Clustering of Non-Alignable Protein Sequences

Abdellali Kelil
Department of Computer Sciences
University of Sherbrooke
Sherbrooke, QC, Canada
1 (819) 823 8616
Abdellali.Kelil@USherbrooke.ca

Shengrui Wang
Department of Computer Sciences
University of Sherbrooke
Sherbrooke, QC, Canada
1 (819) 821 8000 ext 62022
Shengrui.Wang@USherbrooke.ca

Ryszard Brzezinski
Department of Biology
University of Sherbrooke
Sherbrooke, QC, Canada
1 (819) 821 8000 ext 61077
Ryszard.Brzezinski@USherbrooke.ca

ABSTRACT
We are interested in the problem of grouping families of non-alignable protein sequences, such as circular-permutation, multi-domain and tandem-repeat proteins, into clusters (classes) of related biological functions. For such sequences, whose numbers are constantly growing, the commonly used alignment-dependent approaches fail to yield biologically plausible results. To the best of our knowledge, no automatic process yet exists to carry out clustering on these proteins. Biologists often use more complex manual approaches based on secondary and tertiary structures, which require considerably more resources and time.

In this paper, we develop a new similarity measure SMS, applied directly on non-aligned sequences. It allows us to develop a new and original alignment-free algorithm, named CLUSS, for clustering protein families based on a spectral decomposition approach inspired by the latent semantic analysis (LSA) widely used in information retrieval. CLUSS, utilized jointly with SMS, is effective on both alignable and non-alignable protein sequences. To show this, we have extensively tested our algorithm on different benchmark protein databases and families; we have also compared its performance with many alignment-dependent mainstream algorithms. The source code, the application server, and all experimental results are available at CLUSS web site http://prospectus.usherbrooke.ca/CLUSS/.

Categories and Subject Descriptors
J.3 [**Life and Medical Sciences**]: Biology and Genetics; I.5.3 [**Pattern Recognition**]: Clustering

General Terms
Algorithms, Measurement, Experimentation

Keywords
Clustering, Phylogenetic, Biological Function, Protein Sequences, Matching, Similarity Measure, Alignable, Non-Alignable

BIOKDD'07, August 12, 2007, San Jose, California, USA.

1. INTRODUCTION
With the rapid burgeoning of protein sequence data, the number of proteins for which no experimental data are available greatly exceeds the number of functionally characterized proteins. To predict a function for an uncharacterized protein, it is necessary not only to detect its similarities to proteins of known biochemical properties (i.e., to assign the unknown protein to a family), but also to adequately assess the differences in cases where similar proteins have different functions (i.e., to distinguish among subfamilies). One solution is to cluster each family into distinct subfamilies composed of functionally related proteins. Subfamilies resulting from clustering are easier to analyze experimentally. A subfamily member that attracts particular interest need to be compared only with the members of the same subfamily. A biological function can be attributed with high confidence to an uncharacterized protein, if a well-characterized protein within the same cluster is already known. Conversely, a biological function discovered for a newly characterized protein can be extended over all members of the same subfamily.

Almost all automatic clustering approaches deal with only aligned protein sequences, which are performed via alignment algorithms such as the widely known MUSCLE [8], ClustalW [36], MAFFT [18] and T-Coffee [26], and many others. These algorithms often provide information on both conserved and mutated motifs, making it a good approach for measuring similarities between protein sequences. However, they have several serious limitations, including the following:

- **Dependence on the algorithm used.** The results depend heavily on the algorithm selected and the parameters set by the user for the alignment algorithm (e.g., gap penalties). As far as easily-alignable proteins are concerned, almost every existing alignment algorithm can yield good results. However, for protein sequences that are difficult to align, each alignment algorithm finds its own solution. Such variable results create ambiguities and can complicate the clustering task [25].

- **Problem of non-alignable sequences.** For the case of non-alignable protein sequences (i.e., not yet definitively aligned), alignment-based algorithms do not succeed in producing biologically plausible results. This is due to the nature of the alignment approaches, which are based on the matching of subsequences in equivalent positions, while non-alignable proteins often have similar and conserved domains in non-equivalent positions [25], such as circular-permutation, multi-domain and tandem-repeat proteins

There are other known difficulties that limit the reliability of alignment, especially for the case of hard-to-align protein sequences, such as "*repeat*", "*substitution*" and "*gap*" problems, which are well discussed by Higgins [15].

The number of protein sequences that are hard-to-align or not alignable at all is rapidly increasing. These proteins are frequently related to important biological phenomena, and their classification is of primary importance for the comprehension of these phenomena. One example is the group of 33 (α/β)8-barrel proteins belonging to the Glycoside Hydrolase (GH) family [35], which has an important role in the physiology of the alive cell, as discussed in [5,13]. A large number of these are still uncharacterized, since to date the process has been carried out manually with complicated approaches, such as those employed by Côté *et al.* [5] and Fukamizo *et al.* [13] for the characterization of the 33 (α/β)8-barrel proteins belonging to the GH [35] family. Most of the tools currently available are based on the alignment of protein sequences, making them inappropriate for this kind of proteins.

Our aim in this paper is to develop a new approach to the biological interpretation of protein sequences, especially those which cause problems for alignment-dependent algorithms. Our work is an attempt to build an algorithm to help biologists perform analyses of certain kinds of protein sequences, which are now carried out almost manually. In the rest of the paper, we use the terms subfamily and cluster interchangeably.

2. RELATED WORK

The literature reports a number of algorithms for clustering protein databases, such as the widely used algorithm BLAST [1] and its improved versions Gaped-Blast and PSI-Blast [2], and SYSTERS [23], ProtClust [29] and ProtoMap [40] (see [32] for a review). These algorithms have been designed to deal with large sets of proteins by using various techniques to accelerate examination of the relationships between proteins. However, they are not very sensitive to the subtle differences among similar proteins. Consequently, these algorithms are not effective for clustering protein sequences in closely related families. On the other hand, more specific algorithms have also been developed, for instance, the widely cited algorithms BlastClust [3]**,** which uses score-based single-linkage clustering, TRIBE-MCL [10]**,** based on a Markov clustering approach, and gSPC [34]**,** based on a method that is analogous to the treatment of an inhomogeneous ferromagnet in physics. Almost all of these algorithms are either based on sequence alignment or rely on alignment-dependent algorithms for computing pair-wise similarities.

3. APPROACH OVERVIEW

In this paper, we propose an efficient and original algorithm, CLUSS, for clustering protein families based on a new alignment-free measure we propose for protein similarity. The novelty of CLUSS resides essentially in two features. First, CLUSS is applied directly to non-aligned sequences, thus eliminating the need for aligned sequences. Second, it adopts a new measure of similarity, directly exploiting the substitution matrices generally used to align protein sequences and showing a great sensitivity to the relations among similar and divergent protein sequences. CLUSS can be summarized as follows:

Given *F***,** a family containing a given number of proteins:

1. Build a pairwise similarity matrix for the proteins in *F* using SMS our new similarity measure.
2. Create a phylogenetic tree of the protein family *F* using our new clustering approach.
3. Assign a co-similarity value to each node of the tree.
4. Calculate a critical threshold for identifying subfamily branches, by computing the interclass inertia [7].
5. Collect each leaf from its subfamily branch into a distinct subfamily.

4. SMS: SIMILARITY MEASURE

Many approaches to measuring the similarity between protein sequences have been developed. Prominent among these are alignment-dependent approaches, including the well-known algorithm BLAST [1] and its improved versions Gaped-Blast and PSI-Blast [2], whose programs are available at [3], as well as several others such as the one introduced by Varré *et al.* [37] based on movements of segments, and the recent algorithm Scoredist introduced by Sonnhammer *et al*. [33] based on the logarithmic correction of observed divergence. These approaches often suffer from accuracy problems, especially for multi-domain proteins (in general case hard-to-align protein sequences). The similarity measures used in these approaches depend heavily on the alignability of the protein sequences. In many cases, alignment-free approaches can greatly improve protein comparison, especially for non-alignable protein sequences. These approaches have been reviewed in detail by several authors [30,31,9,38]**.** Their major drawback, in our opinion, is that they consider only the frequencies and lengths of similar regions within proteins and do not take into account the biological relationships that exist between amino acids. To correct this problem, some authors [9] have suggested the use of the Kimura correction method [22] or other types of correction, such as that of Felsenstein [12]. However, to obtain an acceptable phylogenetic tree, the approach described in [9] performs an iterative refinement including a profile-profile alignment at each iteration, which significantly increases its complexity.

To overcome these difficulties of alignment-based approaches, we have developed SMS a new approach inspired by biological considerations and known observations related to protein structure and evolution. The goal is to make efficient use of the information contained in amino acid subsequences in the proteins, which leads to a better similarity measurement. The principal idea of our approach is to use a substitution matrix such as BLOSUM62 [14] or PAM250 [6] to measure the similarity between matched amino acids from the protein sequences being compared.

4.1 Matching score

In this section, we will use the symbol |.| to express the length of a sequence. Let *X* and *Y* be two protein sequences belonging to the protein family *F*. Let *x* and *y* be two identical subsequences belonging respectively to *X* and *Y*; we use $\Gamma_{x,y}$ to represent the matched subsequence of *x* and *y*. We use *l* to represent the minimum length that $\Gamma_{x,y}$ should have (i.e., we will be interested only in $\Gamma_{x,y}$ whose length is at least *l* residues). We define E^{l}_{XY}, the key set of matched subsequences $\Gamma_{x,y}$ for the definition of our similarity function, as follows (see Figure 1 for an example):

$$E^{l}_{X,Y} = \left\{ \Gamma_{x,y} \;\middle|\; \begin{array}{l} |\Gamma_{x,y}| \geq l \;, \\ \forall \Gamma_{x',y'} \in E^{l}_{X,Y}, \Gamma_{x',y'} \neq \Gamma_{x,y} \Rightarrow (x' \not\subset x \vee y' \not\subset y) \end{array} \right\} \quad (1)$$

The expression ($x' \not\subset x$) means that x' is not included in x, either in terms of the composition of the subsequences or in terms of their respective positions in X. The matching set E^{l}_{XY} contains all the matched subsequences of maximal length between the sequences X and Y. It will be used to compute the matching score of the sequence pair.

The formula E^{l}_{XY} adequately describes some known properties of polypeptides and proteins. First, protein motifs (i.e., series of defined residues) determine the tendency of the primary structure to adopt a particular secondary structure, a property exploited by several secondary-structure prediction algorithms. Such motifs can be as short as four residues (for instance those found in β-turns), but the propensity to form an α-helix or a β-sheet is usually defined by longer motifs. Second, our proposal to take into account multiple (i.e., ≥2) occurrences of a particular motif reflects the fact that sequence duplication is one of the most powerful mechanisms of gene and protein evolution, and if a motif is found twice (or more) in a protein it is more probable that it was acquired by duplication of a segment from a common ancestor than by acquisition from a distant ancestor.

The construction of $E^{l}_{X,Y}$ requires a CPU time proportional to $|X|*|Y|$. In practice, however, several optimizations are possible in the implementation, using encoding techniques to speed up this process. In our implementation of SMS, we used a technique that improved considerably the speed of the algorithm; we can summarize it as follows:

By the property that all possible matched subsequences satisfy $|\Gamma x,y| \geq l$, we know that each $\Gamma x,y$ in $E^{l}_{X,Y}$ is an expansion of a matched subsequence of length l. Thus we first collect all the matched subsequences of length l, which takes linear time. Secondly, we expand each of the matched subsequences as much as possible on the both left and right sides. And finally, we select all the expanded matched sequences that are maximal according to the inclusion criterion. This technique is very efficient for reducing the execution time in practice. However, due to the variable lengths of the matched sequences, it may not be possible to reduce the worst-case complexity to a linear time. In the Results section, we provide a time comparison between our algorithm and several existing ones.

Figure 1. Matching subsequences

Figure 1 shows an example of $E^{l}_{X,Y}$ construction, with l=4. Let X and Y be two protein sequences, as illustrated. Among the matches shown in Figures 1.A and 1.B, the matched subsequence Γ_1 of X_1 and Y_1, will be added to the matching set $E^{4}_{X,Y}$. Similarly, for Γ_2 the match of X_1 and Y_2, and Γ_3 the match of X_2 and Y_1 will also be added to the matching set E^{4}_{XY}. On the other hand, since $X_2 \subset X_3$ and $Y_2 \subset Y_3$, Γ_4 the matched subsequence of X_2 and Y_2, will not be added to E^{4}_{XY}. Instead, Γ_5 the match of X_3 and Y_3, will be added to the set E^{4}_{XY}, even though X_3 overlaps with X_2.

Let M be a substitution matrix, and Γ a matched subsequence belonging to the matching set E^{l}_{XY}. We define a weight $W(\Gamma)$ for the matched subsequence Γ, to quantify its importance compared to all the other subsequences of E^{l}_{XY}, as follows:

$$W(\Gamma) = \sum_{i=1}^{|\Gamma|} M[\Gamma[i], \Gamma[i]] \quad (2)$$

where $\Gamma[i]$ is the i^{th} amino acid of the matched subsequence Γ, and $W[\Gamma[i],\Gamma[i]]$ is the substitution score of this amino acid with itself. Here, in order to make our measure biologically plausible, we use the substitution concept to emphasize the relation which binds one amino acid with itself. The value of $M[\Gamma[i],\Gamma[i]]$ (i.e., entries on the diagonal of the substitution matrix) estimates the rate at which each possible amino acid in a sequence remains unchanged over time; in other words, $W(\Gamma)$ measures the conservability of the matched subsequence Γ in both X and Y, which is an important concept in biology that emphasizes the importance of each region of the protein sequence.

Now we define S the matrix of matching scores, such as $S_{X,Y}$ is the matching score between X and Y two protein sequences belonging to the family F. The matching score $S_{X,Y}$, understood as representing the substitution relation of the conserved regions in both sequences, is defined as follows:

$$S_{X,Y} = \frac{\sum_{\Gamma \in E^{l}_{XY}} W(\Gamma)}{MAX(|X|,|Y|)} \quad (3)$$

Finally, the pairwise similarity matrix *SMS* of the protein family F is calculated by applying the Pearson's correlation coefficient to the matrix S.

4.2 Minimum length *l*

Our aim is to detect and make use of the significant motifs best conserved during evolution and to minimize the influence of those motifs which occur by chance. This motivates one of the major biological features of our similarity measure, the inclusion of all long conserved subsequences (i.e., multiple occurrences) in the matching, since it is well known that the longer the subsequences, the smaller the chance of their being identical by chance, and vice versa. Here we make use of the theory developed by Karlin *et al*. in [21,19,20] to calculate, for each pair of sequences, the value of l, the minimum length of matched subsequences. According to theorem 1 in [19] we have:

$$K_{r,N} = \frac{\log n(|Seq_1|,...,|Seq_N|) + \log\lambda(1-\lambda) + 0.577}{-\log\lambda} \quad (4)$$

$$n(|Seq_1|,...,|Seq_N|) = \sum_{1 \leq i_1 \leq ... \leq i_r \leq N} \prod_{\nu=1}^{r} |Seq_{i_\nu}| \quad (5),$$

$$\lambda = \max_{1 \leq \nu_1 \leq ... \leq \nu_r \leq N} \left(\sum_{i=1}^{m} \prod_{j=1}^{r} p_i^{(\nu_j)} \right) \quad (6),$$

$$\sigma_{r,N} \approx \frac{1.283}{|\log\lambda|} \quad (7)$$

This formula calculates $K_{r,N}$, the *expected length of the longest common word present by chance at least r times out of N m-letter sequences* [19] (i.e., $Seq_1,\ldots,Seq_N$), where $p_i^{(v)}$ is generally specified as the i^{th} residue frequency of the observed v^{th} sequence, and $\sigma_{r,N}$ the asymptotic standard deviation of $K_{r,N}$.

According to the conservative criterion proposed by Karlin *et al.* .[19], to measure the similarity between two protein sequences, we take into account all subsequences present 2 times out of the 2 sequences which have a length that exceeds $K_{r,N}$ by at least two standard deviations. In other words, for each pair of sequences, matched subsequences shorter than $l=K_{2,2}+2.\sigma_{2,2}$ (i.e., by fixing $N=r=2$) have a real chance of being similar as a result of random phenomena, while those with lengths greater than $l=K_{2,2}+2.\sigma_{2,2}$ are more likely to be conserved motifs. So, for each pair of protein sequences X and Y, we calculate a specific and appropriate value of l to calculate $S_{X,Y}$ the similarity between X and Y.

5. CLUSS: CLUSTERING ALGORITHM

CLUSS is composed of three main stages. The first one consists in building SMS, a pair-wise similarity matrix; the second, in building a phylogenetic tree according to this matrix, using a new clustering approach based on spectral decomposition; and the third, in identifying subfamily nodes from which leaves are grouped into subfamilies.

5.1 Stage 1: Similarity matrix SMS

Using one of the known substitution score matrices, such as BLOSUM62 [14] or PAM250 [6], we compute *SMS*, the *NxN* similarity matrix, where N is the number of sequences of the protein family F to be clustered, and $SMS_{i,j}$ is the similarity between the i^{th} and the j^{th} protein sequences of F. The construction of *SMS* takes CPU time proportional to $N(N-1)T^2/2$, with T the typical sequence length of the N sequences.

5.2 Stage 2: Phylogenetic tree

To build the phylogenetic tree, we adopt a strategy inspired by the latent semantic analysis approach (LSA) [4], widely used in information retrieval, in which data are mapped to a vector space of reduced dimension (i.e., less than the number of data). By using a hierarchical strategy, and starting from the protein sequences, each of which is represented by a vector in a Euclidian space (i.e., step 1 of this stage), and considered as the root node of a (sub)tree containing only one node, we iteratively join a pair of root nodes in order to build a bigger subtree. At each iteration, a pair of root nodes is selected if they are the most similar root nodes (i.e., corresponding vectors have the largest cosine product). This process ends when there remains only one (sub)tree, which is the phylogenetic tree. The present stage is composed of three steps, as follows:

5.2.1 Step1: Spectral decomposition of SMS

The main idea is to perform a spectral decomposition of the similarity matrix *SMS*, to map the protein sequences onto a vector space, thereby making use of its advantages, of which the most important for us is the conservability of distances.

Spectral decomposition of the square symmetric matrix *SMS* is done through Eigen decomposition [39]. We obtain:

$$SMS = V * V^T \quad (8)$$

$$V = \begin{pmatrix} V_1 \\ \vdots \\ V_N \end{pmatrix} \approx \begin{pmatrix} u_1^1 & \cdots & u_p^1 \\ \vdots & \ddots & \vdots \\ u_1^N & \cdots & u_p^N \end{pmatrix} * \begin{pmatrix} \sqrt{\lambda_1} & & \\ & \ddots & \\ & & \sqrt{\lambda_p} \end{pmatrix} \quad (9)$$

where $\lambda_1,\ldots,\lambda_p$ are the p non-negative eigenvalues of *SMS* and $u_1,\ldots,u_p$ are the p eigenvectors corresponding to the p eigenvalues.

For two vectors V_X and V_Y, in $\mathbb{Z}^N$, representing the protein sequences X and Y, respectively, the Euclidian inner product is defined as:

$$SMS_{X,Y} = \langle V_X, V_Y \rangle = \sum_{i=1}^{N} v_i^X * v_i^Y \quad (10)$$

When properly normalized (i.e., as proposed in section 4.1), the matrix *SMS* measures the correlation between protein sequences, which is similar to the role of the covariance matrix in principal component analysis (PCA). However, in the conventional PCA method, we must subtract the averages from the covariance matrix, which means that our method is not a PCA approach.

5.2.2 Step 2: Building the tree

The similarity between two root nodes referred to above is computed in the following way. At the beginning of the iteration, the similarity between any pair of nodes is initialized by the cosine product. We assign to each root node L (i.e., an individual leaf represents one protein sequence) a co-similarity c_L according to its importance in F.

By taking into account information about the neighborhood around each of the nodes L and R, the concept of co-similarity reflects the cluster compactness of all the sequences (leaf nodes) in the subtree. In fact, its value is inversely proportional to the within-cluster variance. As the subtree becomes larger, the co-similarity tends to become smaller, which means that the sequences within the subtree become less similar and the difference (separation) between sequences in different clusters becomes less significant. In simpler terms, the co-similarity is a measure of the balance between two nodes.

At the first iteration, all co-similarities are initialized to zero. Let L and R be the two most similar root nodes (i.e., cosine product of V_L and V_R is the largest) at a given iteration step; they are joined together to form a new subtree. Let P be the root node of the new subtree. P thus has two children, L and R, such that V_P, the corresponding vector of the new root node P. P and V_P have the following properties:

$$V_P = V_L + V_R \quad (11) \quad , \quad c_P = \frac{\|V_L\| * \|V_R\|}{\|V_L\| + \|V_R\|} \quad (12)$$

where V_L, V_R and V_P are vectors corresponding respectively to the root nodes L, R, and P, while $\|V_L\|$ and $\|V_R\|$ are modules of V_L and V_R; and c_P is the co-similarity of P. We assign a "length" value to each of the two branches connecting L and R to P, as follows:

$$d_{L,P} = \frac{\|V_R\|}{\|V_L\| + \|V_R\|} \quad (13) \quad , \quad d_{R,P} = \frac{\|V_L\|}{\|V_L\| + \|V_R\|} \quad (14)$$

These values are the estimate of the phylogenetic distance[1] from either node *L* or *R* to their parent *P* in the tree.

5.2.3 Step 3: Separating nodes

The CLUSS algorithm makes use of a systematic method for deciding which subtrees to retain as a trade-off between searching for the highest co-similarity values and searching for the largest possible clusters. We first separate all the subtrees into two groups, one being the group of high co-similarity subtrees and the other the low co-similarity subtrees. This is done by sorting all possible subtrees in increasing order of co-similarity and computing a separation threshold according to the method based on the maximum interclass inertia [7].

5.3 Stage 3: Extracting clusters

From the group of high co-similarity subtrees, we extract those that are largest. A high co-similarity subtree is largest if the following two conditions are satisfied: 1) it does not contain any low co-similarity subtree; and 2) if it is included in another high co-similarity subtree, the latter contains at least one low co-similarity subtree. Each of these (largest) subtrees corresponds to a cluster and its leaves are then collected to form the corresponding cluster.

6. RESULTS

To illustrate its efficiency, we tested CLUSS extensively on a variety of protein datasets and databases and compared its performance with that of some mainstream clustering algorithms. We analyzed the results obtained for the different tests with support from the literature and functional annotations. Full data files and results cited in this section are available on CLUSS website.

6.1 The clustering quality measure

To highlight the functional characteristics and classifications of the clustered families, we introduce the *Q-measure* which quantifies the quality of a clustering by measuring the percentage of correctly clustered protein sequences based on their known functional annotations. This measure can be easily adapted to any protein sequence database. The *Q-measure* is defined as follows:

$$Q\text{-}measure = \frac{\left(\sum_{i=1}^{C} P_i\right) - U}{N} \quad (15)$$

where N is the total number of clustered sequences, C is the number of clusters obtained, P_i is the largest number of obtained sequences in the i^{th} cluster belonging to the same function group according to the known reference classification, and U is the number of orphan sequences. For the extreme case where each cluster contains one protein with all proteins classified as such, the *Q-measure* is 0, since C becomes equal to N, and each P_i the largest number of obtained sequences in the i^{th} cluster is 1.

[1] This distance has no strict mathematical sense; it is merely a measure of the evolutionary distance between the nodes. It is closer to the notion of dissimilarity.

6.2 COG and KOG databases

To illustrate the efficiency of CLUSS in grouping protein sequences according to their functional annotation and biological classification, we performed extensive tests on the phylogenetic classification of proteins encoded in complete genomes, commonly named the Clusters of Orthologous Groups of proteins database [28]. As mentioned in the web site for the database, the COG (for unicellular organisms) and KOG (for eukaryotic organisms) clusters were delineated by comparing protein sequences encoded in complete genomes, representing major phylogenetic lineages. Each COG and KOG consists of individual proteins or groups of paralogs from at least 3 lineages and each thus corresponds to an ancient conserved domain. COG and KOG contain (to date) 192,987 and 112,920 classified protein sequences, respectively.

To perform a biological and statistical evaluation of CLUSS, we randomly generated two sets of 1000 large subsets, one from the COG database and the other from the KOG database. Each subset contains between 47 and 1840 non-orphan protein sequences (i.e., each selected protein sequence has at least one similar from the same functional classification) from at least 10 distinct groups in the COG or KOG classification. We tested CLUSS on both sets of 1000 subsets using each of the substitution matrices BLOSUM62 [14] and PAM250 [6]. The average *Q-measure* value of the clusterings obtained for the COG classification is superior to **88%** with a standard deviation of **5.61%**, and the value for the KOG classification is superior to **80%** with a standard deviation of **9.50%**. The results obtained show clearly that CLUSS is indeed effective in grouping sequences according to the known functional classification of COG and KOG databases.

In the aim of comparing the efficiency of CLUSS to that of alignment-dependent clustering algorithms, we performed tests using CLUSS, BlastClust [3], TRIBE-MCL [10] and gSPC [34] on the COG and KOG classifications. In all of the tests performed, we used the widely known protein sequence comparison algorithm ClustalW [36] to calculate the similarity matrices used by TRIBE-MCL [10] and gSPC [34]. Due to the complexity of alignment, these tests were done on two sets of six randomly generated subsets, named COG1 to COG6 for COG and KOG1 to KOG6 for KOG. The obtained results are summarized in Table 1.

The results in Table 1 show clearly that CLUSS obtained the best *Q-measure* compared to the other algorithms tested. Globally, the clusters obtained using our new algorithm CLUSS correspond better to the known characteristics of the biochemical activities and modular structures of the protein sequences according to COG and KOG classifications.

The execution time reported in Table 1 for algorithm comparison, show clearly that the fastest algorithm is BlastClust [3], closely followed by our algorithm CLUSS, while TRIBE-MCL [10] and gSPC [34], which use ClustalW [36] as similarity measures, are much slower than BlastClust [3].

6.3 Glycoside Hydrolase family 2 (GH2)

To show the performances of CLUSS with multi-domain protein families which are known to be hard-to-align and have not yet been definitively aligned, experimental tests were performed on 316 proteins belonging to the Glycoside Hydrolases family 2 (FASTA file is provided at CLUSS website) from the CAZy

Table 1. *Q-measure* (Q-m) and execution time (in seconds) obtained on each COG and KOG subset.

Protein sets and number of sequences	CLUSS+SMS		BlastClust		MCL+Clustal		SPC+Clustal	
	Q-m	Time	Q-m	Time	Q-m	Time	Q-m	Time
COG1 (336)	**96.73**	116	81.25	**10**	92.26	332	93.45	340
COG2 (214)	**95.33**	49	84.22	**7**	88.78	141	93.92	146
COG3 (215)	**93.06**	74	87.50	**14**	83.68	273	73.26	285
COG4 (355)	**90.42**	86	82.81	**12**	78.59	315	79.71	324
COG5 (667)	**98,08**	667	94.00	**105**	63.46	5393	70.01	5338
COG6 (309)	**95.15**	68	88.02	**18**	87.70	224	88.99	239
KOG1 (363)	**96.14**	414	67.21	**44**	69.69	1168	76.85	1209
KOG2 (425)	**90.12**	289	31.01	**27**	68.70	1208	53.64	1230
KOG3 (411)	**93.92**	258	42.33	**55**	74.85	270	75.91	325
KOG4 (360)	**93.06**	361	38.88	**127**	66.66	1123	67.22	1220
KOG5 (326)	**97.24**	221	77.91	**33**	75.46	688	82.51	718
KOG6 (590)	**90,68**	779	50.33	**405**	85.25	3782	66.94	4181

database [35]. The CAZy database describes the families of structurally-related catalytic and carbohydrate-binding modules or functional domains of enzymes that degrade, modify, or create glycosidic bonds. Among proteins included in CAZy database, the Glycoside Hydrolases are a widespread group of enzymes which hydrolyse the glycosidic bond between two or more carbohydrates or between a carbohydrate and a non-carbohydrate moiety. Among Glycoside Hydrolases families, the GH2 family, extensively studied at the biochemical level includes enzymes that perform five distinct hydrolytic reactions. Only complete protein sequences were retained for this study. In our experimentation, the GH2 proteins were subdivided into 28 subfamilies, organized in four main branches. Three branches correspond perfectly to enzymes with known biochemical activities. The first branch (subfamilies 1–7) includes enzymes with "*β-galactosidase*" activity from both Prokaryotes and Eukaryotes. The third branch (subfamilies 18 to 22) groups enzymes with "*β-mannosidase*" activity, while the fourth branch (subfamilies 23 to 28) includes "*β-glucuronidases*".

The clustering scheme obtained warrants further comment. The "orphan" subfamily 17 includes nineteen sequences labelled as "*β-galactosidases*" in databases. While the branch 1 "*β-galactosidases*" are composed of five modules, known as the "*sugar binding domain*", the "*immunoglobulin-like β-sandwich*", the "*(αβ)8-barrel*", the "*β-gal small_N domain*" and the "*β-gal small_C domain*", the members of subfamily 17 lack the last two of these domains, which makes them more similar to "*β-mannosidases*" and "*β-glucuronidases*". These enzymes are distinct from those of branch 1 [11] and their separate localization is justified.

The second branch is the most heterogeneous in terms of enzyme activity. However, most of the subfamilies (9 to 16) group enzymes that are annotated as "*putative β-galactosidases*" in databases. To the best of our knowledge, none of these proteins, identified through genome sequencing projects, have been characterized by biochemical techniques, so their enzymatic activity remains hypothetical. At the beginning of this branch, subfamily 8 groups enzymes characterized very recently: "*exo-β-glucosaminidases*" [5,16] and "*endo-β-mannosidases*" [17]. Again, theses enzymes share only three modules with the enzymes from branches 1, 3 and 4. The close proximity among "*exo-β-glucosaminidases*" and "*endo-β-mannosidases*" emerging from this work has not been described so far. Furthermore, subfamily 8 includes closely related plant enzymes with "*endo-β-mannosidase*" activity and bacterial enzymes produced by members of the genus *Xanthomonas*, including several plant pathogens. This could be an example of horizontal genetic transfer between members of these two taxa.

Subfamily 22, also found at the beginning of a branch, has been recently analyzed by Côté *et al.* [5] and Fukamizo *et al.* [13], using structure-based sequence alignments and biochemical structure-function studies. It was shown that proteins from this subfamily have a different catalytic doublet and could recognize a new substrate not yet associated with GH2 members.

Globally, the clustering result for the GH2 proteins corresponds well to the known characteristics of their biochemical activities and modular structures. The results obtained with the CLUSS algorithm were highly comparable with those of the more complex analysis performed by Côté *et al.* [5] and Fukamizo *et al.* [13] using clustering based on structure-guided alignments, an approach which necessitates prior knowledge of at least one 3D protein structure.

6.4 Group of 33 (α/β)8-barrel proteins

To show the performance of CLUSS with multi-domain protein families which are known to be hard to align and have not yet been definitively aligned, experimental tests were performed on the group of the 33 $(\alpha/\beta)_8$-barrel proteins, a group within Glycoside Hydrolases family 2 (GH2), from the CAZy database [35], studied recently by Côté *et al.* [5] and Fukamizo *et al.* [13]. The periodic character of the catalytic module known as "$(\alpha/\beta)_8$-barrel" makes these sequences hard to align using classical alignment approaches. The difficulties in aligning these modules are comparable to the problems encountered with the alignment of tandem-repeats, which have been exhaustively discussed [15]. The FASTA file and clustering results of this subfamily are available on the CLUSS website. This group of 33 protein sequences includes "*β-galactosidase*", "*β-mannosidase*", "*β-glucuronidase*" and "*exo-β-D-glucosaminidase*" enzymatic

activities, all extensively studied at the biochemical level. These sequences are multi-modular, with various types of modules, which complicate their alignment. Clustering such protein sequences using the alignment-dependent algorithms thus becomes problematic. In our experiments, we tested quite a few known algorithms to align the 33 protein sequences, such as MUSCLE [8], ClustalW [36], MAFFT [18], T-Coffee [26] etc. The alignment results of all these algorithms are in contradiction with those presented by Côté *et al.* [5] which in turn are supported by the structure-function studies of Fukamizo *et al.* [13]. This encouraged us to perform a clustering on this subfamily, to compare the behaviour of CLUSS with BlastClust [3], TRIBE-MCL [10] and gSPC [34] in order to validate the use of CLUSS on the hard-to-align proteins. The experimental results with the different algorithms are summarized in Table 2, which shows the cluster correspondence of each of the sequences by approach used. An overview of the results is given below. The corresponding names and database entries of the 33 $(\alpha/\beta)_8$-barrel proteins group are indicated at CLUSS website.

6.4.1 CLUSS results

The 33 $(\alpha/\beta)_8$-barrel proteins were subdivided by CLUSS into five subfamilies, organized in five main branches (details in Figure 2). The first and the second branch correspond, respectively, to the first and the second clusters, which include enzymes with "*β-mannosidase*" activities; the third branch corresponds to the third cluster, which includes enzymes with "*β-glucuronidase*" activities; the fourth branch corresponds to the forth cluster, which includes enzymes with "*β-galactosidase*" activities; the fifth branch corresponds to the fifth cluster, which includes enzymes with "*exo-β-D-glucosaminidase*" activities.

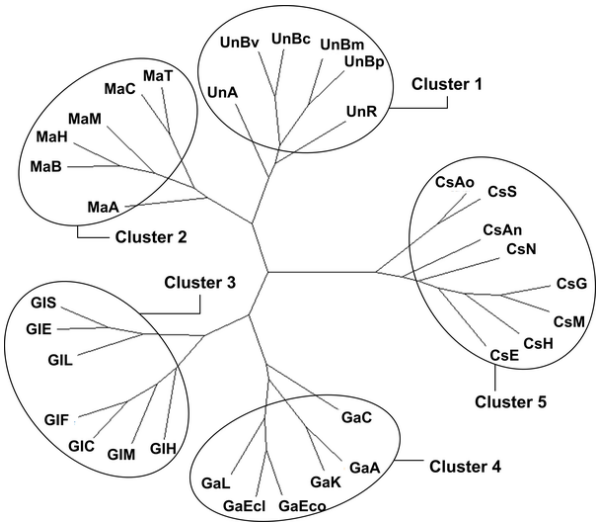

Figure 2. Phylogenetic analysis of 33 $(\alpha/\beta)_8$-barrel group

6.4.2 BLAST results

The 33 $(\alpha/\beta)_8$-barrel proteins were subdivided into five subfamilies. Almost all the enzymes were clustered in the appropriate clusters, except for seven proteins that were unclustered, among which we find the following well-classified enzymes: the "*β-galactosidase*" enzymes GaA, GaK and GaC; the "*β-mannosidase*" enzyme UnBc; and the "*exo-β-D-glucosaminidase*" enzyme CsAo.

Table 2. Clustering results on 33 (α/β)8-barrel group

Protein set	Côté *& al.*	CLUSS	Blast	MCL	SPC
UnA	**1**	**1**	1	1	1
UnBv	**1**	**1**	1	1	1
UnBc	**1**	**1**	/	1	1
UnBm	**1**	**1**	1	1	1
UnBp	**1**	**1**	1	1	1
UnR	**1**	**1**	1	1	1
MaA	**2**	**2**	2	2	1
MaB	**2**	**2**	2	1	1
MaH	**2**	**2**	2	1	1
MaM	**2**	**2**	2	1	1
MaC	**2**	**2**	2	2	1
MaT	**2**	**2**	2	2	1
GIC	**3**	**3**	3	2	2
GIE	**3**	**3**	3	2	2
GIH	**3**	**3**	3	2	2
GIL	**3**	**3**	3	2	2
GIM	**3**	**3**	3	2	2
GIF	**3**	**3**	3	2	2
GIS	**3**	**3**	3	2	2
GaEco	**4**	**4**	4	2	2
GaA	**4**	**4**	/	2	2
GaK	**4**	**4**	/	2	2
GaC	**4**	**4**	/	2	2
GaEcl	**4**	**4**	4	2	2
GaL	**4**	**4**	4	2	2
CsAo	**5**	**5**	/	2	3
CsS	**5**	**5**	5	2	3
CsG	**5**	**5**	5	2	3
CsM	**5**	**5**	5	2	3
CsN	**5**	**5**	/	2	3
CsAn	**5**	**5**	/	2	3
CsH	**5**	**5**	5	2	3
CsE	**5**	**5**	5	2	3

6.4.3 Tribe-MCL results

The 33 $(\alpha/\beta)_8$-barrel proteins were subdivided by TRIBE-MCL into two mixed subfamilies. We find the "*β-mannosidase*" enzymes MaA, MaC and MaT grouped in the "*β-galactosidase*" subfamily. Furthermore, the "*exo-β-D-glucosaminidase*" and "*β-glucuronidases*" enzymes are grouped in the same subfamily.

6.4.4 gSPC results

The 33 $(\alpha/\beta)_8$-barrel proteins were subdivided by gSPC into three subfamilies. Almost all the enzymes were grouped in the appropriate subfamily, except for the "*β-galactosidases*" and the "*β-glucuronidases*" which were grouped in the same subfamily.

Globally, the clustering of the 33 $(\alpha/\beta)_8$-barrel proteins generated by CLUSS corresponds better to the known characteristics of their biochemical activities and modular structures than do those yielded by the other algorithms tested. The results obtained with our new algorithm were highly comparable with those of the more complex, structure-based analysis performed by Côté *et al.* [5] and Fukamizo *et al.* [13].

7. DISCUSSION

The new similarity measure presented in this paper makes possible to measure the similarity between protein sequences based solely on the conserved motifs. Its major advantage compared to the alignment-dependent approaches is that it gives significant results with protein sequences independent of their alignability, which allows it to be effective on both easy-to-align and hard-to-align protein families. This property is inherited by CLUSS, our new clustering algorithm, which uses it as its similarity measure. CLUSS used jointly with SMS is an effective clustering algorithm for protein sets with a restricted number of functions, which is the case of almost all protein families. It more accurately highlights the characteristics of the biochemical activities and modular structures of the clustered protein sequences than do the alignment-dependent algorithms.

Our new clustering algorithm CLUSS gains several advantages by adopting an approach inspired by latent semantic analysis (LSA). The first is its use of high-dimensional space to automate the encoding and comparison of semantic relations. The second is its use of spectral decomposition, thereby benefiting from the global nature of this approach [27], since the Eigen decomposition used depends essentially on the globality of the similarity matrix *SMS*, and a change in one value in *SMS* makes changes in the entire Eigen decomposition.

So far, our similarity measure has been based on pre-determined substitution matrices. A possible future development is to propose an approach to automatically compute the weights of the conserved motifs instead of relying on pre-calculated substitution scores. There is also a need to speed up the extraction of the conserved motifs and the clustering of the phylogenetic tree, to scale the algorithm on datasets that are much larger in size with many more biological functions.

We believe that CLUSS is an effective method and tool for clustering protein sequences to meet the needs of biologists in terms of phylogenetic analysis and function prediction. In fact, CLUSS gives an efficient evolutionary representation of the phylogenetic relationships between protein sequences. This algorithm constitutes a significant new tool for the study of protein families, the annotation of newly sequenced genomes and the prediction of protein functions, especially for proteins with multi-domain structures whose alignment is not definitively established. Finally, the tool can also be easily adapted to cluster other types of genomic data.

8. REFERENCES

[1] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, D. J. Lipman. Basic local alignment search tool. *J. Mol. Bio.* 1990, 215:403–410.

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

[3] Basic Local Alignment www.ncbi.nlm.nih.gov/BLAST.

[4] M. W. Berry, R. D. Fierro. Low-rank orthogonal decomposition for information retrieval applications. *Numerical Linear Algebra Applications, Vol. 1*, 1996, 1-27.

[5] N. Côté, A. Fleury, E. Dumont-Blanchette, T. Fukamizo, M. Mitsutomi, R. Brzezinski. Two exo-β-D-glucosaminidases / exochitosanases from actinomycetes define a new subfamily within family 2 of glycoside hydrolases. *Biochem. J.* 2006, 394:675–686.

[6] M. O. Dayhoff, R. M. Schwartz, B. C. Orcutt. A model of evolutionary change in proteins. *Atlas of Protein Sequence and Structure vol. 5* 1978, suppl. 3:345-352.

[7] R. O. Duda, P. E. Hart, D. G. Stork. *Pattern Classification*, second edition, John Wiley and Sons, 2001.

[8] R. C. Edgar. MUSCLE: A multiple sequence alignment method with reduced time and space complexity. *BMC Bioinformatics* 2004, 5:113.

[9] R. C. Edgar. Local homology recognition and distance measures in linear time using compressed amino acid alphabets. *Nucl. Acids Res.* 2004, 32:380-385.

[10] A. J. Enright, S. Van Dongen, C. A. Ouzounis. An efficient algorithm for large-scale detection of protein families. *Nucl. Acids Res*. 2002, 30:1575-1584.

[11] S. Fanning, M. Leahy, D. Sheehan. Nucleotide and deduced amino acid sequences of Rhizobium meliloti 102F34 lacZ gene: Comparison with prokaryotic beta-galactosidases and human beta- glucuronidase. *Gene* 1994, 141:91-96.

[12] J. Felsenstein. An alternating least squares approach to inferring phylogenies from pairwise distances. *Syst. Biol.* 1997, 46:101.

[13] T. Fukamizo, A. Fleury, N. Côté, M. Mitsutomi, R. Brzezinski. Exo-β-D-glucosaminidase from Amycolatopsis orientalis: Catalytic residues, sugar recognition specificity, kinetics, and synergism. *Glycobiology* 2006, 16:1064-1072.

[14] S. Henikoff, J. G. Henikoff. Amino acid substitution matrices from protein blocks. *Proceedings of the National Academy of Sciences of the United States of America* 1992, 89:10915-10919.

[15] D. Higgins. Multiple alignment. In *The Phylogenetic Handbook: A Practical Approach to DNA and Protein Phylogeny*. Edited by Salemi M, Vandamme A.M. Cambridge University Press, 2004:45-71.

[16] M. Ike, K. Isami, Y. Tanabe, M. Nogawa, W. Ogasawara, H. Okada, Y. Morikawa. Cloning and heterologous expression of the exo-β-D-glucosaminidase-encoding gene (gls93) from a filamentous fungus, Trichoderma reesei PC-3-7. *Appl. Microbiol. Biotechnol.* 2006, 72: 687–695.

[17] T. Ishimizu, A. Sasaki, S. Okutani, M. Maeda, M. Yamagishi, S. Hase. Endo-beta-mannosidase, a plant enzyme acting on N-glycan: Purification, molecular cloning and characterization. *J. Biol. Chem.* 2004, 279:3855-3862.

[18] K. Katoh, K. Misawa, K. Kuma, T. Miyata. MAFFT: A novel method for rapid multiple sequence alignment based on fast Fourier transform. *Nucl. Acids Res.* 2002, 30:3059-3066.

[19] S. Karlin, G. Ghandour. Comparative statistics for DNA and protein sequences: Single sequence analysis. *Proc. Natl. Acad. Sci. USA* 1985, 82:5800-5804.

[20] S. Karlin, G. Ghandour. Comparative statistics for DNA and protein sequences: Multiple sequence analysis. *Proc. Natl. Acad. Sci. USA* 1985, 82:6186-6190.

[21] S. Karlin, F. Ost. Maximal length of common words among random letter sequences. *The Annals of Probability* 1988, 16:535-563.

[22] K. Kimura. Evolutionary rate at the molecular level. *Nature*, 1968 217:624–626.

[23] A. Krause, J. Stoye, M. Vingron. The SYSTERS protein sequence cluster set. *Nucl. Acids Res.* 2000:28:270–272.

[24] H. Lodish, A. Berk, P. Matsudaira, C. A. Kaiser, M. Krieger, M. P. Scott, L. Zipursky, J. Darnell. *Molecular Cell Biology*, 5th ed. New York and Basingstoke: W.H. Freeman and Co., 2004.

[25] D. W. Mount. *Bioinformatics. Sequence and Genome Analysis (2nd ed.)*, Cold Spring Harbor Laboratory Press, New York, 2004.

[26] C. Notredame, D. Higgins, J. Heringa. T-Coffee: A novel method for multiple sequence alignments. *Journal of Molecular Biology* 2000, 302:205-217.

[27] A. Paccanaro, J. A. Casbon, M. A. S. Saqi. Spectral clustering of protein sequences. *Nucleic Acids Research*. 2006, Vol. 34, No. 5 1571–1580.

[28] Phylogenetic classification of proteins encoded in complete genomes: www.ncbi.nlm.nih.gov/COG.

[29] P. Pipenbacher, A. Schliep, S. Schneckener, A. Schonhuth, D. Schomburg, R. Schrader. ProClust. Improved clustering of protein sequences with an extended graph-based approach. *Bioinformatics* 2002, 18:S182–S191.

[30] G. Reinert, S. Schbath, M. S. Waterman. Probabilistic and statistical properties of words: An overview. *J. Comp. Biol.* 2000, 7:1-46.

[31] J. Rocha, F. Rossello, J. Segura. The Universal Similarity Metric does not detect domain similarity. *Q-bio.QM* 2006, 1:0603007.

[32] K. Sjölander. Phylogenomic inference of protein molecular function: Advances and challenges. *Bioinformatics* 2004, 20:170-179.

[33] E. L. L. Sonnhammer, V. Hollich. Scoredist: A simple and robust sequence distance estimator. *BMC Bioinformatics* 2005, 6:108.

[34] I. V. Tetko, A. Facius, A. Ruepp, H. W. Mewes. Super paramagnetic clustering of protein sequences. *BMC Bioinformatics* 2005, 6:82.

[35] The carbohydrate-active enzymes (CAZy) database: afmb.cnrs-mrs.fr/CAZY.

[36] J. D. Thompson, D. G. Higgins, T. J. Gibson. CLUSTAL W: Improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucl. Acids Res.* 1994, 22:4673-4680.

[37] J. S. Varré, J. P. Delahaye, R. Rivals. The transformation distance: A dissimilarity measure based on movements of segments. *Bioinformatics* 1999, 15:194–202.

[38] S. Vinga, J. Almeida. Alignment-free sequence comparison – A review. *Bioinformatics* 2003, 19:513-523.

[39] H. P. William, A. T. Saul, T. V. William, P. F. Brian. *Numerical recipes in C (2nd ed.): The art of scientific computing*, Cambridge University Press, New York, NY, 1992.

[40] G. Yona, N. Linial, M. Linial. ProtoMap: Automatic classification of protein sequences and hierarchy of protein families. *Nucl. Acids Res.* 2000, 28:49-55.