DIMACS Technical Report 2004-33
July 2004

Automatic screening for groups of orthologous genes in comparative genomics using multiple-component clustering

by

Akshay Vashist
Dept. of Computer Science
Rutgers University
New Brunswick, New Jersey 08854

Casimir A. Kulikowski
Dept. of Computer Science
Rutgers University
New Brunswick, New Jersey 08854

Ilya Muchnik
DIMACS
Rutgers University
New Brunswick, New Jersey 08854

DIMACS is a collaborative project of Rutgers University, Princeton University, AT&T Labs–Research, Bell Labs, NEC Laboratories America and Telcordia Technologies, as well as affiliate members Avaya Labs, HP Labs, IBM Research and Microsoft Research. DIMACS was founded as an NSF Science and Technology Center.

ABSTRACT

To understand evolutionary relationships among genes from different organisms is a problem in modeling evolutionary history while solving practical problems related to functional annotation of genes. We have developed automatic method for discovering groups of gene sequences present in different organisms that are functionally related through evolution.

We have developed a new clustering method, which allows us to build clusters from multi-component types of data. In our case the data is a large set of genomes in which one has to find clusters that are groups of orthologous genes, focusing on hyper-inter-similarities among genes from different genomes more than the intra-similarities among genes from the same genome.

We have found that discovering these groups provides a "strong draft" of the complete picture of orthologous relations among genes in the complete genomes studied. Comparisons of these groups with the well-known semi-automatically extracted clusters of orthologous groups, COG [17, 18] shows strong correlation between these two systems of clusters. For instance, more than 85% of our clusters include genes from at least three different genomes and each of these genes belongs to COGs. These studies demonstrate that the method can be applied for an automatic screening of groups of orthologous genes in analyzing a large collection of genomes from different organisms.

1 INTRODUCTION

Many important problems in comparative genomics are based on the evolution of bio-molecules. Genetic elements, related through evolution, are called homologous sequences and provide a good means of approximating and studying the evolutionary origin while extrapolating our knowledge from well-studied organisms to new ones. For instance, such relationships are useful for gene functional annotation.

An important class of homologous sequences is that of orthologous genes, or gene sequences present in different genomes that have arisen through vertical descent from a single ancestral gene in the last common ancestor [8]. Such genes usually perform the same function(s) in respective organisms [12] but the degree of sequence similarity across the organisms may vary.

The main source of data to find groups of orthologous genes is a collection of complete genomes from different organisms. Databases consisting of several thousands of orthologous clusters have already been built based on existing genomic data [17, 9]. Unfortunately, current technology to build these databases requires manual curation. On the other hand, the genomic data, particularly for prokaryotes, is growing so fast that it will very soon be impossible to build the orthologous clusters using a manual analysis, even by a procedure requiring only partial manual curation. So, it is necessary to develop completely automatic "express" screening procedures to discover orthologous clusters within the rapidly growing genomic data. The presented work addresses precisely this problem. We have developed a new type of clustering method, which can serve as the basis for the express screening of such groups. Our method, which we call multiple-component clustering, extracts clusters of elements in a system with known multiple-component structure. An important characteristic of the method is that the extracted clusters mostly represent similarity relations between elements from different components, rather than similarity relations between elements within the components. The need of such clustering is apparent when one intends to determine "margin regions" among elements from predefined components in a space for classification. Such clustering is also useful when one wants to discover, simultaneously, clusters of entities and specific groups of variables that characterize each of the clusters, or, in machine learning problems for the analysis of confusion matrices between multiple classes. In bioinformatics, where bio-molecules are classified according to many classifications and each of these classifications may contain several thousands of classes (Pfam[5], SCOP [2], COG, TOGA [13] etc.) the need for such methods are very pressing. Indeed, the method simultaneously discovers interrelations between the classes while producing adjustments in the number of classes.

The main idea of the research presented here is to adapt the novel method to automatic screening of groups of orthologous genes. It is priori clear that those groups of genes should mostly contain similarity relations between genes from different genomes (orthologs) and pay much less attention to similarity relations among genes within genomes (paralogs).

Accordingly, it is adequate to apply the method to the problem by using a one-to-one correspondence between "components" in the method and individual complete genomes from the genomic data. There are two problems which need to be solved before one can apply the method to comparative genomic data to extract "candidates" for groups of orthologous genes: (i) introduce an appropriate similarity function between genes which can be used with the method, and, (ii) develop strong criteria and tests to validate, from a biological point of view, the candidates against "real orthologous clusters" which have been found by manual curation methods. The paper gives constructive answers to these questions, and, describes experimental results on known cluster data to demonstrate that the proposed tool can be considered as the first approximation for a screening procedure to find orthologous clusters automatically.

The paper is organized into five sections. Section II presents basis of our method. Section III describes an evaluation methodology of the proposed method on a test data using various validation criteria. Section IV discusses the results, and section V presents our conclusions and future directions.

2 MULTIPLE-COMPONENT CLUSTERING METHOD TO FIND ORTHOLOG CLUSTERS

As described earlier, multiple-component clustering involves finding groups of similar elements from different components in multiple-component data. The realization of our multiple-component clustering method is based on a method that actually finds a single cluster within a set of elements. The essence of this method lies in using so called tightness functions as objective criteria to extract clusters [14]. Furthermore, these tightness functions are used to characterize each subset by a tightness value, and the subset related to the maximal "tightness" value is designated as a core, interpreted as a cluster.

Our method involves an extension of the above procedure. We formulate the problem as core extraction on a family of sets instead of one set of elements as described in [14]. This allows us to find multiple-component clusters (clusters containing similar elements from different set-components) in a family of sets. The remainder of this section briefly summarizes this single cluster finding procedure, followed by our extension of this procedure to finding multiple-component clusters and its adaptation to the problem of finding groups of orthologous genes.

Finding a core for a single set of elements [14]: Suppose $W = \{1, 2, \ldots, i, \ldots, n\}$ is a set of elements in which we intend to find a core. To do this, one has to introduce a linkage function $\pi(i, H)$ which is interpreted as a degree of "belongingness" or "membership" of the element i to the subset H, where $i \in H \subseteq W$. The linkage function has to be monotonically increasing i.e., $\pi(i,H) \geq \pi(i,H_1)\ \forall H_1 \subseteq H$. Based on the linkage function, a measure of tightness for the subset H, is defined by the formula, $F(H) = min_{i \in H}$. In other words, the tightness of a subset H is measured by the minimal value attained for the linkage function over all elements of this subset. Then, a core $\hat{H}$, a subset of the set W, is defined by $\hat{H} = argmax_{H \subseteq W} F(H)$. The optimal solution $\hat{H}$ is a combinatorial optimization problem. The monotonicity property of the linkage function is critical and leads to efficient determination of the solution. The largest core, which is a unique solution, can be found efficiently using the polynomial time procedure given in [14].

Our extension of core finding to a family of sets: Consider a family of sets, $G = \{G_1, G_2, \ldots, G_k\}$, where $G_s, s \in < 1, 2, \ldots, k >$ is a set; and let $H = \{H_1, H_2, \ldots, H_k\}$ be a family of subsets where $H_s \subseteq G_s, s \in < 1, 2, \ldots, k >$. We will also use G to denote the set containing elements of all its member sets and H to denote the set containing elements of its member sets [1]. The sense of usage would be clear from the context. The multiple-component cluster we extract as a core should contain similar elements from different sets in the family. Therefore, we need a similarity measure between elements belonging to different sets in the family. For any $i_t \in H_t,\ t = \overline{1, \ldots, k}$ and all $H_s,\ s = \overline{1, 2, \ldots, k}\ s \neq t$, let us consider a monotone linkage function $\pi_{ts}(i_t, H_s) : \pi_{ts}(i_t, H_s) \geq \pi_{ts}(i_t, H'_s)\ \forall H'_s \subseteq H_s$. Using this family of linkage functions, we build our multiple-component linkage function as

$$\pi_{mc}(i,H) = \begin{cases} \sum\limits_{s=1:s\neq t}^{k} \pi_{ts}(i_t, H_s), & if \quad (i = i_t) \wedge (\exists s : H_s \neq \phi) \\ 0, & otherwise \end{cases} \tag{1}$$

It is obvious that $\pi_{mc}(i,H)$ is also a monotone function. According to (1), an element $i_t \in H_t$ is not influenced by any of the elements from its component H_t and the multi-component linkage function considers similarity values between elements belonging to different families in the set. Furthermore, analogous to the tightness function in [9], the *multiple-component tightness function* is defined as $F_mc = min_{i \in H} \pi_{mc}(i,H)$, and measures the homogeneity of the set H by the multiple-component linkage value associated with the "worst" element contained in that set. Then a multi-component cluster is defined as follows:

$$\hat{H} = argmax_{H \subseteq G} F_{mc}(H) \tag{2}$$

Since our multiple-component linkage function is also monotonically increasing, we can obtain the optimal solution in polynomial time using exactly the same procedure as given in [9].

[1]In this case it is clear that G is identified with a single set $W = \{1, 2, \ldots, i, \ldots, n\}$ of all the considered elements in all, $G_s, s =< 1, 2, \ldots, k >$; a similar analogy holds for H.

2.1 Partitioning of data into multiple-component clusters (MCC)

The procedure outlined above permits us to get one multiple-component cluster in the set G using (2). However, many such clusters are likely to be present in the set G. If we assume that these clusters are unrelated to one another, then we can use a simple heuristic of iteratively applying the above procedure to extract all these clusters. To do this one needs to remove the elements belonging to the first cluster $\hat{H}$ from G and extract another multiple-component cluster in the set $G - \hat{H}$. This idea can be iteratively applied to find all multiple component clusters. The procedure given in the figure below formalizes this idea:

Initialization: $G^0 := G;\ m := 0;\ C = \phi;$
Step 1: Extract $\hat{H}^m$ from G^m using (1); Add $\hat{H}^m$ to C;
Step 2: $G^{m+1} := G^m - \hat{H}^m;\ m := m + 1;$
Step 3: if ($(G^m = \phi) \wedge (m^s_{ij}t = 0 \forall i, j \in G^m)$)
Output C, G^m as R, and m; STOP;
else go to step 1

The clustering procedure produces a partition of the set G into the set of m clusters [2], $C = \{\hat{H}^0, \hat{H}^1, \ldots, \hat{H}^m\}$, and a set of residual elements [3], $R = \{i : i \in G \backslash C\}$. Each of the m non-trivial clusters in C contains at least two elements from at least two different components of the set G.

2.2 Automatic screening of groups of orthologous genes

Known methods to find clusters of orthologous genes have at least two stages - automatic and manual. The role of latter is to correct results of the first stage, which is some clustering procedure. Although specific implementations of clustering procedures in different methods vary, they all include some critical steps: build clusters based on a set of "mutually most similar pairs" of genes from different genomes, such pairs are called BBH (bi- directional best hits []) or BeT (best hits []). Unfortunately, this preprocessing is not robust and small changes in the data or in the set of free parameters used in the clustering procedure can alter the results dramatically. So, the current status of the problem to extract groups of orthologous genes has two bottlenecks: (a) the manual curation, and (b) the hypersensitivity of the automatic stage. Our approach tries to resolve these two tasks simultaneously.

Our method considers the following informal definition of groups of orthologous genes - *a subset of genes from two or more organisms such that every gene in this subset is mostly related to genes from other organisms present in this subset.* This definition is just a verbal

[2]Note that the number n of non-trivial clusters is determined in the method automatically, as well as the number of different components presented in each of the non-trivial clusters.

[3]Formally, all these residual elements can also be considered as clusters, but they are "trivial" clusters, singletons.

representation of our idea of identifying these groups of genes with the multiple-component clusters determined above. Since orthologous genes must be present in different organisms, our definition of ortholog clusters rightly ignores similarity among genes within a genome and focuses on similarity values between genes from different organisms only. For the ortholog group clustering problem, the set $G = \{G_1, G_2, \ldots, G_k\}$ corresponds to the set of k different complete genomes under consideration, where G_l is the l^{th} genome and the elements in the component G_l corresponds to genes in that genome. Let $M^{st} = ||m_{ij}^{st}||$ represent the matrix of similarities between genes $i \in G_s$ and $j \in G_t, s \neq t$, then the pair-wise linkage function for genes in genomes G_s and G_t is defined as $\pi_{st}(i_s, H_t) = \sum_{j_t \in H_t} m_{ij}^{st}$. Based on this specific pair-wise linkage function, multiple-component linkage function $\pi_{mc}(i, H)$ is defined using the precise formula for $\pi_{st}(i_s, H_t)$ in (1). Following the earlier definition of the multiple-component tightness function, an orthologous group is extracted as a multiple-component cluster $\overline{H}$ in (2).

The method is efficient in speed and memory to perform orthologous group extraction on a large number of genomes, also, the method is robust to slight variations in pair-wise similarity values. Moreover, the proposed method does not require a high quality of "completeness" of the considered genomes, in other words it is able to extract the groups of orthologous genes in the existing set of genes, which is critical given the current state of genomic data.

3 ORTHOLOG CLUSTER VALIDATION

Generally speaking, validating multiple-component clusters would require developing quantitative measures to summarize homogeneity within the clusters, but validating groups of orthologous genes is a very specific problem. We develop validation methodology that assesses the homogeneity of groups of orthologous genes where prior knowledge of such groups exists. Although classical statistics contains a wide variety of indices dealing with the problem of comparing two classifications, our validation methodology does not strictly subscribe to a classical hypothesis testing paradigm. This is because when two classifications are "highly dependent", which we expect in our case, the quantitative measures regarding similarity remain largely unaddressed by classical statistics [4], especially for cases like ours where we have a large number of classes with large dataset sizes. Aggregative statistical measures summarize the relatedness of two classification schemes but fail to provide insight into per class based agreements between the classifications. We propose a specific methodology oriented towards explorative screening of orthologous relations between genes from different genomes. Other applications of the method may well require different validation tests.

[4] Although some measures exist , they are difficult to interpret and fail to provide enough resolution or an insight into aspects in which classifications agree.

We evaluate the performance of our method on the data set used to construct the COG database - an expertly curated database of groups of orthologous genes [5]. In order to have reliable estimates of association between ortholog clusters in COG and ortholog clusters extracted by our method, we calculate a (large) set of indices of comparison each of which reflects a particular aspect of similarity between the two classifications. Intuitively, the set of indices together should capture the biological notion involved in the classification problem studied here. We also supplement these indices with some of the standard statistical measures to provide an average comparison between the classifications.

To have a quantitative measure of overall similarity between two classifications, we use few statistical coefficients. Association between the two classifications is estimated using the standard χ^2 statistics and a normed version of the χ^2 coefficient, the *Cramer's V-coefficient*. In order to have direct measure of degree of association between COG clusters and our clusters, we also use the *Rand index* and its modification the *Adjusted Rand index*. The Rand indices involve counting the number of agreements between the two classes based on how each of pair of elements in the underlying set is assigned to classes in two classification schemes [6]. The tabulated results for these coefficients along with their use in our analyses are given in section IV.

The first group of indices to compare COGs classification with our clustering estimates the similarity between two data partitions into two classes: (i) COG, the set of sequences which belong to COGs, and $\overline{COG}$, the set of sequences which do not; (ii) the set of proteins, Cl, which belong to the union of our clusters, and the set $\overline{Cl}$ containing the remaining proteins. This group of indices, called *indices for* 2×2 *contingency table*, determines how sequences are classified into two classes - one class containing sequences that share orthologous relationship with other sequences in the data and the other class containing sequences which do not have any orthologous sequences in the data. These indices are denoted by α with indices. The second group of indices is represented by β's and is the set of *indices to assess homogeneity within MCC-clusters*. These indices evaluate each MCC-cluster: (a) what portion of a cluster contains sequences do not belong to any COG, (b) how does the complementary part distribute across all COGs. In effect, the *beta*-indices present a summary of homogeneity within our ortholog clusters. The third group consists of *indices assessing homogeneity within MCC-clusters based on homogeneity of COGs* and are represented by γ's. These indices are similar to the βs and focus on quantitating the relationship between a single COG and all MCC-clusters. The γ-indices involve decomposing each COG into MCC-clusters and are useful in measuring the aggregating ability of our MCC procedure (whether we produce very small but tight cluster or large but "loose" clusters). The α, β,

[5] Some other collections of orthologous genes exist such as KEGG [9] but COG is one of the most popular and trusted databases in the genomics community.

[6] It is easy to see that genometrical interpretation of these coefficients is hamming distance between binary equivalence relations, associated with two partitions on the same data. In our case partitions are defined by COG and our clustering.

and γ indices are described in this section and the results along with performance of MCC procedure as measured by these indices are presented in the following section IV.

3.1 Statistical coefficients for aggregative similarity measure between two partitions

The first and second groups of coefficients are based on an aggregative comparison of two partitions. In our case the two partitions are the classifications induced by COGs and the MCC-clusters. Although there are many statistical coefficients to measure the correspondence between two partitions, there is no single coefficient which suffices to present a case for strong validation or has an interpretation [3], therefore we use three different types of coefficients which complement each other in interpretation.

Since all measures of association we have used can be conveniently expressed on a contingency table (also called the cross-classification table) representing the joint distribution of elements according to the two classifications. Assume that the elements in the set G are classified according to two classifications, $\mathcal{A}$ and $\mathcal{B}$, so that they induce partitions, $A = \{a_1, \ldots, a_r\}$, and $B = \{b_1, \ldots, b_c\}$, respectively, i.e., $\cup_i = 1^r = G = \cup_j = 1^c b_j$ and $a_i \cap a_{i'} = \phi = b_j \cap b_{j'}$ for $1 \leq i \neq i' \leq r$ and $1 \leq j \neq j' \leq c$. A contingency table, T, corresponding to these two classifications is an $r \times c$ matrix, $T = ||t_{ij}||$, where t_{ij} is the number of elements belonging to $a_i \cap b_j$ i.e., t_{ij} is the number of elements co-classified in classes $a_i \in A$ and $b_j \in B$. The marginal sums $t_{i.} = \sum_j = 1^c t_{ij}$ and $t_{.j} = \sum_i = 1^r t_{ij}$ is the number of elements in classes a_i and b_j, respectively and $t = \sum_i = 1^r \sum_j = 1^c t_{ij}$ is the total number of elements in the set G ($t = |G|$).

3.1.1 Test of independence

One of the classical tests for assessing the independence of two classifications is the χ^2 goodness-of-fit test. For an $r \times c$ contingency table, T, this coefficient is given by the formula:

$$\chi^2 = \sum_{i=1}^{r} \sum_{j=1}^{c} \frac{(t_{ij} - t_{i.}t_{.j}/t)^2}{t_{i.}t_{.j}/t} \tag{3}$$

In order to assess if the two classifications are independent, the χ^2 value needs to be compared with the value of χ^2-distribution for appropriate degrees of freedom in the contingency table. The degrees of freedom for a $r \times c$ contingency table is $(r-1)(c-1)$ [7]. Then, to assess the independence of two classifications, for some *a priori* determined significance level, $\alpha, 0 \leq \alpha \leq 1$ (usually, α is 0.05 or 0.01), the χ^2 value in (3) is compared with the value attained by the χ^2-distribution for parameters α and $(r-1)(c-1)$ degrees of freedom. If the

		Partition B				
	Class	b_1	b_2	$\cdots$	b_c	*Sums*
	a_1	t_{11}	t_{12}	$\cdots$	t_{1c}	$t_{1.} = \sum_{j=1}^{c} t_{1j}$
	a_2	t_{21}	t_{22}	$\cdots$	t_{2C}	$t_{2.} = \sum_{j=1}^{c} t_{2j}$
Partition A	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
	a_r	t_{r1}	t_{r2}	$\cdots$	t_{rc}	$t_{r.} = \sum_{j=1}^{c} t_{rj}$
	Sums	$t_{.1} = \sum_{i=1}^{r} t_{i1}$	$t_{.2} = \sum_{i=1}^{r} t_{i2}$	$\cdots$	$t_{.c} = \sum_{i=1}^{r} t_{ic}$	$t = \sum_{i=1}^{r} \sum_{j=1}^{c} t_{ij}$

Table 1: An $r \times c$ contingency table.

value for the χ^2-distribution is larger than our χ^2 value, the two classifications are inferred to be dependent. The χ^2 value can assume any value greater than 0 and this raw value by itself has no relevance in describing the degree of association between the two classifications. There are several modifications of the coefficients, which attempt to normalize the χ^2 value so that the value of resulting statistics lie between 0 and 1 and takes the extreme values under independence and complete association. One such modification thatwe have used in our study is Cramer's V-coefficient given below:

$$V = \frac{\chi^2/t}{min\{r-1, c-1\}} \tag{4}$$

It is the most popular of the chi^2-based coefficients since it provides a good normalization on the 0 to 1 interval regardless of the table size [7]. The V coefficient can be interpreted as the association between two classifications as a percentage of their maximal possible variation; it attains the value 1 when the two marginals associated with each class are equal. The comparative results using these two coefficients are presented in Table 5 in section 4.2 .

3.1.2 Pair-wise agreement based coefficients

The χ^2 statistics allows us to infer if two classifications are randomly related, or they are dependent. The Rand index [15] belongs to the complementary class of coefficients that directly attempts to measure the degree of association between classifications. It involves counting the number of pairs of elements of the set G, which are similarly classified by the two classification methods. Suppose t_1 represents the number of pairs of elements that are classified in the same class in both the classifications, t_2 represents the number of pairs of elements that are classified into different classes in both classifications, t_3 represents the number of pairs of elements that are classified into same class in classification $\mathcal{A}$ but into different classes in classification $\mathcal{B}$, and t_4 represents the number of elements that are classified into different classes in $\mathcal{A}$ but into the same class in $\mathcal{B}$. Using these numbers, one can calculate two coefficients $R = t_1 + t_2$ and $D = t_3 + t_4$. Intuitively, two similar classifications produce large values of R and small values of D. The corresponding indices are calculated by the formulas:

$$R_{ind} = \binom{t}{2} + \sum_{i=1}^{r}\sum_{j=1}^{c} t_{ij}^2 - \frac{1}{2}\left[\sum_{i=1}^{r} t_{i.}^2 + \sum_{j=1}^{c} t_{.j}^2\right] \quad (5)$$

$$D_{ind} = \frac{1}{2}\left[\sum_{i=1}^{r} t_{i.}^2 + \sum_{j=1}^{c} t_{.j}^2\right] - \sum_{i=1}^{r}\sum_{j=1}^{c} t_{ij}^2 \quad (6)$$

In [10] proposed Adjusted Rand index, R_{adj}, assuming a hyper-geometric distribution as the model of randomness, i.e., the $\mathcal{A}$ and $\mathcal{B}$ classifications are random as long as the number of elements in classes of the two classifications remains fixed. Clearly, this is a normalized index which is bounded above by 1 and takes the value 0 when the index equals its expected value, and is given by the formula:

$$R_{adj} = \frac{\sum_{i=1}^{r}\sum_{j=1}^{c}\binom{t_{ij}}{2} - [\sum_{i=1}^{r}\binom{t_{i.}}{2}\sum_{j=1}^{c}\binom{t_{.j}}{2}]/\binom{t}{2}}{\frac{1}{2}[\sum_{i=1}^{r}\binom{t_{i.}}{2} + \sum_{j=1}^{c}\binom{t_{.j}}{2}] - [\sum_{i=1}^{r}\binom{t_{i.}}{2}\sum_{j=1}^{c}\binom{t_{.j}}{2}]/\binom{t}{2}} \quad (7)$$

We use these coefficients to present the comparison of our clustering results with the benchmark ortholog clusters in COG, in section 4.2 .

3.2 Indices for 2×2 contingency table, α's

These indices deal with determining the agreement between two classifications, each with two classes. Almost all statistical coefficients for this case are functions of the values in a "2×2 contingency table". In our case the table is presented in Table 2, where COG-$\overline{COG}$ and Cl-$\overline{Cl}$ are different partitions/classifications; the numbers $a, b, c,$ and d are the cardinalities of intersections $COG \cap Cl$, $COG \cap \overline{Cl}$, $\overline{COG} \cap Cl$, and $\overline{COG} \cap \overline{Cl}$, respectively. For instance, one such average coefficient is $\alpha_1 = \frac{a+d}{a+b+c+d}$. Additionally, we calculate local coefficients α_2, α_3, α_4, and α_5. All the α coefficients are described and explained in Table 3

	Cl	$\overline{Cl}$
COG	a	b
$\overline{COG}$	c	d

Table 2: A 2×2 contingency table (confusion table)

Index	Explanation
$\alpha_1 = \frac{a+d}{a+b+c+d}$	Represents accuracy of classification. $\alpha_1 \in [0, 1]$. Larger values are better.
$\alpha_2 = \frac{a}{a+b+c+d}$	Sequences classified as orthologous sequences in both classifications, expressed as a fraction of all elements. Larger values are better.
$\alpha_3 = \frac{b}{a+b+c+d}$	Fraction of sequences classified as orthologous sequences only COG database. Smaller values are better.
$\alpha_4 = \frac{c}{a+b+c+d}$	Fraction of sequences classified as orthologous sequences only by MCC procedure. Smaller values are better.
$\alpha_5 = \frac{d}{a+b+c+d}$	Sequences classified as non-orthologous sequences by both classifications, expressed as a fraction of all elements. Larger values are better.

Table 3: The expressions for the α-indices with their explanations.

and the comparison results using these coefficients are presented in section 4.3.

3.3 Indices to assess homogeneity within MCC-clusters, β's

This group of indices focuses on determining homogeneity within a MCC-cluster and is geared towards brining out the relationship between a MCC-cluster and the classification in COG. This set of indices evaluates individual clusters for various notions of homogeneity; further, the averages for each of the indices in this set summarize the homogeneity across all clusters. The various notions of homogeneity are based on the decomposition of a MCC-cluster into subsets such that elements in each subset belong to a single COG or to the set $\overline{COG}$. To formally define these indices, assume that i^{th} MCC-cluster, $\hat{H}_i, 1 \leq i \leq m$, contains l_0^i (l_0^i possibly 0) sequences that do not belong to any COG, and the remaining sequences belonging to $k_i (k_i \geq 0)$ different COGs with l_1^i sequences belonging to the COG that shares the largest number of elements with this cluster, l_2^i to the COG that shares the second largest

Index	Explanation
$\beta_1 = \frac{1}{m}\left\{\sum_{i=1:\beta_1^i \leq 0.5}^{m} \beta_1^i + \sum_{i=1:\beta_1^i > 0.5}^{m}(1-\beta_1^i)\right\}$, $\beta_1^i = \frac{l_0^i}{n_i}$	Average fraction of sequences in a cluster that belong to the set $\overline{COG}$. $0 \leq \beta_1 \leq 0.5$, lower values are better.
$\beta_2 = \frac{1}{m}\sum_{i=1}^{m} \beta_2^i,\ \beta_2^i = k_i$	Average number of COGs in a MCC-cluster. $\beta_2 \geq 1$, lower values are desirable.
$\beta_3 = \frac{1}{m}\sum_{i=1}^{m} \beta_3^i$ where $\beta_3^i = min_j\{(\sum_{w=1}^{j}) \geq 0.5\}$	Average of the minimum of number of COGs required to make at least half the cluster. $\beta_3 \geq 1$, lower values are desirable.

Table 4: Expressions and explanation of the indices to evaluate homogeneity of clusters with respect to COGs.

number of elements with this cluster, and so on, i.e., $l_1^i \geq l_2^i \ldots l_{k_i}^i$, $\sum_{a=0}^{k_i} l_a^i = n_i = |\hat{H}_i|$. This class of indices is summarized in Table 4.

The index, β_1^i, is the fraction of non-COG sequences in the i^{th} cluster, correspondingly the index, β_1, is the average of β_1^i over all MCC-clusters and provides an estimate of homogeneity within our clusters with respect to distinguishing elements in the sets COG and $\overline{COG}$. The lower values of this index are better, however, when cluster, $\hat{H}_i$, is a subset of the set $\overline{COG}$ this index assumes a value 1 but according to notion of homogeneity which this index is expected to capture, it is a perfectly homogeneous clusters, so, β_1 is not a simple average but modified so as to be appropriate for the intended notion of homogeneity (see Table 3). The index β_2^i is the number of different COGs in the cluster and β_2 is the simple average over all clusters. Since, for the purpose of evaluating, we count the set $\overline{COG}$ as a COG, it is clear that β_2 is at least 1 and the value 1 indicates that all clusters are perfectly homogeneous, i.e., each cluster contains either elements from single COG, or all its elements belong to the set $\overline{COG}$. The index β_2 provides only average estimate in that it fails to provide a good estimate of the degree of homogeneity within a cluster, for instance if β_2 is high but most sequences in a cluster belong to a single COG, the cluster should still be considered as acceptable. To consider this phenomenon, we devised another index β_3 which is an average of β_3^i's over all clusters. β_3^i is defined as the minimal number of COGs required to make up at least half the cluster, $\hat{H}_i$. Obviously, β_3 is at least 1, and since the lower values indicate that at least half the elements are in a few COGs, 1 is the best possible value it can achieve. The comparison between our ortholog clusters and COG ortholog clusters in presented in section 4.4 .

Index	Explanation
$\gamma_1 = \frac{1}{\lvert COG\rvert}\sum_{j=1}^{\lvert COG\rvert}\gamma_1^i,\ \gamma_1^i = \frac{c_j^0}{c_j}$	Average number of sequences in a COG that are not part of any MCC- cluster. $0 \leq \gamma_1 \leq 1$, lower values are preferred.
$\gamma_2 = \frac{1}{\lvert COG\rvert}\sum_{j=1}^{\lvert COG\rvert}\gamma_2^i,\ \gamma_2^i = s_j$	Average number of clusters into which a COG is shattered. $\gamma_2 \geq 1$, small values are better.
$\gamma_3 = \frac{1}{\lvert COG\rvert}\sum_{j=1}^{\lvert COG\rvert}\gamma_3^i$ where $\gamma_3^i = \sum_{w=1}^{s_j}\beta_2^{c_j^w}$	Average number of COGs in the clusters into which a COG is shattered. $\gamma_3 \geq 1$, small values are better.

Table 5: Expressions and explanation of indices used to indirectly evaluate our clusters, through the evaluation of COGs.

3.4 Indices assessing homogeneity within MCC-clusters based on homogeneity of COGs, γ's

The third group of indices, which is similar to the indices in the second group, is an indirect measurement of homogeneity within our clusters. In order to accomplish this, we actually evaluate COGs in such a way that they provide an evaluation of our clusters. Suppose that j^{th} COG, C_j, contains c_j elements of which c_j^0 elements that do not belong to any MCC-cluster and the remaining c_j^1 elements belong to s_j different MCC-clusters, $\hat{H}_{c_j^1}, \hat{H}_{c_j^2}, \ldots, \hat{H}_{c_j^{s_j}}$. Then, the index γ_1^j measures the fraction of elements in the COG C_j that do not belong to any MCC-clusters and the average index γ_1 is an indicator of fraction of COG elements that are not accounted for in the MCC-clusters. The index γ_1 can take values between 0 and 1 inclusive and lower values indicate that a larger fraction of data is correctly screened as orthologs by our clustering. Another measure of quality of our clusters is the number of clusters into which a single COG is shattered, a low value of such measure would mean that we correctly determine the orthologous relationships across orthologous family present in the COG, on the other hand, a higher value would mean that we recognize the orthologous relationships in closely related sequences only. To study this effect, we designed the index γ_2^j to assess the number of different MCC-clusters present in the j^{th} COG, and the related average index, γ_2 measures this effect for all COGs. It is apparent that this index is at least 1 and lower values are desirable. Although, the index γ_2 provides a good insight into the shattering phenomenon of our clusters it fails to bring out an important homogeneity related assessment of our clusters, viz., do the clusters that make up a COG contain elements from that specific COG only or they contain elements from other COGs as well? To investigate this, we studied another index, γ_3^j, which is the number of different COGs in the clusters required to make up the j^{th} COG, the corresponding index γ_3 measures the average number of number of different COGs in the clusters that make up a COG. The comparison results using the three γ indices are presented in secitno 4.5 .

4 EXPERIMENTAL STUDY: AUTOMATIC DETECTION OF ORTHOLOGOUS CLUSTERS IN 43 PROKARYOTE GENOMES

The Cluster of Orthologous Groups, COG [7], is a database of ortholog clusters constructed from 43 complete genomes containing 104,101 genes (version updated Feb 2003). Some of these genes are multi-domain sequences (two or more subunits, each with independent biological role) and are manually divided based on domains, as a consequence there are 108,091 sequences from which COG is constructed. Some of the genes in complete genomes are unique to organisms and do not have orthologous genes in genomes of other organisms, moreover, the construction procedure for COGs requires that orthologous genes be present in three or more organisms, as a result only 74,059 genes, or about 71% of all sequences in 43 complete genomes, belong to 3,307 ortholog clusters in the COG database. However, some of these genes are multi-domain genes, so such multi-domain sequences are divided into subsequences based on domains which results in a total of 77,114 sequences in COG clusters. Although 43 genomes are used to construct COGs, the genomes of closely related organisms are merged together, and the procedures sees only 26 different groups of genomes [8]

We compared the performance of our ortholog model on two datasets: a) all 108,091 sequences in the 43 complete genomes, and b) the 77,114 sequences belonging to orthologous clusters in COG database to test our ortholog model. In contrast to COG, which discover ortholog clusters across lineages, orthologs by our procedure were discovered across 43 complete genomes because we only have knowledge about which genomes are merged together but do not have access to the precise information about which genes in the merged genomes are actually considered as paralogs.

A first step in any procedure to discover relationships between genes is to assess similarity

[7]This is the largest version of COGs classification in which most of the considered proteins are single domain sequences. In the subsequent version of COGs classification over 66 organisms [16] there are many more multi-domain proteins. The proposed screening procedure does not carry out preprocessing for domain parsing, therefore, the experimental study was conducted on the older version.

[8]The subject of analysis in COGs is not genomes from individual organisms but organisms from different lineages of organisms, therefore, genomes of closely related organisms are merged together. Further, similar genes within the closely related genomes (merged genomes) are identified as paralogs and are considered as single units to discover orthologous relationships to construct COGs. This merging of genomes reduces the number of groups of genomes to 26.

among genes. Pair-wise similarity values among sequences in test data computed using the sequence search tool Blast [1] are available at the COG website, we used these pre-computed similarity scores in order to carry out a comparison with our model. The raw pair-wise similarity values measured by Blast not symmetric (a characteristic of Blast), so they were made symmetric by considering the stronger (smaller) of the e-value in either direction. Thus, similarity value $sim(i, j)$ used in our procedure is given by $sim(i, j) = max(s(i, j), s(j, i))$ where $s(i, j) = -\log\{e - value(i, j)\}$. It may be noted that when $e - value(i, j)$ assumes the value 0, the function $s(i, j)$ is undefined, therefore, we set $s(i, j) = 325$ in those cases.

Groups of orthologous genes produced by our method in the 43 complete genomes are analyzed and an overview of results is presented in section 4.2. This is followed by a validation of identified clusters against those in COGs, using the indices described in section III. Comparison results using the α, β, γ, and standard statistics are described in sections 4.3 through 4.6, respectively. Section 4.7 presents analyses of singletons and cluster of size 2, discovered by our procedure. Ortholog clusters produced by our procedure on COG data (77,114 sequences) are analyzed in section 5.1 and comparison of these clusters with those in COG using the α, β, γ, and standard statistics in presented in sections 5.2 through 5.5, respectively. The trivial clusters (clusters containing sequences from less than 3 organisms) produced by our procedure are the subject of an analysis in section 5.6.

4.1 Results on complete data set

Our first test data contains all sequences in 43 organisms. This is a strict evaluation of the procedure since the method is evaluated not only for correct clustering but also for predicting the sequences that should not belong to clusters i.e., the set of genes unique to some of the considered organisms. On this data, our procedure produced 38,285 clusters including 13,202 singletons, 16,806 clusters of size 2, and 8,277 clusters of size 3 or more. We divided these clusters into two disjoint sets: the good-clusters, clusters containing sequences from at least three different organisms, and the complementary set which contains either singletons or sequences that belong to clusters containing sequences from at most two different genomes. We found 7,701 good clusters and these are considered as MCC-clusters or our ortholog clusters, which are compared with COGs to evaluate our procedure.

A first step in comparison of clusters of orthologs extracted by our procedure and COG clusters involves the study of relationship between the sequences in these two clustering results with respect to the data from which the two sets of ortholog clusters are extracted. Our procedure produced 7,701 ortholog clusters containing 51,134 sequences compared to 3,307 COGs containing 77,114 sequences (see figure 3). The set of sequences in our ortholog clusters are mostly (over 98% of sequences) contained in the set of sequences in COGs. The sequences in our clusters contain sequences from 2,590 different COGs, among which 104 COGs are found to match our clusters exactly in terms of sequences.

χ^2	$V(\in[0,1])$	$R/\binom{t}{2}(\in[0,1])$	$R_{adj}(\in[0,1])$
168078155	0.505	0.804	0.665

Table 6: Values for classical statistical coefficients to measure the degree of correspondence between two classifications. The degrees of freedom for the contingency table, on which χ^2 value is computed, is 25,498,011.

Considering COGs as benchmark ortholog clusters on this data, the figure above shows that for every sequence predicted to be in the set of orthologs, there is some ortholog cluster in COG which contains it. On the other hand, there are 721 COGs whose sequences are not contained in our ortholog clusters. As described earlier, our ortholog model has a tendency to extract groups of orthologous genes such that all sequences in a group are mostly similar to each other; so, to test this claim, we investigate the nature of COGs that intersect with our ortholog clusters. Towards this, we quantitatively measured the homogeneity, h_a, of each COG as average similarity between sequences within a COG, i.e., $h_a = \frac{1}{|COG|^2}\sum_{i\in COG_a}\sum_{j\in COG_b} sim(i,j)$. We found that average of this measure, $\overline{h}$, for COGs that perfectly matched our clusters was 128, and the average for COGs that intersect with our clusters was 42 but this measure averaged only 13 for COGs which do not intersect with our clusters. Thus variation in values of $\overline{h}$ across COGs sharing different relationship with our clusters confirms the above stated characteristic of our method.

4.2 Comparison using standard statistics

We now present the use of standard or aggregative statistics, discused in section 3.1, to compare our MCC clustering results with ortholog clusters in the COG database. A χ^2 value of 168,078,155 for a table with 25,498,011 degrees of freedom means that classifications in COG and in our cluster are not independent with p-value 0. In rejecting the independence hypothesis between two classifications with a high confidence, one can assume that the considered classifications are dependent but the test provides no insight to the level of dependence. The Cramer's V-coefficient with a value 0.505 shows the degree of dependence between the two classifications. Although this value is normalized between 0 and 1, there is no known distribution followed by these values, which can suggest the relevance of the attained value.

The rand index with value 0.804 should be construed as 80% of all pairs in the data agree in being together classified into same classes in both classifications or into different classifications in both classifications. Although this normalized number seems large enough, there are no benchmarks against which this value can be compared. This is factored into the adjusted rand index which is not only normalized but is also standardized with respect to the expected value of two random classifications having exactly same number of classes and same number of elements in each class as the classifications being compared. A value of 0.665 for the adjusted rand index is an a high value implying a very high degree of statistical similarity between the two classifications.

In summary, classical statistics (the χ^2-coefficient, Cramer's V, and the rand indices) imply that results of MCC-clustering and clustering in COG are highly dependent. We now compare the results of two classifications using the indices designed us to have a detailed analysis of some of the aspects of MCC clustering procedure.

4.3 Comparison using α-indices

The α indices, mentioned in section 3.2 assess the similarity based on a two-class contingency table, and are presented in Table 7. The index α_1, which is widely known as accuracy of comparing two classification has a high value about 72% and most of this contribution comes from the positive prediction for orthologous sequences in COGs, as correctly captured by the index α_2, whose value is 0.499. This value may seem to be very far from the stated maximum 1.0 but it must be noted that according to the benchmark classification the maximum value of this index can be only 0.766. If we consider our classification also, a tighter bound for α_2 can be given by $0 \leq \alpha_2 \leq min\{\frac{a+b}{a+b+c+d}, \frac{a+c}{a+b+c+d}\}$, which for our 2×2 contingency table implies an upper bound of 0.508. Considering this tighter value of the upper bound, the obtained value of α_2 is extremely good.

Another index, conspicuous by its value, is α_4 with a value of only 0.010 - this indicates our procedure is almost perfect in not making false prediction about orthologs, i.e., sequences that do not have orthologous sequences in other genomes are rarely predicted to have orthologous sequences. Although a tighter upper bound for α_4 is $min\{\frac{a+c}{a+b+c+d}, \frac{c+d}{a+b+c+d}\}$, which is 0.234 for our table, the value 0.010 is very far from it strongly suggesting a very low rate of false positives.

The index α_5 has relatively moderate value of 0.224 and, deceptively, does not seem to carry any strong message but this value of α_5 in light of value of α_4 suggests that our procedure makes correct prediction for sequences in the set $\overline{COG}$ by assigning them to class $\overline{Cl}$. This is also obvious from the fact that refined bound for this index is $0 \leq \alpha_5 \leq min\{\frac{c+d}{a+b+c+d}, \frac{b+d}{a+b+c+d}\} = 0.234$, a value very close to 0.224.

The index α_3 also has a moderate value and it highlights the characteristic tendency of the procedure to recognize closely related sequences as orthologous clusters. We delve into this deeper, later, when we would investigate what kind of COGs lead to high value of this index by analyzing homogeneity of COGs and the kind of sequences that are lost in our MCC-clusters.

α_1	α_2	α_3	α_4	α_5
0.723	0.499	0.267	0.010	0.224

Table 7: Values of indeices to compare our ortholog clusters with ortholog clusters in COG database. Larger values for α_1, α_2, and α_5 are indicators of similarity between the two classifications, where as smaller values are considered better for α_3 and α_4.

The analysis using the α indeices suggests that genes predicted to be in our orthologous groups are almost always classified as orthologous genes in the benchmark classification COG, on the other hand, the procedure does mis-classify some of the genes that are classified as orthologous genes in COG. In summary, the α indeices bring out the high specificity of our method towards the orthologous genes. This analysis along with the $\overline{h}$ index used to characterize genes implies that it is mostly the orthologous genes that show weak similarity to the corresponding COGs are misclassified.

4.4 Comparison using β-indices

The homogeneity within MCC-clusters is evaluated by the β indices whose values are presented in Table 8. These indices are discussed in section 3.3. The low value 0.021 assumed by β_1 suggests that it is rare that MCC-clusters contain sequences that belong to the set $\overline{COG}$. In fact, this index, being a simple average does not say anything about the kind of clusters that make a high contribution to its value - it must be emphasized that small sized clusters (size 3 and 4) contribute disproportionately large values because even a single sequence from the set $\overline{COG}$ makes the corresponding β_1^i quite large. In fact, most of our clusters are more homogeneous than reflected by this average value.

The index β_2 with value 1.087 is very close to 1, the best possible value, suggesting that most of the MCC-clusters contain sequences from a single COG. The index β_3 having value 1.005 goes much further by implying that for most clusters contain a single COG that constitutes at least half of each cluster [9]. In fact, on average 1.082 COGs are required to make up at least 95% of a MCC-cluster, suggesting that even in cases where a MCC-cluster contains sequences from more than one COG, most sequences actually belong to a single COG.

The set of β-indices suggests that most of the MCC-clusters are homogeneous in that they contain sequences from a single COG. On the other hand, on average a COG is broken

[9] β_3 is 1.060 if we take the average of minimum number of COGs required to make up at least 80% of a cluster. It increases to 1.075 if we want to make up 90% of the cluster, and is 1.082 if we want to make up at least 95% of the cluster. As can be seen, β_3 is approaching β_2, in fact, they must be equal if we count COGs required to make a cluster.

$\beta_1(\in [0, 0.5]$	$\beta_2(\geq 1)$	$\beta_3(\geq 1)$
0.021	1.087	1.005

Table 8: Values of indices to compare our ortholog clusters with ortholog clusters in COG database. Smaller values are better for the β indices.

into many homogeneous MCC-clusters. It is rare that a MCC-cluster may contain sequences belonging to more than one COG; in fact even in such cases the sequences belong to COGs that are, actually, related at the annotation level.

4.5 Comparison using γ-indices

The values of various α and β indices consistently suggest that the set of MCC-clusters, Cl, contain sequences belonging to COGs with a high degree of accuracy and our clusters are homogeneous too. As discussed in section 3.4, the value of γ indices, evaluated on the decomposition of COGs into MCC-clusters, would give us insight into the ortholog from a different perspective. A value of 0.47 for γ_1 seems to be large and suggests that on an average about half the sequences in a COG are not classified into Cl; it must be emphasized that most of the contribution to this indices comes from small sized COGs and such COGs actually constitute a large fraction of all COGs, the simple average is oblivious to such a skewed distribution in size and produces a potentially misleading value for this index. The value 2.463 for the index γ_2 means that on an average a COG contains about 2.5 MCC-clusters, which is close to value 2, the ratio of number of MCC-clusters (7,701) and the number of COGs (3,307). This suggests that number of MCC-clusters in a COG is almost constant, although large COGs are more likely to be decomposed into relatively larger numbers of MCC-clusters. One of the more interesting indices in this category is γ_3 which has value of 3.033, which might be incorrectly interpreted as the total number of COGs in an MCC-cluster into which an average COG is decomposed. This is very misleading because each MCC-cluster would contains sequences from at least one COG, thus each MCC-cluster contributes an additional 1 for that COG itself. Therefore, a value which truly reflects the number of COGs in MCC-clusters into which an average COGs is decomposed would be $1 + \gamma_3 - \gamma_2$=1.57, which is a reasonably low value, close to 1, and, in fact, highlights that our clustering is stable even at these higher levels of correlations.

The analysis using the γ-indices shows that on average a COG is broken into 2.5 MCC-clusters and most of these MCC-clusters contain sequences that belong to the COG under consideration. This demonstrates the ability of MCC-clusters to capture the intended notion of groups of orthologous genes shows that the procedure is very specific in detecting closely related orthologous sequences.

γ_1	γ_2	γ_3
0.470	2.463	3.033

Table 9: Values of indices to compare our ortholog clusters with ortholog clusters in COG database. Smaller values are better for all the γ indices.

In summary, the indices designed by us to compare the two clustering results provide us with a detailed view of the results. All the coefficients evaluate the MCC-clusters close to the COG clusters but also bring out the characteristic feature of the MCC method to aggregate closely related sequences. The feature is a desirable one, since we are unlikely to err in pronouncing a sequence to be an ortholog when it is not.

A naive criticism for the feature to aggregate closely related sequences can be its tendency to recognize only closely related sequences, but it must be emphasized that we have all along considered COG to be a benchmark, which possibly can have errors. Moreover, orthologous groups in COGs are constructed on 26 groups of genomes with access to entire sequences information with a subsequent manual curation, where as the MCC-clusters are built only from quantitative measures of pair-wise sequence similarities between sequences across genomes.

5 CONCLUSIONS AND FUTURE WORK

In this section we begin with a description of what has been accomplished and possible directions in which the current work can be extended. These directions include incoporation of additional information and technical improvements to the method itself.

We have formulated a multiple-component clustering (MCC) approach and applied it to screening groups of orthologous genes in multiple genomes. The analysis of the results shows that orthologous clusters obtained using the MCC approach show a high degree of consistency with the "gold standard" in ortholog clustering, the COG clustering, which is manually curated.

The proposed method is not only suitable for explorative orthologous sequence analysis in large number of genomes but also for ortholog analysis in a group of closely related genomes. From a practical point of view, this is critically important because only closely related organisms allow the transfer of knowledge with high confidence. For instance, if some experiment on mouse shows important genetic event, from medical point of view, it is very likely to affect the biochemical pathways humans in a similar manner. Since performing experiments on humans is highly restricted for a variety of reasons, such knowledge is invaluable.

It must be noted that finding orthologous relations among genes from similar genomes is much more difficult problem than the general one, even if ortholog detection invoves careful manual curation. These difficulties arise because the similarities among related genes drawn from various organisms depends on the evolutionary distance between the organisms. In this context, it is important to point out that during the construction of COGs, very similar genomes are merged to form "hyper-genomes" prior to the extraction of orthologs. Our approach allows practitioners to "zoom in" and focus on an analysis appropriate for closely related organisms automatically, because the method guarantees finding an exact optimum for the chosen objective function, even when the difference between genome is small.

Incorporating additional information

Most studies related to the extraction of groups of orthologous genes, use taxonomical information for the organisms under study. We plan to incorporate additional information in our method. A common approach it to merge genomes of closely related organisms, as a preprocessing step. Since genes in genomes of closely related organisms are highly similar, such preprocessing improves the results of studies by allowing the procedures (possibly manual) to focus on orthologous genes that are wide spread in a large group of organisms. Incorporating taxonomical information in our procedure is likely to improve the results of our procedure. We explored some modifications of the procedure in these directions and found some encouraging results. Taxonomical information is available for most organisms, we will be introducing it into our procedure so that it can be used automatically.

Some of the most difficult cases in extracting of groups of orthologous genes are related to multi-domain protein families. Our procedure currently ignores this aspect of the problem and extracts clusters at the sequence level. We are working on a modifications of the procedure which would automatically extract domain level information from the collection of sequences and incorporate such information for ortholog group extraction.

An interesting application of the proposed method is to extract ortholog clusters related to a particular function in related genomes. This can shed light on the evolution and variations in these biochemical pathways in organisms under study. Although such an analysis can be performed as a part of general-purpose ortholog analysis, we would like to modify the procedure to be specific to the orthologous genes involved in the biochemical pathways of interest.

Technical Improvements

We have seen that the MCC procedure has a tendency to extract small size ortholog clusters. This feature ensures that clusters that contain sequences which share orthologous relationship, and is critical to compensate for the manual curation of orthologous clusters. Such a high sensitivity of the procedure has the drawback that our ortholog clusters are often subsets of orthologous clusters in a COG, which reduces the correlation among our clusters and the COG clusters. We can alleviate this by employing a second stage in clustering wherein closely related clusters from the first stage are merged together to produce final ortholog clusters. This second stage requires defining quantitative measures of similarity among first stage clusters, subsequent aggregation procedure for the second stage can then be very similar to the previous stage. Such a method would also minimize the orthologous sequences which are lost as trivial clusters, and hence classified as non-orthologous sequences.

It must be emphasized that clustering results obtained are closely linked to the choice of linkage function used to formulate the optimization problem. In demonstrating the utility of the proposed approach, we focused on a concrete example of linkage function. It would be interesting to study other types of reasonable functions, which we plan to do in the near future.

Although the current MCC procedure works efficiently on the data used to construct COGs (more than 100,000 genes), current trends in genomic data need a procedure which can efficiently screen orthologous clusters in higher order of magnitude or millions of sequences. Such an enhancement requires use of efficient data structures to work with limited memory as well as improvements to the current algorithm. The theory underlying the extraction of multiple component clusters permits certain approximations without compromising the optimality of clusters. We have explored some improvements in this direction and preliminary results are encouraging. We hope to be able to have an efficient procedure which will enable us to perform orthologous analysis for at least a million sequences in a reasonably short time.

References

[1] Altschul, S., Madden, T., Schaffer, A., Zhang, J., Zhang, Z., Miller, W., and Lipman, D. Gapped blast and psi-blast: a new generation of protein database search programs. *Nucleic Acids Reserach 25*, 17 (1997), 3389–3402.

[2] Andreeva, A., Howorth, D., Brenner, S. E., Hubbard, T. J. P., Chothia, C., and Murzin, A. G. Scop database in 2004: refinements integrate structure and sequence family data. *Nucleic Acids Reserach 32*, 1 (2004), 226–229.

[3] Arabie, P., and Boorman, S. Multidimensional scaling of measures of distance between partitions. *Journal of Mathematical Psychology 10* (1973), 148–203.

[4] B. Boeckmann, e. a. The swiss-prot protein knowledgebase and its supplement trembl in 2003. *Nucleic Acids Reserach 31*, 1 (2003), 365–370.

[5] Bateman, A., Coin, L., Durbin, R., Finn, R. D., Hollich, V., Griffiths-Jones, S., Khanna, A., Marshall, M., Moxon, S., Sonnhammer, E. L. L., Studholme, D. J., Yeats, C., and Eddy, S. R. The pfam protein families database. *Nucleic Acids Reserach 32*, 1 (2004), 138–141.

[6] Eddy, S. R. A review of the profile hmm literature from 1996-1998. *Bioinformatics 14* (1998), 755–763.

[7] Everitt, B. *The Analysis of Contingency Tables.* John Wiley & Sons Inc. , New York., 1977.

[8] Fitch, W. M. Distinguishing homologous from analogous proteins. *Systematic Zoology 19* (1970), 99–113.

[9] Fujibuchi, W., Ogata, H., Matsuda, H., and Kanehisa, M. Automatic detection of conserved gene clusters in multiple genomes by graph comparison and p-quasi grouping. *Nucleic Acids Reserach 28*, 2 (2002), 4096–4036.

[10] Hubert, L. J., and Arabie, P. Comparing partitions. *Journal of Classification 2* (1985), 193–218.

[11] J.Westbrook, e. a. The protein data bank:unifying the archive. *Nucleic Acids Reserach 30*, 1 (2002), 245–248.

[12] Koonin, E. V. An apology for orthologs - or brave new memes. *Genome Biology 2*, 4 (2001).

[13] Lee, Y., Sultana, R., Pertea, G., Cho, J., Karamycheva, S., Tsai, J., Parvizi, B., Cheund, F., Antonescu, V., White, J., Holt, I., Liang, F., and Quackenbush, J. Corss-referncing eukaryotic genomes: Tigr orthologous gene alignments (toga). *Genome Research 12*, 3 (2002), 493–502.

[14] Mirkin, B., and Muchnik, I. Layered clusters of tightness set functions. *Appl. Math. Lett. 15* (2002), 147–151.

[15] Rand, W. M. Objective criterion for the evaluation of clustering methods. *Journal of the American Statistical Association 66* (1971), 846–850.

[16] Tatusov, R., Fedorova, N., Jackson, J., Jacobs, A., Kiryutin, B., Koonin, E., Krylov, D., R, R. M., Mekhedov, S., Nikolskaya, A., Rao, B., Smirnov, S., Sverdlov, A., Vasudevan, S., Wolf, Y., Yin, J., and Natale, D. The cog database: an updated version includes eukaryotes. *BioMed Central Bioinformatics* (2003).

[17] Tatusov, R., Koonin, E., and Lipmann, D. A genomic perspective on protein families. *Science 278* (1997), 631–637.

[18] Tatusov, R., Natale, D., Garkavtsev, I., T.A.Tatusova, Shankavaram, U., Rao, B., Kiryutin, B., Galperin, M., Fedorova, N., and Koonin, E. The cog database: new developments in phylogenetic classification of proteins from complete genomes. *Nucleic Acids Reserach 29*, 1 (2001), 22–28.