JOURNAL OF COMPUTATIONAL BIOLOGY
Volume 25, Number 12, 2018

Pp. 1328–1338
DOI: 10.1089/cmb.2017.0266

Homology Detection Using Multilayer Maximum Clustering Coefficient

CAIO SANTIAGO,[1] VIVIAN PEREIRA,[2] and LUCIANO DIGIAMPIETRI[2]

ABSTRACT

Homologous sequences are widely used to understand the functions of certain genes or proteins. However, there is no consensus to solve the automatic assignment of functions to protein problem and many algorithms have different ways of identifying homologous clusters in a given set of sequences. In this article, we present an algorithm to deal with specific sets, the set of coding sequences obtained from phylogenetically close genomes (of the same species, genus, or family). When modeled as a graph, these sets have their own characteristics: they form more homogeneous and denser clusters. To solve this problem, our algorithm makes use of the clustering coefficient, which maximization can lead to the expected results from the biological point of view. In addition, we also present an algorithm for the identification of sequence domains based on graph topology. We also compared our results with those of the TribeMCL tool, a well-established algorithm of the area.

Keywords: clustering coefficient, domain detection, graph modeling, homology detection, local alignment, sequence clustering.

1. INTRODUCTION

THE DISCOVERY OF NEW PROTEIN FUNCTIONS is very important to understand the metabolic processes and even the behavior of organisms. Unfortunately, this is a very complex and costly process, and it becomes impractical to be performed entirely experimentally for all coding sequences (CDSs) of an organism. Thus, most researchers use only automatic tools to identify possible homologies with known sequences. This occurs because sequences that share common ancestors tend to also share their functions (Hardison, 2003; Xia, 2013), thus facilitating the discovery of probable function through a simpler and cheaper process.

The detection of homology is relatively simple for experts, but it is time-consuming compared with automatic processes. Although the automatic process is not as reliable as the one performed by the experts and there are many considerations on that approach (Bork and Koonin, 1998), the automatic detection of homologues is widely used in databases (Apweiler et al., 2004; Zdobnov et al., 2017) and has been fundamental for the understanding of genomes (Pieretti et al., 2009).

Many of the approaches to improve gene function automatic identification are based on graph theory (Enright and Ouzounis, 2000; van Dongen, 2000; Bolten et al., 2001; Abascal and Valencia, 2002; Pipenbacher et al., 2002). In this model, the sequences are represented as vertices and the edges receive values according to

[1]Bioinformatics, University of São Paulo, São Paulo, Brazil.
[2]School of Arts, Sciences and Humanities, University of São Paulo, São Paulo, Brazil.

some criteria, typically defined from local alignments. Some also use the concept of transitivity to find homologous groups, a concept derived from mathematics that states that if *a* and *b* are related, as well as *b* and *c*, then *a* and *c* should also be related. These relationships are easily applied to the ancestry of sequences (Sasson et al., 2002). Thus, each related component will be considered a group of homologous CDSs, or a family of CDSs.

Although homology is transitive, similarity is not, and works that are based on similarity can produce false relationships that merge two groups incorrectly (Sasson et al., 2002). Therefore, it is potentially problematic to group sequences in a way that is highly dependent on transitivity, which do not have more direct relationships, producing low-density groups.

Multidomain proteins are well-known cases that can undermine automatic annotation algorithms because they have well-conserved domains that are very representative of two or more unrelated sequences. These cases make it necessary for algorithms to define strategies, either by preventing the joining of unrelated groups (Pipenbacher et al., 2002) or as a postprocessing step (Enright and Ouzounis, 2000).

In this work, the focus is to group the set of all the CDSs obtained from completely sequenced genomes of phylogenetically close organisms. Thus, some characteristics are expected, among them, it is expected that the graph formed by the homology relationships is more homogeneous and denser. A significant part of the clusters, therefore, should contain at least one gene from each genome (the core genome of the genome set).

This work presents a sequence clustering algorithm that aims to maximize the density of the sequence clusters through a graph's metric called clustering coefficient, thus reducing the number of related groups strongly based only on transitivity. In addition, a case study involving 55 complete genomes of *Streptococcus pyogenes* strains and 69 complete genomes of the *Xanthomonadaceae* family was performed.

2. RELATED WORK

Most of the sequence clustering algorithms use graph-based modeling. This model allows making decisions considering the neighborhood, but the use of this model can imply a higher computational cost. Because of this, there are some works that use alternative representations. An example is the CD-Hit (Li et al., 2001), which is based on greedy decisions and thus it is very fast and was designed to work with a huge volume of data. However, it was not designed to identify homology, but rather to index CDS databases.

Another method of clustering common to many works is the use of single linkage, which consists of starting from single elements and then joining them in an iterative way. These algorithms determine a metric not only for comparing sequences but also for the comparison of groups. The linkage of sequences or groups aims to maximize or minimize this metric. Sasson et al. (2002) use the maximization of several metrics applied to the alignments' e-value of all the sequences of the groups. The Abascal and Valencia (2002) approach is based on the entropy of graphs, starting not from unitary sets, but from groups formed by the Ncut algorithm (normalized cut). This model is based on local decisions and, thus, it is important to consider the whole groups to avoid false relationships. However, the treatment of multidomain proteins is implicit, potentially not grouping the multidomain proteins correctly.

Bolten et al. (2001) developed a graph-based approach using the Smith and Waterman (1981) metric. The clusters found are the strongly connected components of the graph. This method evidences the creation of clusters that contain multidomains, not including them in the clusters of other sequences that have some of their domains. The algorithm needs a minimum initial alignment for the edges. This algorithm was extended (Pipenbacher et al., 2002) to work with edge significance filters and an additional step was also proposed for the separation of some groups.

TribeMCL (Enright et al., 2002) is widely used for clustering sequences and it is an application of the MCL (van Dongen, 2000) that was proposed for the clustering in graphs with strong biological motivation. The algorithm is based on hidden Markov models to simulate graph flow walks. This algorithm is quite robust, fast, and is little affected by small changes in the graph topology (Brohée and van Helden, 2006).

GeneRage (Enright and Ouzounis, 2000) uses a different approach for graphs, representing them as an array of similarity. And unlike other approaches, local alignments are only used for a kind of initial edge filter (considering only alignments that have e-value smaller than 10^{-10}). A symmetrization process is then applied. Since this is a transitivity-based algorithm, there is a postprocessing step specific to deal with multidomain sequences. One interesting thing about this approach is that it relies primarily on the topology of the graph and then performs an alignment check.

All these algorithms are based on local decisions, between neighboring sequences or near groups. However, none of these approaches makes use of a global metric that considers all sequences in decision-making. Therefore, we present in this article an algorithm that uses the clustering coefficient.

3. CLUSTERING COEFFICIENT

The clustering coefficient is a topological metric for graphs in which, for each subset of three connected vertices, it calculates the probability of these three vertices being a clique of size three, that is, these three vertices are all connected to each other (Algorithm 1). The denser the components of the graph, the closer to 1 will be its clustering coefficient; otherwise, the components with few edges tend to have their coefficient closer to 0. An important characteristic of this coefficient is that it is not impacted by isolated components, and if in each component all the vertices are totally connected to each other, then the graph clustering coefficient will be 1.

Algorithm 1: Graph clustering coeffecient calculation

```
   Data: Graph g
 1 possible = 0;
 2 cliques = 0;
 3 for v in vertices(g) do
 4 |  for α in neighbors(v) do
 5 |  |  for β ∃ neighbors(v) | α ≠ β do
 6 |  |  |  possible++;
 7 |  |  |  if α ≠ β and neighbors(α) ⊂ β then
 8 |  |  |  |  cliques++;
 9 |  |  |  end
10 |  |  End
11 |  End
12 End
13 return cliques / possible;
```

For each vertex of the graph, it is necessary to check all the combinations of pairs of its neighbors, and whether or not there is an edge that connects them (Algorithm 1). Therefore, the total complexity of this algorithm is $O(|neighbors(v)|^2 - |neighbors(v)|)$ for every vertex v of the graph, considering line 7 can be performed in constant time (O(1)) with the aid of a specific data structure.

Therefore, in the worst case, the complexity is of the order of $O(|V|^3)$, where $|V|$ is the number of vertices in the graph. The worst case is one that considers a complete graph (in which all nodes are connected to all others). In real-world applications of comparative genomics, typically, each gene is bound to at the most a limited number of genes. Thus, we can assume that the number of neighbors for a given gene is limited to $\sqrt{|V|}$. Therefore, the complexity of this algorithm is $O(|V|^* \sqrt{|V|}^* \sqrt{|V|})$, which is equal to $O(|V|^2)$. Moreover, the problem can easily be divided into smaller, independent activities, allowing the parallelization of the algorithm in a very scalable way.

4. METHODS

To take advantage of the characteristics of phylogenetically closely related genomes (such as the organization of coding sequences (CDs) in more homogeneous groups and the fact that a significant part of the groups is expected to be formed by orthologous genes), an algorithm was developed using the average clustering coefficient, where the sum of the coefficients of each vertex is divided by the number of vertices. The higher the value of this coefficient, denser are the components of the graph, and therefore, its maximization produces graphs according to the expected characteristics for a graph of homologies.

The first step of the algorithm is to perform a local alignment, for example, using the BLAST tool, of all sequences against all. A minimum threshold was established and alignments above it are discarded. The threshold used was a maximum e-value of 10^{-10}. In the tested cases, the results improved considerably when a minimum percentage of the length of the alignment (in relation to the size of the sequence) was also defined.

In addition, you can also set limits for other attributes, such as the identity percentage or the maximum number of gaps. The results that satisfy the defined thresholds are transformed into edges for the graph.

The next step is to progressively remove edges that prevent the graph from obtaining the expected topology, as discussed earlier. For this, an e-value between 10^{-10} and 10^{-180} that excludes the edges corresponding to alignments above this value and maximizes the clustering coefficient is chosen. Due to the computational unfeasibility of an algorithm that maximizes this function in a continuous space, it was necessary to define an interval with n integer values between 10 and 180, which in turn generate n e-values (10^{-i}). The computational cost without large modifications in the algorithm would be $O(n * (|V|^2) - n * |V|)$, but with the use of dynamic programming the cost is only $O(|V|^2 - |V| + n)$.

For each one of the tested e-value, the alignments with values greater than this value are removed. Then the groups that are fully connected are separated, and the process is repeated for the remaining groups until a new e-value does not improve the clustering coefficient of the graph. The result at the end of the process is a list of e-values, forming increasingly restrictive layers.

From the biological point of view, the sequences evolve differently. In the first layer the most well-defined sequences are grouped, separating the sequences that are more distant from each other, and in the next ones the separations occur among sequences that have closer relationships.

To take advantage of the characteristics of phylogenetically closely related genomes (such as the organization of CDs in more homogeneous groups and the fact that a significant part of the groups is expected to be formed by orthologous genes), an algorithm was developed using the average clustering coefficient, where the sum of the coefficients of each vertex is divided by the number of vertices (Algorithm 2). The higher the value of this coefficient, the denser are the components of the graph and, therefore, its maximization produces graphs according to the expected characteristics for a graph of homologies.

Algorithm 2: Average graph clustering coefficient calculation

```
Input: g
Input: start
Input: end
list = ?;
while true do
    new = max_{i=start}^{end} (AvgClusteringCoefficient(g, i));
    if new = start then
        return list;
    End
    start = new;
    list ← sub;
    Graph next;
    for sub in components (graph, start) do
        if |nodes(sub)| > 2 & AvgClusteringCoefficient(sub, i) < 1 then
            next ← sub;
        End
    End
    graph = next;
End
```

Two sets of genomes from bacteria were chosen to evaluate the proposed solution. One composed of 55 *S. pyogenes* genomes and the other of 69 *Xanthomonadaceae* genomes, both formed by fully sequenced genomes and selected based on their importance in the fields of medicine (Lamagni et al., 2008; Ferretti et al., 2016) and agronomy (Jalan et al., 2013), respectively. All genomes are available at the National Center for Biotechnology Information and were automatically annotated using the PATRIC tool (Wattam et al., 2017).

Based on the amino acid sequences and their respective annotated functions, the correspondences of the functions in the homologous groups were analyzed. These data were considered classes in a homology classification problem, that is, if two genes were grouped in the same connected component they were classified as positive (P), otherwise as negative (N), and if there was a correspondence between their function then they are classified as true (T), otherwise they are classified as false (F).

This approach allows us to use evaluation metrics widely used in the classification area, such as the following:

- Accuracy: $\frac{TP+TN}{TP+TN+FP+FN}$
- Sensitivity: $\frac{TP}{TP+FP}$
- Specificity: $\frac{TN}{TN+FN}$
- Efficiency: $\frac{Sensitivity}{Specificity}$

Unlike what is common for classification problems, a considerable part of the classes are unknown (between 13% and 27%) and marked as hypothetical proteins. In addition, the annotated functions were not cured by a specialist because of the large volume of data from these sets. Although these limitations do not allow us to accurately assess all cases of homology, it makes possible to automatically evaluate the performance of the classification for the subset of proteins with known functions. Therefore, it was necessary to separate the evaluation of the results into two groups, the first with only sequences with known functions and a second formed by all of them.

The results were compared with the results of a state-of-the-art tool, the TribeMCL, that had excellent performance and had its quality already verified (Brohée and van Helden, 2006).

5. RESULTS

Different alignment percentage lengths were tested to maximize the total number of families present in the core genome. Since the sets of analyzed genomes are quite close, it is expected that there is a broad set of homologous CDSs shared by all genomes.

The distribution of the number of families in the core genome as a function of the percentage size of alignment tested (Fig. 1) indicated a maximum of 38% for the *Streptococcus* group. For the *Xanthomonadaceae* group, the core genome showed to be decreasing as a function of the alignment percentage. The value used in the experiments for this parameter was 30%. Our algorithm organized the genes at an initial layer plus six layers for the set of *Streptococcus* (the e-value threshold for each level was 10^{-14}, 10^{-27}, 10^{-43}, 10^{-46}, 10^{-47}, 10^{-51}, and 10^{-59}) and an initial layer plus four layers for the *Xanthomonadaceae* (the e-value threshold for each level was 10^{-15}, 10^{-23}, 10^{-31}, 10^{-35}, and 10^{-46}); in the last layer were 1275 and 1063 families in the core genome of the respective sets.

The same strategy was used to choose the *inflation* parameter of the TribeMCL, in an exploratory way. Therefore, the following results are based on the inflation of 15.0 for the group of *Streptococcus* (with a core genome of 1237 families) and 10.0 for the group of *Xanthomonadaceae* (with a core genome of 988 families).

The results found for the analyzed algorithms prove to be quite positive, given the complex nature of the problem (Tables 1 and 2). Although the TribeMCL obtains better true positive (TP) values in some cases, this does not necessarily correspond to a better classification as discussed based on the other metrics.

The accuracy results are better for our approach (Figs. 2 and 3), mainly because of the values of true negative (TN) that are the vast majority of instances in this type of problem. This is justified by the density

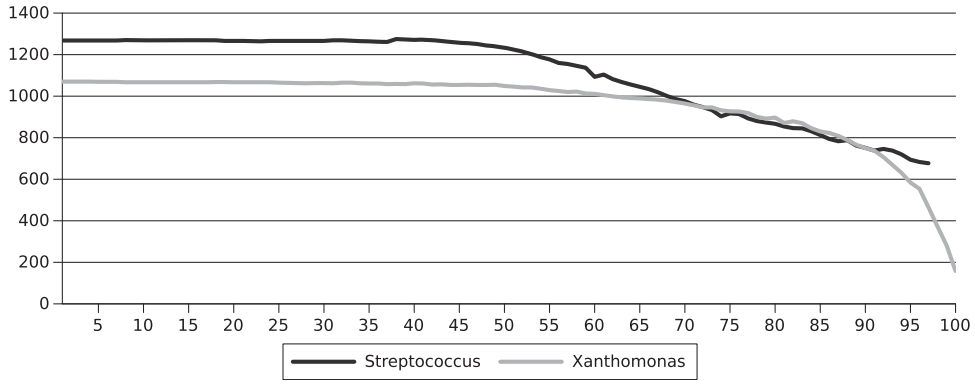

FIG. 1. Number of gene families in the core genome using the proposed algorithm.

Table 1. Classification Results Using the Proposed Algorithm

	Without hypothetical		*With hypothetical*	
	Streptococcus	Xanthomonadaceae	Streptococcus	Xanthomonadaceae
1[a]				
TP	2,610,724	17,215,957	2,874,088	18,163,658
FP	3,281,104	31,994,843	3,307,026	34,657,608
TN	3,874,748,818	25,176,769,293	5,105,439,320	46,961,418,375
FN	7,083,285	63,462,660	11,073,156	768,757,312
2[a]				
TP	2,472,000	13,356,460	2,725,443	14,281,810
FP	840,629	18,884,252	864,292	20,530,341
TN	3,877,189,293	25,189,879,884	5,107,882,054	46,975,545,642
FN	7,222,009	67,322,157	11,221,801	772,639,160
3[a]				
TP	2,462,605	12,340,091	2,715,989	13,258,950
FP	793,497	9,861,963	817,075	10,606,057
TN	3,877,236,425	25,198,902,173	5,107,929,271	46,985,469,926
FN	7,231,404	68,338,526	11,231,255	773,662,020
4[a]				
TP	2,447,485	10,747,632	2,700,869	11,650,042
FP	788,443	6,101,948	812,021	6,636,351
TN	3,877,241,479	25,202,662,188	5,107,934,325	46,989,439,632
FN	7,246,524	69,930,985	11,246,375	775,270,928
5[a]				
TP	2,439,807	N/A	2,693,069	N/A
FP	410,250	N/A	433,731	N/A
TN	3,877,619,672	N/A	5,108,312,615	N/A
FN	7,254,202	N/A	5,108,312,615	N/A
6[a]				
TP	2,403,435	N/A	2,655,104	N/A
FP	329,082	N/A	351,925	N/A
TN	3,877,700,840	N/A	5,108,394,421	N/A
FN	7,290,574	N/A	11,292,140	N/A

FN, false negative; FP, false positive; N/A, not available; TN, true negative; TP, true positive.

of the graph. However, individually the groups are dense, globally the graph is sparse. The classification, including sequences with unknown functions, was better than TribeMCL and this classification only showed less accuracy by mixing sequences of unknown function with the already known ones, giving indications that they could share the same function.

Figures 4 and 5 present the sensitivity results. Although our solution for the set of phylogenetically closest genomes (*Streptococcus*) obtained considerably better results than with TribeMCL, the same did not happen with the more distant genomes (*Xanthomonadaceae*). Due to the fact that at this stage we still do not treat cases with multidomain sequences, many misleading junctions can still be avoided in both sets.

Table 2. Classification Results Using TribeMCL

	Without hypothetical		*With hypothetical*	
	Streptococcus	Xanthomonas	Streptococcus	Xanthomonas
TP	2,510,553	8,804,005	2,787,488	9,773,633
FP	599,795	2,458,627	655,159	2,948,122
TN	3,877,430,127	25,206,305,509	5,108,091,187	46,993,127,861
FN	7,183,456	71,874,612	11,159,756	777,147,337

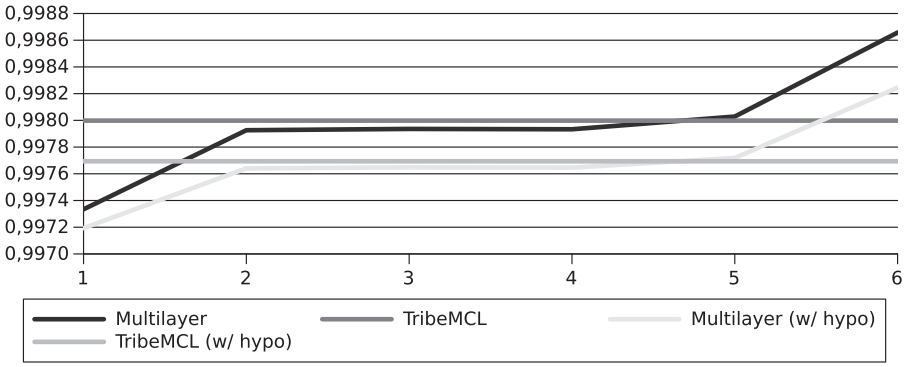

FIG. 2. Overall accuracy for the *Streptococcus pyogenes* genomes.

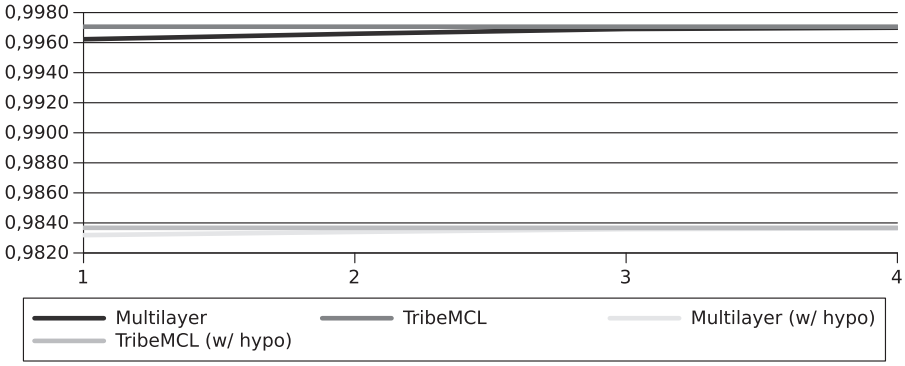

FIG. 3. Overall accuracy for the *Xanthomonadaceae* genomes.

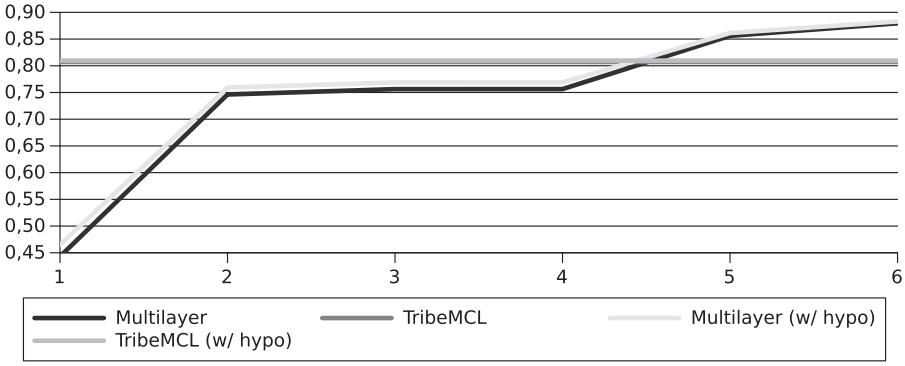

FIG. 4. Sensitivity for the *Streptococcus pyogenes* genomes.

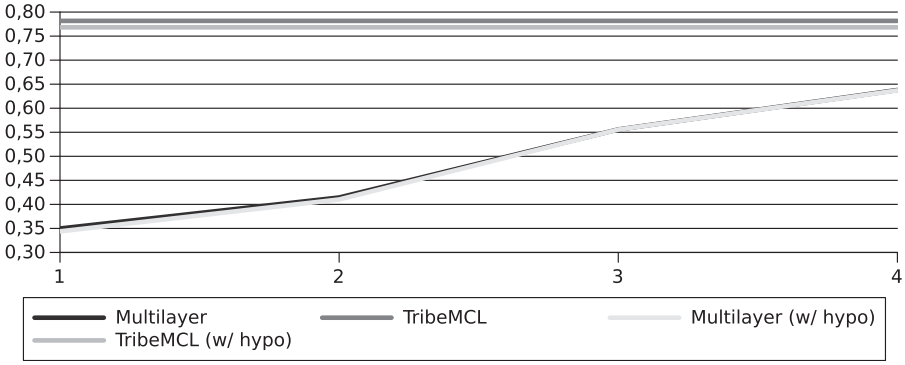

FIG. 5. Sensitivity for the *Xanthomonadaceae* genomes.

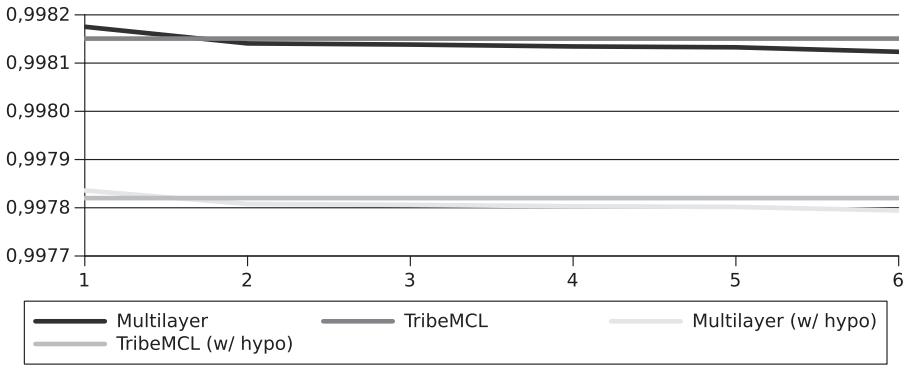

FIG. 6. Specificity for the *Streptococcus pyogenes* genomes.

The differences found in the specificity (Figs. 6 and 7) were small in comparison between the two algorithms (<0.01%), this is due to the large amount of TNs, as well as already seen with accuracy.

Our solution produced better sensitivity results for the set of *Streptococcus* than TribeMCL, something that did not happen with the set of *Xanthomonadaceae* (Figs. 8 and 9). These results do not represent a reliable performance of both algorithms, for reasons previously discussed such as the lack of knowledge of the function of part of the sequences and the lack of curatorship by specialists. However, this experiment helps the understanding of the behavior of a subspace of the problem. In addition, our approach preserves the structure of the graph allowing other topological analyses, such as domain identification. The identification of domains has the potential to further improve peer identification, and shows that even without a defined strategy in this regard, TribeMCL also achieved very good results.

6. IDENTIFICATION OF MOTIFS AND/OR SEQUENCE DOMAINS

Since our approach preserves the relationships between the vertices in the graph, it allows us to make additional analyses regarding the topology. One is the identification of possible domains and motifs, which is highly relevant for genetic studies (Vogel et al., 2004).

Multidomain sequences are a known problem for clustering algorithms because they can be grouped based on local alignments with sequences that do not have homology relationships to each other. This situation is very problematic because multidomain sequences can lead the clustering algorithms to produce groups of nonhomologous sequences. From the point of view of graph theory and topological analysis, these are vertices with smaller clustering coefficients than their neighbors. Therefore, this is the first step in the domain identification process: to identify vertices that have a clustering coefficient smaller than the average of their neighbors. These vertices are marked as possibly multidomains.

Following, the graph goes through a simplification step. The related groups formed of vertices considered here as single domain ones are converted to a single vertex each (a symbolic representation of the group), preserving their edges and the values of their local alignments to other vertices outside the

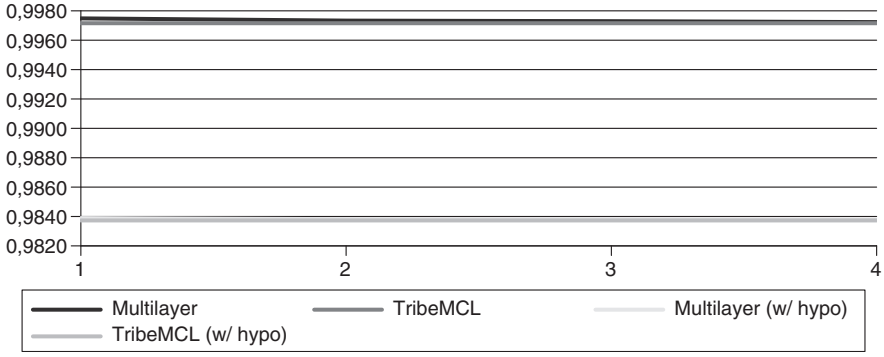

FIG. 7. Specificity for the *Xanthomonadaceae* genomes.

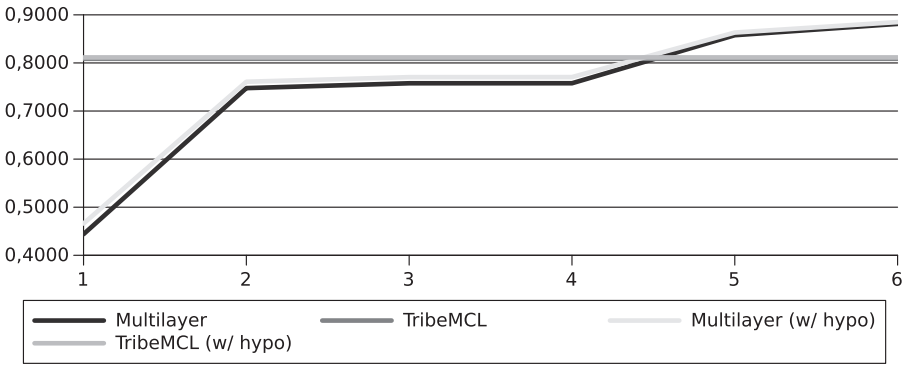

FIG. 8. Efficiency for the *Streptococcus pyogenes* genomes.

group. The same happens with possibly multidomain vertices, and those with the same neighbors in common are converted to a single vertex to represent these vertices.

In the next step, the edges are converted to directed ones. For this, all the edges connected to the vertices that have been converted to the respective vertex are verified, if all the local alignments from a group to another are greater than two defined parameters, then the edge will become directed of the vertex with the smaller sequence to the greater. The two parameters used are based on the difference in length between the two aligned sequences, the first is the absolute value of the difference and the second is the difference divided by the length of the alignment. In the empirical analyses performed, values 100 and 0.3 were found to be adequate to solve the problem and were used to obtain the following results.

The use of a directed graph implies that not all vertices will be accessible from a given beginning (in a connected component). The domains considered are all sets of vertices accessible from all vertices.

The classification of the domains is different from the previous analyses because the groups are not disjoint. For each group, positive, negative, true, and false values are calculated, in which, given a domain, the vertices belonging to the domain are considered true and those that do not belong are considered false. Tables 3 and 4 present the results.

Accuracy and specificity varied <0.5%. And the main advance was in the sensitivity metric and its reflection on the efficiency of the algorithm. The *Streptococcus* group increased from 87.9% to 90.4%, but the increase in *Xanthomonadaceae* was considerable, from 63.7% to 90.2%, directly impacting the efficiency of the classification that increased from 88.1% to 90.6% and from 63.9% to 90.9%, respectively.

7. CONCLUSIONS

This article presents a sequence clustering algorithm based on graph theory. The focus of the algorithm is the CDS of complete genomes of phylogenetically close organisms, which for this particularity have some characteristics of their own: they tend to have a broader core genome and homology relationships tend to be more homogeneous, forming denser components.

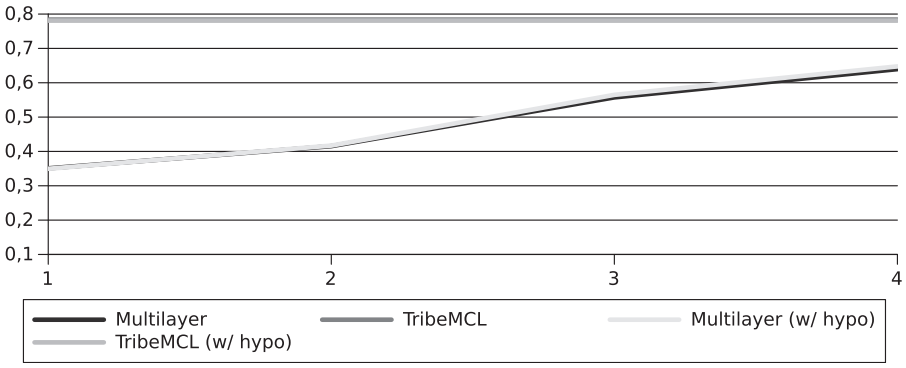

FIG. 9. Efficiency for the *Xanthomonadaceae* genomes.

Table 3. Classification Results Using the Proposed Algorithm Considering Multiple Domains

	Without hypothetical		*With hypothetical*	
	Streptococcus	Xanthomonadaceae	Streptococcus	Xanthomonadaceae
TP	1,898,096	4,313,358	2,655,104	5,058,857
FP	200,631	463,626	351,925	643,403
TN	4,890,956,187	4,374,1697,616	5,108,394,421	75,931,484,135
FN	8,841,324	293,378,177	11,292,140	1,041,731,867

Table 4. Accuracy of the Classification Results Considering Multiple Domains

	Without hypothetical		*With hypothetical*	
	Without domains	*With domains*	*Without domains*	*With domains*
Streptococcus				
Multilayer	0.8795681783	0.9044034789	0.8829658776	0.8829658776
TribeMCL	0.8071614495	0.8071614495	0.9978200412	0.9978200412
Xanthomonadaceae				
Multilayer	0.6378575608	0.9029458755	0.98376896	0.887167018
TribeMCL	0.7817004942	0.7817004942	0.9837315709	0.9837315709

From these principles it was proposed an algorithm that maximizes the clustering coefficient, thus maximizing the density of the connected components of the graph. The result of the algorithm is hierarchical groups in which each layer is more restrictive than the previous one, and thus, by removing edges, the graph reaches the topology with the expected characteristics.

The problem of homologous gene family identification was treated as a problem of homology classification and the results of our solution were compared with the results of the TribeMCL. For both algorithms, two sets of input obtained from phylogenetically close genomes were presented. The algorithms obtained good classification results, considering the complexity of the problem, differing more strongly by the sensitivity metric, in which our algorithm showed better results in the set of the nearest genomes. This metric was negatively influenced by the number of false positives resulting from the set with the more distant genomes.

In addition, we also presented a domain identification algorithm that improved classification. Through the identification of domains, there was an improvement in the sensitivity metric, making the efficiency of the presented algorithm superior to TribeMCL.

As future work we intend to develop algorithms for performing the intracluster analysis to identify the phylogeny of each of the clusters, which may aid in the phylogenetic analysis of groups of closely related genomes.

ACKNOWLEDGMENT

This work was partially funded by CAPES.

AUTHOR DISCLOSURE STATEMENT

The authors declare that no competing financial interests exist.

REFERENCES

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

Apweiler, R., Bairoch, A., Wu, C.H., et al. 2004. UniProt: The Universal Protein knowledgebase. *Nucleic Acids Res.* 32, D115–D119.

Bolten, E., Schliep, A., Schneckener, S., et al. 2001. Clustering protein sequences—structure prediction by transitive homology. *Bioinformatics.* 17, 935–941.

Bork, P., and Koonin, E.V. 1998. Predicting functions from protein sequences—where are the bottlenecks? *Nat. Genet.* 18, 313–318.

Brohée, S., and van Helden, J. 2006. Evaluation of clustering algorithms for protein-protein interaction networks. *BMC Bioinformatics.* 7, 488.

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

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

Ferretti, J.J., Stevens, D.L., and Fischetti, V.A. 2016. *Streptococcus pyogenes: Basic Biology to Clinical Manifestations*. Oklahoma City, OK: University of Oklahoma Health Sciences Center.

Hardison, R.C. 2003. Comparative genomics. *PLoS Biol* 1, E58.

Jalan, N., Kumar, D., Andrade, M.O., et al. 2013. Comparative genomic and transcriptome analyses of pathotypes of *Xanthomonas citri* subsp. *citri* provide insights into mechanisms of bacterial virulence and host range. *BMC Genomics.* 14, 551.

Lamagni, T.L., Darenberg, J., Luca-Harari, B., et al. 2008. Epidemiology of severe *Streptococcus pyogenes* disease in Europe. *J. Clin. Microbiol.* 46, 2359–2367.

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

Pieretti, I., Royer, M., Barbe, V., et al. 2009. The complete genome sequence of *Xanthomonas albilineans* provides new insights into the reductive genome evolution of the xylem-limited *Xanthomonadaceae*. *BMC Genomics.* 10, 616.

Pipenbacher, P., Schliep, A., Schneckener, S., et al. 2002. ProClust: Improved clustering of protein sequences with an extended graph-based approach. *Bioinformatics.* 18, 182–191.

Sasson, O., Linial, N., and Linial, M. 2002. The metric space of proteins—comparative study of clustering algorithms. *Bioinformatics.* 18, 14–21.

Smith, T.F., and Waterman, M.S. 1981. Identification of common molecular subsequences. *J. Mol. Biol.* 147, 195–197.

van Dongen, S. 2000. Graph clustering by flow simulation [Ph.D. dissertation]. University of Utrecht, Utrecht, The Netherlands.

Vogel, C., Bashton, M., Kerrison, et al. 2004. Structure, function and evolution of multidomain proteins. *Curr. Opin. Struct. Biol.* 14, 208–216.

Wattam, A.R., Davis, J.J., Assaf, R., et al. 2017. Improvements to PATRIC, the all-bacterial Bioinformatics Database and Analysis Resource Center. *Nucleic Acids Res.* 45, D535–D542.

Xia, X. 2013. *Comparative Genomics*. Springer Briefs in Genetics. Springer Berlin Heidelberg, Berlin, Heidelberg.

Zdobnov, E.M., Tegenfeldt, F., Kuznetsov, D., et al. 2017. OrthoDB v9.1: Cataloging evolutionary and functional annotations for animal, fungal, plant, archaeal, bacterial and viral orthologs. *Nucleic Acids Res.* 45, D744–D749.

Address correspondence to:
Caio Santiago, MS
Bioinformatics
University of São Paulo
Rua do Matão, 1010
São Paulo 05508-090, Brazil

E-mail: caio.santiago@usp.br