimal cut partitioning**

As mentioned earlier, both partitions with the minimum cut capacity and the minimum normalized cut capacity are not necessarily optimal for dividing a set of proteins. We approach this problem by using a locally minimal cut criterion.

To select an appropriate partition from multiple ones with a locally minimal cut capacity in a graph, we use the KL heuristic, a heuristic approach proposed by Kernighan and Lin (1970). The algorithm starts from an initial partition and searches for a partition with a smaller and locally minimal cut capacity. Thus, the resulting partition depends on the initial one. The heuristic repeats one path improvement of the graph partitioning. In a path of improvement, each node is moved once at most. It is implemented with a procedure to mark or unmark nodes. Fiduccia and Mattheyses (1982) proposed a more efficient implementation (hereafter, referred to FM). The heuristic is adopted to improve SL clusters by SL-KL (Kawaji et al., 2001).

The KL heuristic can adjust the preference of a balanced partition of the normalized cut. Pothen et al. (1990) reported that a locally minimal cut partitioning algorithm succeeded in finding a cut with a smaller capacity than the initial partition that is similar to the normalized cut. The use of a partition with the minimum normalized cut capacity as an initial partition enables us to find the partition $P_2$ in Figure 1.

In the original formulation of KL and FM, the size of the resulting partition is given in advance. However, we cannot determine the size of a cluster when clustering proteins. SL-KL uses another implementation based on FM that determines the size automatically by selecting the first partition with a locally minimal cut while moving nodes (Kawaji et al., 2001).

Based on the implementation of SL-KL, we move nodes one by one so that the decrease in the cut capacity is maximized. We only allow nodes adjacent to edges in cut to be moved into the other set. When we have an initial partition $P_3$ (Fig. 1) that is a partition with the minimum normalized cut capacity, one of the nodes between $P_2$ and $P_3$ can be moved into the right of $P_3$, because of the smallest increase in the cut capacity in adjacent nodes to edges in the cut. A moved node is marked. An unmarked node is moved iteratively if the above condition permits, until it finds a partition with a locally minimal and smaller cut capacity like $P_2$. When such a partition is found, another partition is explored again with the same procedure after all the marks are cleared. As a result, we have a partition at $P_2$ because we cannot find any better partitions. Below, we use this implementation for the KL heuristic.

**Clustering algorithm**

We describe a clustering algorithm that uses both the normalized cut partitioning and the KL heuristic. It takes a graph representation of protein sequence similarities $G = (V, E)$ and a connectivity cutoff $t$ as inputs, and outputs a set of protein clusters.

\[
\text{Ncut-KL}((V, E, t))
\]

1. If $|V| \leq 3$, output $V$ as a cluster, and then return.
3. Compute a partition with a locally minimal and smaller capacity of the cut, $(A', V - A')$, by applying the KL heuristic to the result of (2).
4. If connectivity $(A', V - A') \geq t$, output $V$ as a cluster, and then return.
5. Ncut-KL($(A', E_A), t)$.
7. Exit.

The depth of recursion is $c$ in the worst case and is bounded by $O(c)$, where $c$ is the number of the resulting clusters. (2) takes $O(|V| + |E|)$ time and (3) takes a few iteration of $O(|E|)$ time, as mentioned above. Extra space is not needed for the recursion in (5) and (6). Thus, Ncut-KL takes $O(c|V| + c|E|)$ time and $O(|V| + |E|)$ space to compute all clusters, where $i$ is the maximum number of iterations in (3).

If (3) is deleted, the above algorithm repeats the normalized cut partitioning. Hereafter, we call this algorithm NCUT. Its computational cost is $O(c|V| + c|E|)$ due to the deletion of the cost of (3).

**Data and implementation**

In order to evaluate our algorithm, we use all the 40 703 human proteins in SWISS-PROT (Release 40.23) and TrEMBL (Release 21.2). A BLAST homology search is executed for each protein against all the proteins with an $e$-value threshold of 0.1 after filtering by SEG (Wootton and Federhen, 1996) and COILS (Lupas et al., 1991) with default parameters. An edge between two nodes exists only when the bit score between the proteins is more than 30. The edge is weighted by the bit score. As a result, 1 541 361 pairwise similarities are detected. The InterPro classifications are extracted from the cross-reference entries in SWISS-PROT and TrEMBL. We only use the signatures included by three or more proteins for evaluation. Thus, 26 917 proteins include 1 721 signatures. Note that there are many overlaps among the classifications, which is caused by family/subfamily relations of
the classifications and multi-domain proteins. Each protein has 1.76 signatures on average.

For comparison purposes, other clustering methods, NCUT, TRIBE-MCL, SL-KL and SL, are also applied to the above data. We implement Ncut-KL, NCUT, SL-KL and SL in the C language. In Ncut-KL and NCUT, meschach library (Version 1.2b; http://www.math.uiowa.edu/~dstewart/meschach/) is used for computing the eigenvalues and eigenvectors with the sparse matrix representation. In Ncut-KL and SL-KL, a bucket list is also implemented as a data structure. In TRIBE-MCL, the Amsterdam implementation of the MCL algorithm (Version 02-095; http://micans.org/mcl/) is used with default parameters except for inflation. All programs are executed on a Compaq AlphaServer ES45 (CPU: Alpha 21264 1 GHz).

Notation for evaluation

A set of clusters by a clustering method and a set of classifications by the InterPro signatures are denoted by \( \mathbf{C} \) and \( \mathbf{I} \), respectively. For two sets of proteins, \( c, i \) \((c \in \mathbf{C}, i \in \mathbf{I})\), the matching rate is defined as:

\[
\text{matching rate}(c, i) = \frac{|c \cap i|}{|c \cup i|}
\]

match\((c)\) is used to indicate an InterPro classification with the highest matching rate, and vice versa.

\[
\text{match}(c) = \arg \max_{i \in \mathbf{I}} \text{matching rate}(c,i)
\]

\[
\text{match}(i) = \arg \max_{c \in \mathbf{C}} \text{matching rate}(c,i)
\]

where argmax returns an argument that maximizes its following formula. double_match is a function that has a value of 1 only when both a cluster matches to an InterPro classification and the classification matches to the cluster. We call \( c \) as a double match cluster of \( i \), and \( i \) as a double match classification of \( c \) when double_match\((c, i)\) = 1.

\[
\text{double_match}(c, i) = \begin{cases} 
1 & \text{if match}(c) = i \text{ and } \text{match}(i) = c \\
0 & \text{else}
\end{cases}
\]

The following overlap\(^\Sigma\), precision and recall criteria are used as benchmarks to measure clustering quality:

\[
\text{overlap}^\Sigma(\mathbf{C}, \mathbf{I}) = \sum_{\text{double_match}(c,j) = 1} |c \cap i|
\]

\[
\text{precision}(\mathbf{C}, \mathbf{I}) = \frac{\text{overlap}^\Sigma(\mathbf{C}, \mathbf{I})}{\text{sequnum}(\mathbf{C})}
\]

\[
\text{recall}(\mathbf{C}, \mathbf{I}) = \frac{\text{overlap}^\Sigma(\mathbf{C}, \mathbf{I})}{\text{sequnum}(\mathbf{I})}
\]

where sequnum\((\mathbf{C})\) is the number of proteins in \( \mathbf{C} \) and \( \mathbf{C}^f \) refers to a set of double match clusters. Overlap\(^\Sigma\) is the number of appropriately classified proteins, and it is the main measure of the clustering quality. A clustering result contains less false positives when the precision is large, and it includes more true positives when the recall is large.

RESULTS

To evaluate the performance of the clustering methods, we use two data sets. To reveal the difference between Ncut-KL and NCUT, we apply them to a small set of human proteins containing two subfamilies of the eukaryotic protein kinases (IPR000719) and doublecortin (IPR003533). Next, we apply them and the other clustering methods, TRIBE-MCL, SL-KL and SL, for all the human proteins in SWISS-PROT and TrEMBL. Their performances are evaluated by comparing their results with the InterPro classification.

Protein kinase subfamilies and doublecortin

Receptor tyrosine kinase, class V (IPR001426) and MAP kinase (IPR003527) are subfamilies of the eukaryotic protein kinases. They are distantly related to each other because they share the same signature, the eukaryotic protein kinases. Although regions corresponding to the doublecortin are not similar to the kinase, one of the proteins sharing the signature also contains the kinase. The situation of the proteins is shown in Figure 2. The result of Ncut-KL is a partition with doublecortin-containing proteins and the others with the connectivity of 0.08, whereas NCUT divides proteins sharing the eukaryotic protein kinase signature into the subfamilies with the same connectivity. The KL heuristic succeeds in improving the NCUT clusters.

All human proteins in SWISS-PROT and TrEMBL

We use all the 40 703 human proteins in SWISS-PROT and TrEMBL in order to evaluate the performance of the clustering methods.

The granularity parameters of the clustering methods should be adopted appropriately. The number of double match clusters is small when all resulting clusters are large when compared to the InterPro classifications. On the other hand, the number of proteins contained in both the InterPro classification and its double match cluster is small when the clusters are small. We apply the clustering methods with some granularity parameter settings. The results are summarized in Table 1.

Overlap\(^\Sigma\) is maximized when the connectivity is 0.05, the cutoff is 100 and the inflation is 1.2. We use these settings hereinafter. Overlap\(^\Sigma\), recall and precision of Ncut-KL and NCUT are larger than those of TRIBE-MCL and SL with the granularities. Although the precision of SL-KL is similar to ones of Ncut-KL and NCUT, its recall is smaller than those of Ncut-KL and NCUT with all the granularities. The normalized cut based methods, Ncut-KL and NCUT, classify proteins more appropriately than the other methods.

The difference between Ncut-KL and NCUT is that the former has a larger overlap\(^\Sigma\) and the latter has a larger recall.
The signature of the eukaryotic protein kinases (IPR000719) represents a conserved catalytic core that is common in many enzymes containing serine/threonine and tyrosine protein kinases, which proteins play a central role in signal transduction and cellular regulation. Many of them have multi-domain structures like other proteins involved in signaling processes. Its double match cluster in SL contains 4070 proteins that are remarkably more than proteins that include the signature, although its double match clusters in Ncut-KL, NCUT and TRIBE-MCL contains 1018, 1191 and 997 proteins, respectively. This demonstrates the influence of the multi-domain proteins in SL, although the other methods succeed in avoiding it. The PDZ/DHR/GLGF domain (IPR001478)

<table>
<thead>
<tr>
<th>Method</th>
<th>Connectivity</th>
<th>Overlap</th>
<th>Recall (%)</th>
<th>Seqnum (Cd)</th>
<th>Precision (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ncut-KL</td>
<td>0.01</td>
<td>18,966</td>
<td>70%</td>
<td>28,231</td>
<td>67%</td>
</tr>
<tr>
<td>NCUT</td>
<td>0.01</td>
<td>18,177</td>
<td>68%</td>
<td>26,031</td>
<td>70%</td>
</tr>
<tr>
<td>TRIBE-MCL infla.</td>
<td>1.15</td>
<td>18,549</td>
<td>69%</td>
<td>25,151</td>
<td>74%</td>
</tr>
<tr>
<td>SL-KL Overlap</td>
<td>17,020</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SL-KL Recall</td>
<td>63%</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SL-KL Seqnum</td>
<td>21,502</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SL-KL Precision</td>
<td>79%</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SL cutoff</td>
<td>80</td>
<td>12,891</td>
<td>48%</td>
<td>21,584</td>
<td>60%</td>
</tr>
<tr>
<td>Overlap</td>
<td>12,891</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Recall</td>
<td>48%</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seqnum</td>
<td>19,241</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Precision</td>
<td>70%</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

SL-KL has no granularity parameters except for a threshold used in a graph representation of protein sequence similarities.

Table 1. Overlap, recall, and precision values with different granularity parameters for the five methods.
is also found in signaling proteins, and many of them have multi-domain structures. Its double match cluster in TRIBE-MCL contains 407 proteins that are remarkably larger than the InterPro classification. On the other hand, its double match clusters in Ncut-KL and NCUT contains 244 and 277 proteins and more overlaps with the classification than in TRIBE-MCL. With regard to the signature, the normalized cut-based methods, Ncut-KL and NCUT, avoid classifying other signatures into the same group.

Although the difference between the resulting clusters of Ncut-KL and NCUT in Table 2 is comparatively small, Ncut-KL tends to cover more proteins. It is consistent with the values in Table 1 that the recall of Ncut-KL is large and the precision is small, comparatively.

### Cluster and overlap size distribution

The size distributions of the clusters by the five methods and the classification by InterPro are shown in Figure 3. The distributions of the cluster sizes are moderately similar to that of InterPro, except for the SL-KL and SL in the size distribution of 80–100. SL-KL and SL produce less clusters than the other methods and InterPro.

The distributions of overlap sizes between the classifications and their double match clusters are shown in Figure 4. The
Fig. 3. The size distributions of the clusters by the five methods. The x- and y-axes represent the size of clusters (or classifications) and the number of clusters (or classifications) with the size, respectively.

Fig. 4. The size distributions of the overlaps between the InterPro classifications and their double match clusters. The x- and y-axes represent the size of the overlaps and the number of clusters with the overlap size, respectively.

distributions are moderately similar to the classification size in Figure 3, except for the SL in the size distribution of 80–100. These figures show that the clusters of Ncut-KL and NCUT show reasonable overlap not only in the clusters in Table 2, but also in the others.

**Computational resources**

Time and space used in computing clusters with the five methods are summarized in Table 3. Although Ncut-KL and NCUT are faster than SL-KL, they are slower than SL and TRIBE-MCL. An eigenvector computation for a large sparse matrix takes a large memory space. The normalized cut-based methods are applicable for a large set of proteins (e.g. all the proteins of an organism), although requiring comparatively large computational costs.

NCUT takes more time than Ncut-KL because the number of clusters increases, although some procedures for the KL heuristic are added to Ncut-KL. The number of Ncut-KL and NCUT clusters are 7056 and 8323, respectively.

**DISCUSSION**

We have proposed an algorithm for the clustering of a large set of proteins to find distantly-related groups. The algorithm uses a graph representation of pairwise sequence similarities. It adopts the normalized cut partitioning and a locally minimal cut partitioning based on the KL heuristic. It also adopts a constant value of connectivity as the stopping condition of the partitioning recursion.

The algorithm was evaluated by applying it on all the 40,703 human proteins in SWISS-PROT and TrEMBL. The resulting clusters show a 76% recall (20,529 proteins) of 26,917 classified by InterPro, and a 78% precision. This constitutes the best performance in the five compared clustering methods, including NCUT that divides an input graph iteratively by the normalized cut partitioning. In consideration of possible relationships not currently identified by InterPro, the value of precision is relatively reasonable.

With regard to the large classifications by InterPro, Ncut-KL and NCUT can find more relationships than the other methods, avoiding the classification of proteins with different signatures into the same group. In addition, the other clusters also reasonably overlap the InterPro classifications.

Ncut-KL and NCUT divide a set of proteins recursively while the connectivity is smaller than the threshold. Their results indicate that the partitioning criteria are appropriate and that the stopping condition of recursion avoids dividing clusters that are too small. Ncut-KL attempts to find more relationships than NCUT by using the KL heuristic based
on a locally minimal cut criterion. More overlaps with the InterPro classifications indicate that our partitioning criterion, the combination of the normalized cut partitioning and the KL heuristic, is more appropriate than that of NCUT.

An analysis of the double match clusters to MHC class I also shows that Ncut-KL can find distant relationships not identified by InterPro and the other methods. It also underlines that the importance of protein clustering even for the known signatures.

Taking computational costs into consideration, Ncut-KL is proved to be applicable to a large set of proteins, such as all the human proteins, although it is less scalable than TRIBE-MCL and SL. Thus, it is also applicable, even with current computer performance, for all the proteins of another organism or of two or more organisms with relatively small genomes.

Abascal and Valencia (2002) also employed the normalized cut criterion to identify protein families. A major difference between their method and NCUT lies in the target for clustering. The former applies the normalized cut partitioning to a protein set carefully retrieved by iterative homology searches. The latter applies it to a large set of proteins not carefully limited. In this paper, it is shown that the resulting clusters of a clustering method using the normalized cut work well for a large protein set that includes proteins with different kinds of signatures and multi-domain structure proteins.

It is concluded that the normalized cut criterion is more appropriate for modeling distantly-related protein groups, and that some improvement of NCUT based on the graph representation is possible. Proteins (24%) including any InterPro signatures are not classified appropriately by Ncut-KL. This may be due to the limitation of the homology search algorithm used in the construction of the graph, or instead, may indicate that further improvement in graph partitioning is possible. The analyses and further improvement of the clustering method could constitute a future work. In addition, graph partitioning does not find all the relationships correctly for multi-domain proteins. A scalable clustering method according to such structures could be also an important future work.

ACKNOWLEDGEMENTS

This work was supported by Grant-in-Aid for Scientific Research on Priority Areas ‘Genome Information Science’ from the Ministry of Education, Culture, Sports, Science and Technology of Japan.

REFERENCES


