A Clustering Method for Molecular Sequences based on Pairwise Similarity

H. Matsuda T. Ishihara A. Hashimoto
{matsuda, tatuya-i, hasimoto}@ics.es.osaka-u.ac.jp

Department of Informatics and Mathematical Science,
Graduate School of Engineering Science,
Osaka University
1-3 Machikaneyama, Toyonaka, Osaka 560 Japan

Abstract

This paper presents a method for clustering a large and mixed set of uncharacterized sequences provided by genome projects. As the measure of the clustering, we use a fast approximation of sequence similarity (FASTA score). However, in the case to detect similarity between two sequences that are much diverged in evolutionary process, FASTA sometimes underestimates the similarity compared to the rigorous Smith-Waterman algorithm. Also the distance derived from the similarity score may not be metric since the triangle inequality may not hold when the sequences have multi-domain structure. To cope with these problems, we introduce a new graph structure called p-quasi complete graph *for describing a cluster of sequences with a confidence measure. We prove that a restricted version of the p-quasi complete graph problem (given a positive integer k, whether a graph contains a 0.5-quasi complete subgraph of which size* $\geq k$ *or not) is NP-complete. Thus we present the outline of an approximation algorithm for clustering a set of sequences into subsets corresponding to p-quasi complete graphs. The effectiveness of our method is demonstrated by the result of clustering Escherichia coli protein sequences by our method.*

1 Introduction

As the result of genome projects on several organisms, a large number of molecular sequences have been available to scientific research community. Especially in some organisms, their complete nucleotide sequences are available and the protein sequences encoded in their genomes have been analyzed. It is recognized as one of the important issues to predict the functions of uncharacterized protein sequences.

For this purpose, various sequence comparison methods [1, 2, 3, 4] have been devised to explore similarities among characterized and uncharacterized protein sequences. Also multiple sequence alignment methods (e.g., [5]) are often used to reveal multiple similarity relationships and the conserved regions among their sequences.

By these methods, however, it is rather difficult to examine diverged relationships among a large mixed set of uncharacterized sequences which genome projects provide at a time. For example, in a mixed set of sequences, even if the whole sequences are not similar to each other, there may exist regional similarities among the partial fragments of their sequences. It gives very complicated structure of sequence similarities. We will discuss about this in Section 2.2.

We propose a clustering method of a mixed set of sequences based on their pairwise similarities. We formulate this problem as a graph covering problem by connected subgraphs where vertices and edges of the graph denote sequences and similarity between sequences, respectively.

A similar approach to clustering sequences is proposed [6]. This method provides a bird's-eye view of similarity relationships between large numbers of proteins with the aid of single-linkage clustering and graphical/numerical representation; whereas, our method does not intend to do single-linkage clustering but explores groups of sequences tightly related to each other with sequence similarity.

2 Sequence Similarity

2.1 Approximation to similarity score

Several scoring systems on sequence similarity have been proposed. Two types of them have been widely used; one is based on the dynamic programming method [1, 2] and the other is based on the statistical significance [4]. We took the former (especially the score by the Smith-Waterman algorithm [2]) as the measure of sequence similarity since it is useful for revealing regional similarities among the partial fragments of sequences.

The computational cost of the Smith-Waterman algorithm is, however, rather high for computing every pair of given sequences. Thus we took the approximate scores computed by the FASTA program [3]. According to the document in the program package [7], the FASTA program is about 50-times faster than the SSEARCH program that is an implementation of the Smith-Waterman algorithm in the same program package.

Figure 1 shows the relationship between similarity scores computed by FASTA and SSEARCH for the same pairs of sequences. Where the sequence similarity is high (approximately greater than 100), FASTA provides a good approximation for the SSEARCH score. Otherwise, the FASTA score is sometimes apart from the SSEARCH score. Also in all cases, the FASTA score is not greater than the SSEARCH score. Thus, we consider the FASTA score is reliable only when it is higher than a threshold value (say, 100).

2.2 Distance from similarity score

Clustering methods generally utilize some kind of distance among data sets. According to a review [8], given a set of sequences, the distance between any two sequences i and j can be formulated using the similarity between i and j as follows:

$$d(i,j) = -\ln s_{norm}(i,j), \tag{1}$$

where $d(i,j)$, ln and $s_{norm}(i,j)$ denote the distance between i and j, the natural logarithm and the *normalized* distance between i and j such that $0 \leq s_{norm}(i,j) \leq 1$ and $s_{norm}(i,i) = 1$, respectively. From the score computed by FASTA (or SSEARCH), $s_{norm}(i,j)$ can be approximated as follows:

$$s_{norm}(i,j) \simeq \frac{s_{dp}(i,j)}{l(i,j) \cdot w} , \tag{2}$$

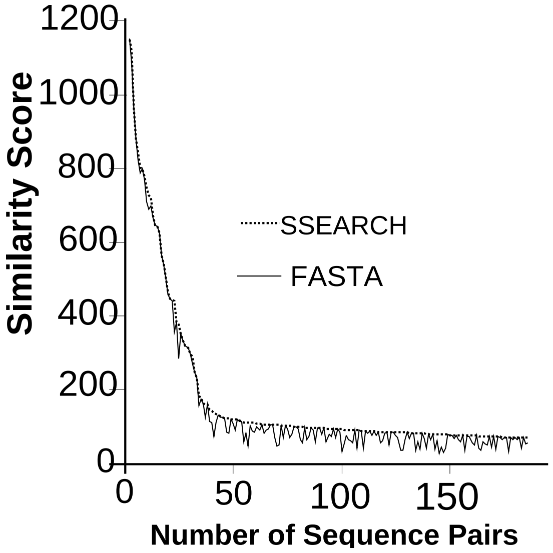


Figure 1: The relationship between similarity scores computed by FASTA and SSEARCH (Smith-Waterman score).

where $s_{dp}(i,j)$, $l(i,j)$ and w denote the similarity score of FASTA (or SSEARCH) between sequence i and j, the length of the regions aligned by the dynamic programming method and the score when two identical characters (DNA bases or amino acid residues) are matched with each other. The value of w depends on the distribution of characters in the alignment of i and j and the scoring matrix such as PAM, BLOSUM, etc.

Several clustering methods based on pairwise distances of data have been proposed. Most of these methods assume that the following two conditions hold on the distance metric.

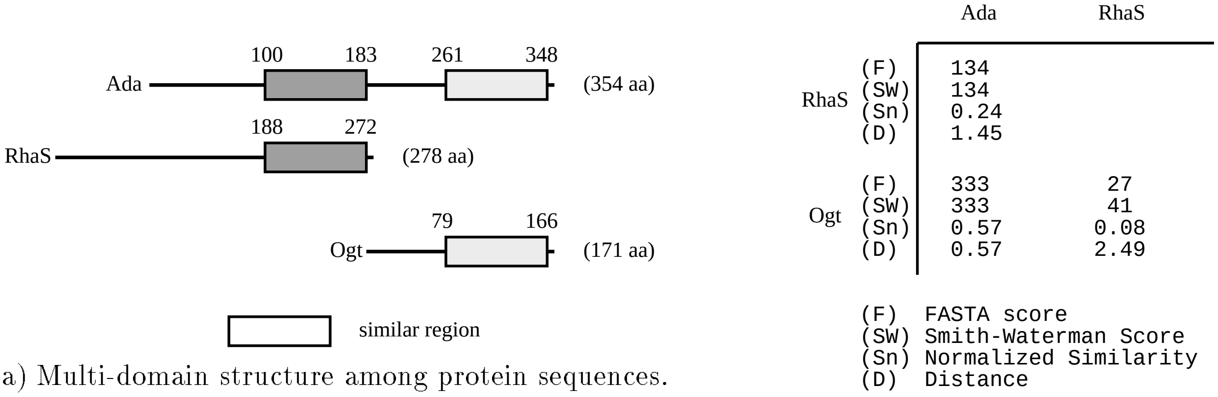
(1) $d(i,j) = d(j,i) \geq 0$, for all i, j ,
(2) $d(i,j) \leq d(i,k) + d(k,j) \geq 0$, for all i, j, k (triangle inequality).

Sequence similarity, however, forms multi-domain structure in some cases [6]. Figure 2 shows an example of multi-domain structure among protein sequences Ada, RhaS and Ogt in *Escherichia coli*. In this example, the above condition (1) holds but the condition (2) does not hold since $d(\mathrm{RhaS}, \mathrm{Ogt}) > d(\mathrm{RhaS}, \mathrm{Ada}) + d(\mathrm{Ada}, \mathrm{Ogt})$. According to SWISS-PROT Rel. 33.0 [10], Ada is a bifunctional protein whose N-terminal part functions as a transcription activator and C-terminal part functions as a methyltransferase. RhaS and Ogt function as a transcription activator and as a methyltransferase, respectively. The multi-domain structure among these proteins reflects the bifunctionality of Ada. In such cases, we cannot use the distance-based clustering methods, such as UPGMA (the unweighted pair-group method with arithmetic mean) [9]. The approach we used is described in Section 3.

3 Clustering Method

3.1 p-quasi complete graph

As described in Section 2, if we use a fast approximation of similarity score (such as FASTA score), every score is not always reliable especially when the similarity between sequences is



(a) Multi-domain structure among protein sequences.

		Ada	RhaS
RhaS	(F)	134	
	(SW)	134	
	(Sn)	0.24	
	(D)	1.45	
Ogt	(F)	333	27
	(SW)	333	41
	(Sn)	0.57	0.08
	(D)	0.57	2.49

```
(F)  FASTA score
(SW) Smith-Waterman Score
(Sn) Normalized Similarity
(D)  Distance
```

(b) Similarity scores and distances.

Figure 2: An example of multi-domain structure of protein sequences in *Escherichia coli* and the relationship between their similarity scores and distances.

low (typically less than 100). Also the distance derived from the score may not be metric since the triangle inequality may not hold.

To cope with these issues, we took an approach to graph covering method. The outline of our method is as follows:

(1) Protein sequences are regard as the vertices of a graph. Given approximate similarity scores among the sequences and a threshold value for the lowest reliable score, draw edges between any pair of vertices only if the score between them is higher than the threshold.

(2) Search for a subgraph composed of vertices connected to each other in a ratio of at least a given value. Repeat to find such subgraphs so that the whole graph is covered by these subgraphs. The goal of this method is to find the minimum cover by a set of the subgraphs.

To formulate the above, we introduce a new graph structure called *p-quasi complete graph.*

Definition 1 p-quasi complete graph $G = (V, E)$ *is a graph such that* $deg(v) \geq \lceil p(|V| - 1) \rceil$, *for all* $v \in V$, *where* $deg(v)$, p *and* $|V|$ *denote the degree of a vertex* v, *connectivity ratio of* G $(0.5 \leq p \leq 1)$ *and the number of all vertices in* V, *respectively.*

Clearly a 1-quasi complete graph (p-quasi complete graph such that $p = 1$) is a complete graph. Thus the connectivity ratio p means how close to a complete graph. Figure 3 shows an example of p-quasi complete graph such that $p = 0.5$ and $|V| = 8$.

On p-quasi complete graphs, the following theorem holds.

Theorem 1 *Every p-quasi complete graph is connected.*

Proof: Assume a p-quasi complete graph $G = (V, E)$ is not connected. Then there exists a partition so that it divides G into two subgraphs ($G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$) and no edge exists between G_1 and G_2.

Here we can assume $|V_1| \leq |V_2|$ without loss of generality. So $|V_1| \leq \lfloor |V|/2 \rfloor$ holds. Here, in general, the degree of any vertex $v \in V_1$ is $|V_1| - 1$ or less. Thus,

$$deg(v) \leq \lfloor |V|/2 \rfloor - 1, \tag{3}$$

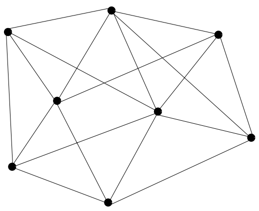


Figure 3: An example of p-quasi complete graph ($p = 0.5, |V| = 8$ and the degree of every vertex $\geq \lceil 0.5 \cdot (8-1) \rceil = 4$).

where $deg(v)$ denotes the degree of v.

From the definition of p-quasi complete graph,

$$deg(v) \geq \lceil 0.5 \cdot (|V| - 1) \rceil . \tag{4}$$

Here Eqs. (3) and (4) contradict each other. ■

A number of methods on graph decomposition based on the connetivity among vertices have been presented. For example, Kortsarz and Peleg concerns the dense subgraph problem, which is the problem of finding the densest subgraph of size k with maximum number of edges in a given graph [11]. On the other hand, our approach does not intend to find the densest subgraph but focuses on finding subgraphs that are denser than a given connectivity ratio.

3.2 NP-completeness on the p-quasi complete graph problem

In this section, we analyze the computational cost to find a p-quasi complete subgraph in a given graph. Before the analysis, we define some problems. The following discussion is based on the Garey and Johnson's book [12].

Definition 2 (RESTRICTED CLIQUE)
INSTANCE: *A graph $G = (V, E)$ and a positive integer k ($\lfloor |V|/2 \rfloor + 1 \leq k \leq |V|$).*
QUESTION: *Does G contain a clique of size k or more, that is, a subset $V' \subseteq V$ such that $|V'| \geq k$ and every two vertices in $|V'|$ are adjacent by an edge in E?*

The RESTRICTED CLIQUE problem is just a restricted version of the CLIQUE problem such that $k \geq \lfloor |V|/2 \rfloor + 1$. Since it is proved that the CLIQUE problem is NP-complete [13], it is easy to prove that the RESTRICTED CLIQUE problem is also NP-complete.

Definition 3 (RESTRICTED 0.5-QUASI COMPLETE GRAPH)
INSTANCE: *A graph $G = (V, E)$ and a positive integer k ($\lfloor |V|/2 \rfloor + 1 \leq k \leq |V|$).*
QUESTION: *Does G contain a 0.5-quasi complete subgraph of size k or more, that is, a subset $V' \subseteq V$ such that $|V'| \geq k$ and the degree of every vertex in V' is at least $\lceil 0.5 \cdot (|V'| - 1) \rceil$?*

The RESTRICTED 0.5-QUASI COMPLETE GRAPH is also a restricted version of a general problem such that $k \geq \lfloor |V|/2 \rfloor + 1$. The NP-completeness of the RESTRICTED 0.5-QUASI COMPLETE GRAPH problem can be proved by transforming this problem to the RESTRICTED CLIQUE problem.

Theorem 2 *The RESTRICTED 0.5-QUASI COMPLETE GRAPH problem is NP-complete.*

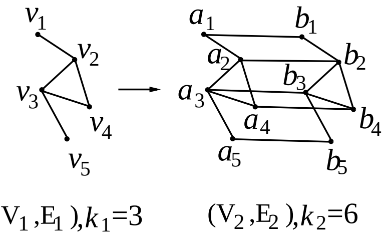


Figure 4: An example of transformation from an instance of the RESTRICTED CLIQUE problem to the corresponding instance of the RESTRICTED 0.5-QUASI COMPLETE GRAPH problem.

Proof: Given a subset $V' \subseteq V$, it can be decided within polynomial time whether V' satisfies the condition described in Definition 3. Thus the RESTRICTED 0.5-QUASI COMPLETE GRAPH problem belongs to NP.

We transform the RESTRICTED CLIQUE problem to the RESTRICTED 0.5-QUASI COMPLETE GRAPH problem. Let $G_1 = (V_1, E_1)$ and k_1 be any instance of the RESTRICTED CLIQUE problem. By the following transformation, the instance of the RESTRICTED CLIQUE problem can be transformed to an instance (a graph $G_2 = (V_2, E_2)$ and a positive integer k_2) of the RESTRICTED 0.5-QUASI COMPLETE GRAPH problem (see Figure 4).

Transformation 1

i) Double every vertex $v_i \in V_1$ to two vertices (say, $a_i, b_i \in V_2$). Let A and B be a set of a_i and a set of b_i, respectively. Here $|A| = |B| = |V_1|$ and $|V_2| = |A| + |B| = 2|V_1|$.

ii) Let $e(v_i, v_j)$ denote an edge $\in E_1$ between two vertices $v_i, v_j \in V_1$. Double every edge $e(v_i, v_j) \in E_1$ to two edges $e(a_i, a_j), e(b_i, b_j) \in E_2$. Also construct edges $e(a_i, b_i)$ $(1 \leq i \leq |V_2|/2)$.

iii) Set k_2 is $2k_1$.

This transformation can be carried out in polynomial time.

Then we shall prove that the RESTRICTED CLIQUE problem has a yes-instance if and only if the transformed RESTRICTED 0.5-QUASI COMPLETE GRAPH problem has a yes-instance.

Case 1: An instance of the RESTRICTED CLIQUE problem (a graph (V_1, E_1) and a positive integer k_1) is a yes-instance.

In this case, there exists a clique (V_1', E_1') of size $\geq k_1$ (for example, $V_1' = \{v_2, v_3, v_4\}$ and $E_1' = \{e(v_2, v_3),\ e(v_3, v_4),\ e(v_2, v_4)\}$ in Figure 4).

According to Transformation 1, we transform (V_1, E_1) to (V_2, E_2) and k_1 to k_2. By this transformation, we obtain a subgraph $(V_2', E_2') \subseteq (V_2, E_2)$ corresponding to (V_1', E_1') (e.g. $V_2' = \{a_2, a_3, a_4, b_2, b_3, b_4\}$ and $E_2' = \{e(a_2, a_3),\ e(a_3, a_4),\ e(a_2, a_4),\ e(b_2, b_3),\ e(b_3, b_4),\ e(b_2, b_4),\ e(a_2, b_2),\ e(a_3, b_3),\ e(a_4, b_4)\}$ in Figure 4).

Since the degree of any vertex $v \in V_2'$ is always the degree of any vertex in clique V_1' plus 1 (for edge $e(a_i, b_i)$),

$$deg(v) = (|V_1'| - 1) + 1 = |V_1'| = \frac{|V_2'|}{2} \geq \lceil 0.5 \cdot (|V_2'| - 1) \rceil .$$

Also $|V_2'| = 2|V_1'| \geq 2k_1 = k_2$. Thus the subgraph (V_2', E_2') is a 0.5-quasi complete graph of size $\geq k_2$. Hence the transformed RESTRICTED 0.5-QUASI COMPLETE GRAPH problem has a yes-instance.

Case 2: An instance of the RESTRICTED 0.5-QUASI COMPLETE GRAPH problem (a graph (V_2, E_2) and a positive integer k_2), which is transformed from an instance of the RESTRICTED CLIQUE problem, is a yes-instance.

In this case, there exists a 0.5-quasi complete subgraph (V_2', E_2') of size $\geq k_2$ $(\lfloor |V_2|/2 \rfloor + 1 \leq k_2 \leq |V_2|)$. By Transformation 1, $|V_2|$ can be divided into two subsets A and B such that $V_2 = A \cup B$, $A \cap B = \phi$ and $|A| = |B|$ where ϕ denotes an empty set. Thus $|A| = |B| = |V_2|/2$. Since $|V_2'| \geq \lfloor |V_2|/2 \rfloor + 1 = |A| + 1 = |B| + 1$, V_2' has at least one vertex from both A and B. Thus V_2' can be formulated as follows:

$$V_2' \;=\; A' \cup B' \text{ such that } A' \subseteq A, |A'| \geq 1, B' \subseteq B, |B'| \geq 1, |A'| + |B'| = |V_2'|.$$

Since (V_2', E_2') is a 0.5-quasi complete graph, for every vertex $a_i \in A'$,

$$deg(a_i) \geq \lceil 0.5 \cdot (|V_2'| - 1) \rceil = \lceil 0.5 \cdot (|A'| + |B'| - 1) \rceil \,. \tag{5}$$

Also for any vertex $a_i \in A'$, let $|E_{A'}(a_i)|$ be the number of edges between a_i and $a_j \in A'$ $(i \neq j)$ and $|E_{B'}(a_i)|$ be the number of edges between a_i and any vertex $\in B'$. Then the following relationship holds.

$$deg(a_i) = |E_{A'}(a_i)| + |E_{B'}(a_i)| \text{ such that } |E_{A'}(a_i)| \leq |A'| - 1, |E_{B'}(a_i)| \leq 1, \text{ (by Transformation 1)} \tag{6}$$

From Eq. (6),

$$deg(a_i) \leq |A'|. \tag{7}$$

If we assume $|A'| < |B'|$, $deg(a_i) > \lceil 0.5 \cdot (2|A'| - 1) \rceil = |A'|$. But this contradicts Eq. (7). Similarly $|A'| > |B'|$ cannot hold. Thus,

$$|A'| = |B'| = |V_2'|/2. \tag{8}$$

From Eqs. (5) and (8),

$$deg(a_i) \geq \lceil 0.5 \cdot (2|A'| - 1) \rceil = |A'|. \tag{9}$$

Eqs. (6), (7) and (9) conclude $|E_{A'}(a_i)| = |A'| - 1$, which means A' is a clique of which size is $|V_2'|/2 \geq k_2/2 = k_1$. Thus there exists a clique $V_1' (\subseteq V_1)$ of size $\geq k_1$ corresponding to A' by Transformation 1.

Thus the instance of the RESTRICTED CLIQUE problem, which is corresponding to the instance of the transformed RESTRICTED 0.5-QUASI COMPLETE GRAPH problem, is a yes-instance. ■

By Theorem 2, a restricted version of the 0.5-QUASI COMPLETE GRAPH problem is NP-complete. The 1-QUASI COMPLETE GRAPH problem is identical to the CLIQUE problem that is also NP-complete. Although it is not proved whether the problem in $0.5 < p < 1$ is NP-complete or not, we infer the problem is NP-complete.

If we assume that the general p-QUASI COMPLETE GRAPH problem is NP-complete, it is inferred that there does not exist a polynomial-time algorithm to solve the problem described in Section 3.1. Consequently, we need to develop some approximation algorithm for this problem.

3.3 An approximation algorithm

We developed an approximation algorithm in which we relax the two conditions on the original clustering problem in Section 3.1; (1) the solution is to be the *minimum* cover of clusters, and (2) each cluster is to be a *maximum* p-quasi complete graph.

Our algorithm is a kind of greedy algorithm that constructs clusters; (a) starting from the initial clusters so that each cluster has only one sequence and (b) growing up the size of each cluster by a stepwise addition of a sequence selected from outside of the cluster in the order of similarity scores until no additions yield a p-quasi complete graph.

Although it is guaranteed that the result of our algorithm is a set of clusters that are p-quasi complete graphs, each cluster is not always a maximum p-quasi complete graph since the sequence addition into a cluster is restricted to one-by-one. For example, for given sequence data described as (V_2, E_2) in Figure 4 and connectivity ratio $p = 0.5$, our algorithm can construct two clusters $\{a_2, a_3, a_4\}$ and $\{b_2, b_3, b_4\}$ but cannot combine them into $\{a_2, a_3, a_4, b_2, b_3, b_4\}$ since the addition of any one sequence to either of the two clusters does not yield any 0.5-quasi complete graph. The computational cost is $O(n^3 \log n)$ in average and $O(n^4)$ in the worst case where n denotes the number of sequences. The detail of this analysis is described elsewhere [14].

4 Preliminary Result

To evaluate the performance of our clustering method, we classified *Escherichia coli* protein sequences in the EcoProt7 library (available at `ftp://ncbi.nlm.nih.gov/repository/Eco/EcoProt/`). From the library, we extracted 1246 protein sequences that are classified into 299 clusters by Koonin, et al. [15]. As the method by Watanabe and Otsuka [6], Koonin, et al. use a single-linkage algorithm based on similarity scores. In their method, a cluster is defined as a group of protein sequences connected by BLASTP scores above 70.

On the other hand, we defined a cluster as a group of protein sequences described by a 0.5-quasi complete graph of which edges corresponding to FASTA scores (`opt` scores above 100 with `ktup=2`), and 370 clusters were obtained by our method. The computational time is about a total of 4410 seconds (3920 seconds for FASTA execution and 490 seconds for clustering) on Sun SPARCstation-20 (SuperSPARC-II, clock 75 MHz).

Although the number of clusters in our method is larger than 299 clusters done by Koonin, et al., these results become very similar by combining some overlapped clusters (some clusters that share the same sequences) to a cluster in our method.

Figure 5 shows a part of clusters on transcription regulation proteins that have helix-turn-helix DNA-binding domains extracted from the result of our method. Figure 5 clearly presents that a set of sequences that are similar to each other have the structure of the p-quasi complete graph. Also the relationship between Cluster 1 and 2 is corresponding to the multi-domain structure mentioned in Section 2.2.

In the result by Koonin et al., all the sequences in Cluster 1, 2 and 3 (except Ogt, RbsB and XylF) are classified into a helix-turn-helix domain cluster. Although Cluster 2 and 3 do not overlap with each other in the sense of p-quasi complete graph, we confirmed that Cluster 2 (including Ada) and Cluster 3 (except RbsB and XylF) are classified into two independent clusters by motif analysis using PROSITE Rel. 13.0 [16]; every sequence in the former cluster has the `HTH_ARAC_FAMILY_1` motif (ACC# PS00041), whereas every sequence in the latter cluster has the `HTH_LACI_FAMILY` motif (ACC# PS00356), and none of them have both motifs.

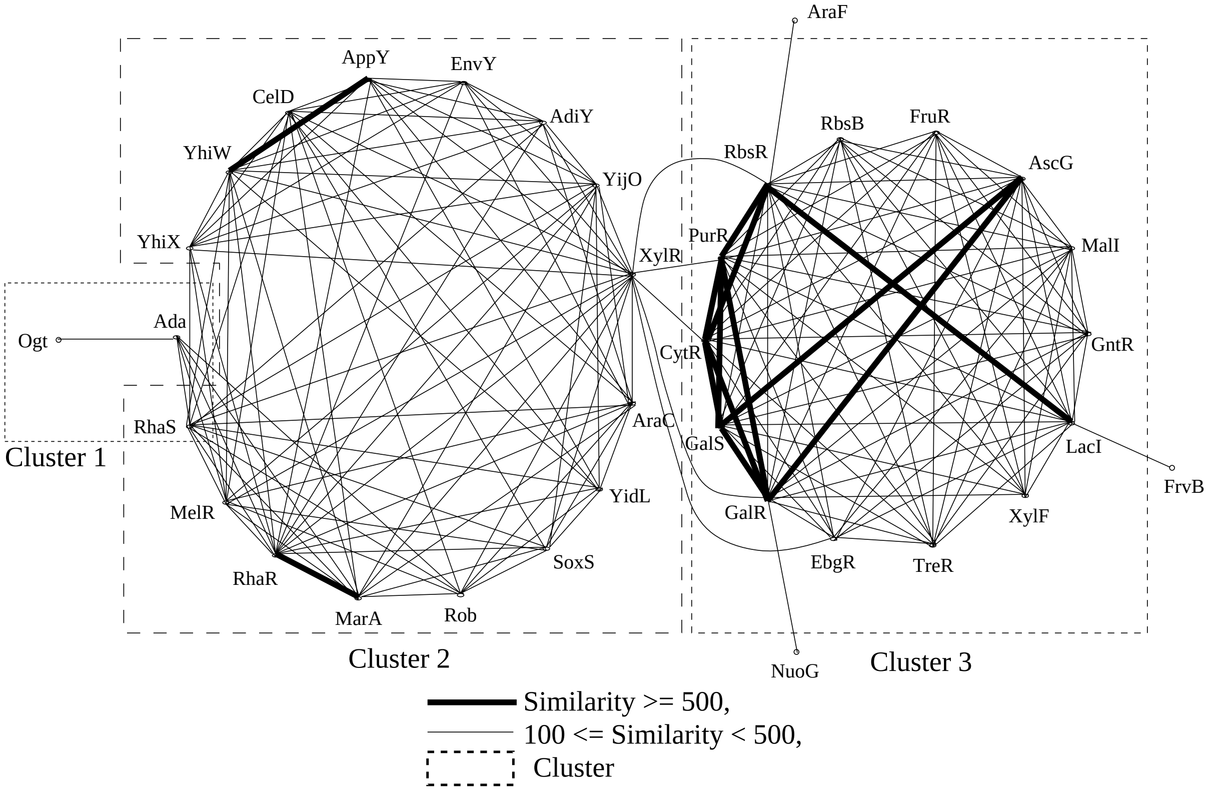


Figure 5: Clusters on transcription regulation proteins that have helix-turn-helix DNA-binding domains. Each cluster consists of a 0.5-quasi complete graph.

5 Summary and Conclusions

We have developed a method for clustering a large and mixed set of uncharacterized sequences provided by genome projects based on their pairwise similarities. On the use of sequence similarity, we indicated two problems; (1) when one uses sequences of which average similarity is relatively low, a fast approximation algorithm (e.g. FASTA) may underestimate their similarities, and (2) the distance derived from similarity may not be metric due to the multi-domain structure.

To cope with these problems, we introduce a new graph structure called p-quasi complete graph for describing a cluster of sequences with similarity scores above a chosen threshold. On the computational cost to find clusters from sequence data, we proved that a restricted version of the p-quasi complete graph problem ($p = 0.5$) is NP-complete. Thus we present a polynomial-time approximation algorithm.

By using 1246 *Escherichia coli* protein sequences, our method classified them into 370 clusters. Compared to the result done by a single-linkage algorithm [15], although the clusters constructed by our method included a few inappropriate sequences due to too low threshold, our method successfully detected more precise grouping that fits the result of motif analysis.

Acknowledgement

This work was supported in part by a Grant-in-Aid (08283103) for Scientific Research on Priority Areas from The Ministry of Education, Science, Sports and Culture in Japan.

References

[1] S. B. Needleman and C. D. Wunsch, “A General Method Applicable to the Search for Similarities in the Amino Acid Sequences of Two Proteins,” *J. Mol. Biol.*, Vol. 48, pp. 444–453, 1970.

[2] T. F. Smith and M. F. Waterman, “Identification of Common Molecular Subsequences,” *J. Mol. Biol.*, Vol. 147, pp. 195–197, 1981.

[3] W. R. Pearson and D. J. Lipman, “Improved Tools for Biological Sequence Comparison,” *Proc. Natl. Acad. Sci. USA*, Vol. 85, pp. 2444–2448, 1988.

[4] S. F. Altschul, W. Gish, W. Miller, E. W. Myers and D. J. Lipman, “A Basic Local Alignment Search Tool,” *J. Mol. Biol.* Vol. 215, pp. 403–410, 1990.

[5] J. D. Thompson, D. G. Higgins, and T. J. Gibson, “CLUSTAL W: Improving the Sensitivity of Progressive Multiple Sequence Alignment through Sequence Weighting, Position-Specific Gap Penalties and Weight Matrix Choice,” *Nucleic Acids Res.*, Vol. 22, pp. 4673–4680, 1994.

[6] H. Watanabe and J. Otsuka, “A Comprehensive Representation of Extensive Similarity Linkage between Large Numbers of Proteins,” *Comp. Appl. Bio. Sci.*, Vol. 11, No. 2, pp. 159–166, 1995.

[7] W. R. Pearson, The FASTA Program Package Version 2.0, Available at `ftp://ftp.virginia.edu/pub/fasta/`.

[8] D. L. Swofford and G. J. Olsen, “Phylogeny Reconstruction,” In *Molecular Systematics,* ed. D. M. Hillis and C. Moritz, Chap. 11, pp.411–501, Sinauer Associates, Sunderland, MA, 1990.

[9] R. R. Sokal and C. D. Michener, “A Statistical Method for Evaluating Systematic Relationships,” University of Kansas Sci. Bull. Vol.28, pp.1409–1438, 1958.

[10] A. Bairoch and R. Apweiler, “The SWISS-PROT Protein Sequence Data Bank and Its New Supplement TREMBL,” *Nucleic Acids Res.*, Vol. 24, No. 1, pp. 21–25, 1996.

[11] G. Kortsarz and D. Peleg, “On Choosing a Dense Subgraph,” *Proc. 34th IEEE Symp. Foundations of Comp. Sci.*, pp. 692–701, 1993.

[12] M. R. Garey and D. S. Johnson, “Computers and Intractability – A Guide to the Theory of NP-Completeness,” W. H. Freeman and Company, CA, 1979.

[13] R. M. Karp, “Reducibility among Combinatorial Problems,” In *Complexity of Computer Computations*, ed. R. E. Miller and J. W. Thatcher, Plenum Press, NY, pp. 85–103, 1972.

[14] T. Iwade, T. Ishihara, H. Matsuda and A. Hasimoto, “Classifying Method of Large Size Set of Sequences based on Pairwise Similarities,” *IPSJ SIG Report*, 96-AL-52, pp. 33–40, 1996 (in Japanese).

[15] E. V. Koonin, R. L. Tatusov and K. E. Rudd, “Sequence Similarity Analysis of *Escherichia coli* Proteins - Functional and Evolutionary Implications,” *Proc. Natl. Acad. Sci. USA*, Vol. 92, pp. 11921–11925, 1995.

[16] A. Bairoch, P. Bucher and K. Hofmann, “The PROSITE Database, Its Status in 1995,” *Nucleic Acids Res.*, Vol. 24, No. 1, pp. 189–196, 1996.