Sequence analysis

High-quality sequence clustering guided by network topology and multiple alignment likelihood

Vincent Miele*, Simon Penel, Vincent Daubin, Franck Picard, Daniel Kahn and Laurent Duret

Laboratoire Biométrie et Biologie Évolutive, Université de Lyon, Université Lyon 1, CNRS, INRA, UMR5558,
Villeurbanne, France

ABSTRACT

Motivation: Proteins can be naturally classified into families of homologous sequences that derive from a common ancestor. The comparison of homologous sequences and the analysis of their phylogenetic relationships provide useful information regarding the function and evolution of genes. One important difficulty of clustering methods is to distinguish highly divergent homologous sequences from sequences that only share partial homology due to evolution by protein domain rearrangements. Existing clustering methods require parameters that have to be set a priori. Given the variability in the evolution pattern among proteins, these parameters cannot be optimal for all gene families.

Results: We propose a strategy that aims at clustering sequences homologous over their entire length, and that takes into account the pattern of substitution specific to each gene family. Sequences are first all compared with each other and clustered into pre-families, based on pairwise similarity criteria, with permissive parameters to optimize sensitivity. Pre-families are then divided into homogeneous clusters, based on the topology of the similarity network. Finally, clusters are progressively merged into families, for which we compute multiple alignment likelihood. To evaluate this method, called HiFiX, we analyzed simulated sequences and manually curated datasets. These tests showed that HiFiX is the only method robust to both sequence divergence and domain rearrangements. HiFiX is fast enough to be used on very large datasets.

Availability and implementation: The Python software HiFiX is freely available at http://lbbe.univ-lyon1.fr/hifix

Supplementary information: Supplementary data are available at Bioinformatics online.

Received on September 16, 2011; revised on February 20, 2012; accepted on February 23, 2012

1 INTRODUCTION

Genomes are the result of a long evolutionary process that began >3 billion years ago. Reconstructing the evolutionary history of genes contained within these genomes is of major interest, not only to uncover the phylogeny of organisms, but also to understand the functioning of living systems. Thanks to the progress of genome sequencing projects, millions of protein-coding genes, from thousands of species, are now available in sequence databases. Several specialized databases have been developed with the aim of providing systematic information about the homology relationships between these sequences, either at the level of protein domains (such as ProDom (Brin et al., 2005) or Plant Dom (Kim et al., 2011)), or among entire proteins, considered as a whole (such as COG (Tatusov et al., 2001), Treefam (Ruan et al., 2006), Ensembl, Compara (Vilella et al., 2008) or HOGENOM (Penel et al., 2009)). The systematic analysis of homology relationships among sequences typically involves two steps: (i) pairwise comparison of all proteins to detect homology (e.g. with BLAST) and (ii) clustering of homologous proteins (or protein domains) into families. The choice of the clustering strategy is crucial for all subsequent analyses, and depends on the purpose of the study. Here, we will discuss specifically clustering strategies that aim at describing homology relationships between entire proteins, not protein domains.

By definition, a family of homologous sequences is a set of sequences that derive from a common ancestor. Hence, in principle, modular proteins containing domains with distinct evolutionary histories should not be included in the same family. The main problem is that the detection of homology by sequence similarity search is subject both to false positive (FP) and false negative (FN) errors. Typically, homology cannot be detected by sequence similarity when proteins have diverged too much. Given that the rate of evolution often varies along proteins, some sequences that are homologous over their entire length may be only locally alignable. Such cases would be erroneously considered as partially homologous, and will lead to FNs. Conversely, it has been shown that similarity search programs, such as BLAST, sometimes tend to extend the local alignment beyond the actual homologous domain (Gonzalez and Pearson, 2004). Thus, in some cases, modular proteins sharing only partial homology can be aligned over their entire length, and hence be classified in the same family (FPs, see Fig. 1). The rate of FPs can be decreased by using more stringent sequence similarity criteria, but this necessarily leads to an increase in the rate of FNs.

All clustering methods have some parameters that can be tuned to optimize the tradeoff between sensitivity and specificity. However, the choice of these parameters is totally empirical, and the default parameters that are proposed by the authors generally reflect the result of trial and error on a limited benchmark set of sequences (Panchan et al., 2011; Wittkop et al., 2011). One other problem
overextension can also be found with natural protein sequences (Gonzalez 2009). BLAST alignment extends over the entire length of both proteins. Such an overextension can also be found with natural protein sequences (Gonzalez and Pearson, 2010).

is that the tempo of sequence evolution is highly variable among proteins, and hence optimal parameters may vary among families. Moreover, the optimal parameters may also vary according to the size of the sequence dataset. For example, we recently developed an algorithm that extends over the entire length of both proteins. Such an overextension can also be found with natural protein sequences (Gonzalez and Pearson, 2010).

For the sake of simplicity, we now consider a single pre-family retrieved by STIX but it is straightforward to independently apply the following approach to all the pre-families. We consider the list of pairs of similar sequences that satisfy alignment coverage constraints. The strategy is based on the following ideas: (1) Hierarchical clustering of pre-families into homogeneous sequence clusters, and then for the progressive merging of clusters into families.

2.1 Search for homogeneous clusters of homologous sequences as communities in networks

To circumvent these problems, we propose here a new clustering strategy. The main idea is that the decision to include or not a protein in a given family should be based on the examination of multiple sequence alignments, not simply on the analysis of pairwise sequence similarities. Indeed, a multiple alignment contains information about the mode and tempo of evolution at each amino-acid position of a given protein family, which can be used to decide whether or not a new sequence belongs to that family. The strategy we propose consists in three steps:

(1) Rapid clustering with STIX, using low-stringency criteria, to get a first set of pre-families, with a low rate of FNs (but possibly including FPs).

(2) Decomposition of each pre-family into homogeneous protein clusters, by analysing the topology of similarity networks.

(3) Hierarchical clustering of previous clusters into families, by progressive multiple alignment of protein clusters and evaluation of alignment quality at each step.

In this article, we describe in details the methods used in the last two steps. We tested this procedure on several well-studied sets of biological sequences, and on a set of simulated sequences. These tests show a significant improvement over existing methods, notably for large and highly divergent protein families. The method, called HiFiX, is fast enough to be used in practice for very large datasets. HiFiX is available as a Python software at http://lcbi.univ-lyon1.fr/hifix.

2 METHODS

Similarity relationships can be modeled through graph theory representation, with sequences as vertices and similarities as edges forming a similarity network. Analyzing the topology of this network was proved to be relevant for studying protein or domain homology (Medini et al. 2009; Zhang et al. 2010). In the following, we propose to consider the topology as a guide, first to subdivide each pre-family into homogeneous sequence clusters, and then for the progressive merging of clusters into families.

2.1 Search for homogeneous clusters of homologous sequences as communities in networks

To circumvent these problems, we propose here a new clustering strategy. The main idea is that the decision to include or not a protein in a given family should be based on the examination of multiple sequence alignments, not simply on the analysis of pairwise sequence similarities. Indeed, a multiple alignment contains information about the mode and tempo of evolution at each amino-acid position of a given protein family, which can be used to decide whether or not a new sequence belongs to that family. The strategy we propose consists in three steps:

(1) Rapid clustering with STIX, using low-stringency criteria, to get a first set of pre-families, with a low rate of FNs (but possibly including FPs).

(2) Decomposition of each pre-family into homogeneous protein clusters, by analysing the topology of similarity networks.

(3) Hierarchical clustering of previous clusters into families, by progressive multiple alignment of protein clusters and evaluation of alignment quality at each step.

In this article, we describe in details the methods used in the last two steps. We tested this procedure on several well-studied sets of biological sequences, and on a set of simulated sequences. These tests show a significant improvement over existing methods, notably for large and highly divergent protein families. The method, called HiFiX, is fast enough to be used in practice for very large datasets. HiFiX is available as a Python software at http://lcbi.univ-lyon1.fr/hifix.

2 METHODS

Similarity relationships can be modeled through graph theory representation, with sequences as vertices and similarities as edges forming a similarity network. Analyzing the topology of this network was proved to be relevant for studying protein or domain homology (Medini et al. 2009; Zhang et al. 2010). In the following, we propose to consider the topology as a guide, first to subdivide each pre-family into homogeneous sequence clusters, and then for the progressive merging of clusters into families.

2.1 Search for homogeneous clusters of homologous sequences as communities in networks

To circumvent these problems, we propose here a new clustering strategy. The main idea is that the decision to include or not a protein in a given family should be based on the examination of multiple sequence alignments, not simply on the analysis of pairwise sequence similarities. Indeed, a multiple alignment contains information about the mode and tempo of evolution at each amino-acid position of a given protein family, which can be used to decide whether or not a new sequence belongs to that family. The strategy we propose consists in three steps:

(1) Rapid clustering with STIX, using low-stringency criteria, to get a first set of pre-families, with a low rate of FNs (but possibly including FPs).

(2) Decomposition of each pre-family into homogeneous protein clusters, by analysing the topology of similarity networks.

(3) Hierarchical clustering of previous clusters into families, by progressive multiple alignment of protein clusters and evaluation of alignment quality at each step.

In this article, we describe in details the methods used in the last two steps. We tested this procedure on several well-studied sets of biological sequences, and on a set of simulated sequences. These tests show a significant improvement over existing methods, notably for large and highly divergent protein families. The method, called HiFiX, is fast enough to be used in practice for very large datasets. HiFiX is available as a Python software at http://lcbi.univ-lyon1.fr/hifix.
Algorithm 1: HiFiX — High Fidelity Clustering of Sequences

**Input:** $K$ clusters as communities
1. perform $K$ multiple alignments
2. build profile-HMM models $M_q, 1 \leq q \leq K$
3. compute $ICL(K)$
4. for $Q = K - 1$ to 1 do
5. merge two clusters $q$ and $l$ with highest $\tau_{ql}$
6. perform alignment of alignments $q$ and $l$
7. build profile-HMM models $M_q, 1 \leq q \leq Q$
8. calculate $ICL(Q)$
9. update $\pi$
10. end for

**Output:** $Q^* = \text{argmax}_Q \text{ICL}(Q)$

Because $\pi$ is used as a guide in the algorithm but not directly related to likelihoods, we add the possibility to examine the $J = 1$ highest values of $\pi$ as basis of Algorithm [3]. Following the strategy presented in [Han et al., 2008], we select the merging that minimizes the log-likelihood loss per sequence.

### 2.3 The HiFiX software package

The presented algorithm is implemented in the HiFiX software program, written in Python and designed to be executed in parallel on multiprocessor architectures. In HiFiX, community detection is performed with the Louvain algorithm, available as a software package from [http://sites.google.com/site/findcommunities/](http://sites.google.com/site/findcommunities/). Multiple alignments are computed with [mafft](http://mafft.cbrc.jp/alignment/software/) and [mafft-profile](http://mafft.cbrc.jp/alignment/profile/) with default parameters. Log-likelihoods in Formula (ICL) are derived from Hidden Markov Models (HMMs) using [HMMER v3.0](http://hmmer.wustl.edu) as follows: (i) HMMs are constructed using [ hmmbuild](http://hmmer.wustl.edu) with the -e option in order to avoid rescaling of log-likelihood ratios; (ii) ICL scores of all sequences in each community are obtained with [farmsearch](http://wjdug.etna.rice.edu/software/farmsearch) and summed to give log-likelihood ratios; and (iii) changes in log-likelihood ratios are taken as changes in log-likelihoods because the background model is constant. Finally, at each iteration of Algorithm [3] we probe the $J = 2$ highest values of $\pi$.

HiFiX requires the use of [SILIX](http://bioinformatics.oxfordjournals.org/content/31/15/1078) [Miele et al., 2012], default parameters recommended) as a preliminary step. HiFiX is licensed under the General Public License [http://www.gnu.org/licenses/](http://www.gnu.org/licenses/). This software is available as a freely downloadable package from [http://www.bioinformatics.org](http://www.bioinformatics.org).

### 2.4 Other program parameters

To perform our experiments, we ran BLASTP with the following options:

- `--soft-seq 3.1` `--e-value 0.005` `--b 10000`
- `--soft-seq 3.1` `--e-value 0.001` `--b 10000`

**ICL** [Enright et al., 2002] was used with default parameters. The **TransClust** threshold parameter was set at 55 as suggested by [Winkop et al., 2011]. Maximal cluster size for [hcluster-sg](http://bioinformatics.oxfordjournals.org/content/31/15/1078) was set as the total number of proteins in each family.

We performed simulation of protein families using the [INDELible](http://www.evolveo.cz/) program with default settings, the WAG substitution model, a continuous gamma rate heterogeneity with $\alpha = 1$ and insertion rates of 0.0005 and 0.0001 for the low and high divergence case, respectively.

### 3 RESULTS

To evaluate the results of HiFiX and compare them to those of other methods, we used several benchmark sets of known homologous protein families. On each set of sequences, we first compared all proteins against all with BLASTP [Altschul et al., 1997]. The BLAST results were then analyzed with different clustering methods.

For each method, each reference family of the benchmark set was compared with the results of the clustering to identify the cluster with the highest number of corresponding sequences (target cluster).

Each sequence of the reference family that was absent from the target cluster was counted as a FN. Each sequence of the target cluster that was present or absent from the reference family was counted as a true positive (TP) or a FP, respectively. We computed sensitivity TP/(TP+FN) and specificity TP/(TP+FP) for each cluster and report their values averaged over all reference families. We also report F-measure II, a weighted harmonic mean between sensitivity and specificity.

**Fig. 2.** Example of a community structure in the similarity network corresponding to HBG79407. HOG51530MM, families retrieved by SILIX. Network was drawn with Cytoscape [Shannon et al., 2000].
We first used a reference dataset published by (Brown TCRR) corresponds to the 4-th largest family in the HOGENOM. Therefore this benchmark is a relatively easy case for clustering. HiFiX performs very well (Table 1a). The best results by (Wittkop et al. 2010) for a recent update. This set of sequences (hereafter called SiLiX, generated by HiFiX, used alone, or followed by the two additional HiFiX steps: step 2: decomposition of pre-families into communities with Louvain; step 3: hierarchical clustering into meta-communities that are the final families. To quantify the gain due to the third step, we also analyzed the results of the first two steps, i.e. SiLiX followed by Louvain, assuming that the found communities are the protein families.

For a comparison, we also performed the clustering with several other methods that were recently evaluated among the best currently available: MCL (Enright et al., 2002), hcluster_sg (Ruan et al., 2006); TransClust (Wittkop et al. 2010) and TransClust (Bru et al., 2005) and has been very well studied. Indeed an analysis of domain decomposition of its member proteins with ProDom (Bru et al., 2005) indicates 10 distinct protein types. HiFiX, retrieved six families (Fig. 3) and performs much better than other methods (Table 1b). Compared with SiLiX, HiFiX is much more specific without losing much sensitivity. MCL performs quite well on this test set, but with a relatively high number of FPs compared with HiFiX. MCL gave many clusters, and notably clustered most single domain CheY-type proteins together with two-domain OmpR-type proteins, while other CheY-type proteins were scattered into a dozen clusters. hcluster_sg behaved similarly. The results of TransClust on this test set were poor in terms of sensitivity.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>HiFiX</td>
<td>96</td>
<td>0.94 0.91 0.90</td>
</tr>
<tr>
<td>SiLiX</td>
<td>99</td>
<td>0.94 0.91 0.90</td>
</tr>
<tr>
<td>Louvain</td>
<td>100</td>
<td>0.95 0.79 0.82</td>
</tr>
<tr>
<td>MCL</td>
<td>47</td>
<td>0.84 0.98 0.87</td>
</tr>
<tr>
<td>TransClust</td>
<td>96</td>
<td>0.95 0.90 0.91</td>
</tr>
<tr>
<td>hcluster_sg</td>
<td>27</td>
<td>0.66 1.00 0.75</td>
</tr>
<tr>
<td>(a) on the Brown et al. (2006) benchmark of 866 enzymes Wittkop et al. (2010)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(b) on bacterial TCRRs (TCRR family)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Spec: specificity; Sens: sensitivity; F: F-measure II. F values >0.85 are in bold.

3.1 Evaluation of clustering methods on known protein families

We first used a reference dataset published by Brown et al. (2006) containing 866 enzymes that were manually assigned to 91 protein families, and that was recently used to compare clustering methods by Wittkop et al. (2010). On this test set, all methods except hcluster_sg perform very well (Table 1a). The best results are obtained with TransClust, SiLiX and HiFiX. The results of HiFiX are identical to those of SiLiX because families generated by SiLiX are homogeneous on this dataset. It should be noted that these protein families are relatively small: the largest family contains only 215 sequences (average of 10 sequences per family). As discussed in the introduction, the risk of FPs due to alignment over-extension increases with the size of the family. Therefore this benchmark is a relatively easy case for clustering methods.

To evaluate the results of these methods on larger protein families, we analyzed a set of 14 260 bacterial TCRRs containing a conserved receiver domain and divergent output domains (see Galperin, 2010 for a recent update). This set of sequences (hereafter called TCRR) corresponds to the 4-th largest family in the HOGENOM database (Pascareno et al. 2004) and has been very well studied. However its large size makes it more difficult to process correctly for clustering algorithms such as SiLiX and hence the resulting family is heterogeneous in HOGENOM (accession number HBG753323).

Indeed an analysis of domain decomposition of its member proteins with ProDom (Bru et al., 2005) indicates 10 distinct protein types. HiFiX, retrieved six families (Fig. 3) and performs much better than other methods (Table 1b). Compared with SiLiX, HiFiX is much more specific without losing much sensitivity. MCL performs quite well on this test set, but with a relatively high number of FPs compared with HiFiX. MCL gave many clusters, and notably clustered most single domain CheY-type proteins together with two-domain OmpR-type proteins, while other CheY-type proteins were scattered into a dozen clusters. hcluster_sg behaved similarly. The results of TransClust on this test set were poor in terms of sensitivity.

3.2 Evaluation of clustering methods with simulated protein families

In order to assess the effects of sequence divergence and protein modularity on the behavior of HiFiX, we tested it further on a dataset of 866 enzymes (Wittkop et al., 2010) [see details in Supplementary Material and also (Wittkop et al., 2010)]. This procedure was used to evaluate performance compared with other clustering programs. The results of these methods on larger protein families, as well as the results on smaller families, are summarized in Table 1a. The table shows that HiFiX performs quite well on this test set, but with a relatively high number of FPs compared with HiFiX. MCL gave many clusters, and notably clustered most single domain CheY-type proteins together with two-domain OmpR-type proteins, while other CheY-type proteins were scattered into a dozen clusters. hcluster_sg behaved similarly. The results of TransClust on this test set were poor in terms of sensitivity.

Table 1. HiFiX performance compared with other clustering programs

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>HiFiX</td>
<td>96</td>
<td>0.94 0.91 0.90</td>
</tr>
<tr>
<td>SiLiX</td>
<td>99</td>
<td>0.94 0.91 0.90</td>
</tr>
<tr>
<td>Louvain</td>
<td>100</td>
<td>0.95 0.79 0.82</td>
</tr>
<tr>
<td>MCL</td>
<td>47</td>
<td>0.84 0.98 0.87</td>
</tr>
<tr>
<td>TransClust</td>
<td>96</td>
<td>0.95 0.90 0.91</td>
</tr>
<tr>
<td>hcluster_sg</td>
<td>27</td>
<td>0.66 1.00 0.75</td>
</tr>
<tr>
<td>(a) on the Brown et al. (2006) benchmark of 866 enzymes Wittkop et al. (2010)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(b) on bacterial TCRRs (TCRR family)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Spec: specificity; Sens: sensitivity; F: F-measure II. F values >0.85 are in bold.

---

Fig. 3. Correspondence between manually built clusters with domain arrangements from ProDom (Bru et al., 2005) and HiFiX clusters on bacterial TCRRs (TCRR family).
set of artificially evolved sequences. We simulated the evolution of protein sequences (250 amino-acid long), subject both to progressive divergence by accumulation of point mutations and to modular rearrangements. The progressive divergence was simulated using the tool INDELible (Fletcher and Yang, 2009), starting from a random ancestral sequence, and following a true phylogeny (based on 23S rRNA sequences) of 536 bacterial species (Fig. 4). We modulated the rate of sequence divergence by controlling the tree depth between 1.33 and 4.5 substitutions per site (which corresponds to an average of 32% identity between sequences from the outgroup and from the ingroup clades). Domain rearrangements were performed by the replacement of the C-terminal end of the protein by a non-homologous sequence: for a given clade of simulated sequences, i.e. deriving from a common ancestor, we exchanged the C-terminal end of each sequence with those of a different set of simulated sequences, species by species (Fig. 5). In our simulations, we introduced three independent events of domain rearrangements (i.e. we assumed that the rate of domain rearrangement was relatively low). We modulated the relative size of the rearranged C-terminal domain (x) from x = 0% to x = 40% of the protein length. Thus, when x = 0%, the whole set of simulated sequences corresponds to one single family of proteins, homologous over their entire length. When x > 0%, simulated sequences correspond to four different families: the three clades that derive from the nodes where the domain rearrangements occurred, and the rest of the sequences, that conserved the ancestral domain architecture. For each set of parameters (protein divergence rate, length of rearranged domain) we performed 20 simulations.

When protein divergence is low (average of 1.33 substitutions per site), both TransClust and HiPiX perform very well (Table 4). hcluster_sg, MCL and SiLiX give good results in the absence of domain rearrangement (x = 0%), but perform poorly when x > 0% because they tend to cluster in the same family sequences with distinct domain architectures. In the case of SiLiX, this behavior is clearly due to over-extension of BLAST alignments beyond the homologous domain (e.g. see Figure 4), which leads to FP similarity links.

When protein divergence is high (average of 4.5 substitutions per site), HiPiX clearly outperforms other methods. The sensitivity of Transclust is very low with this set: as with the TCRR dataset analyzed above, it tends to create a large number of small homogeneous clusters. MCL works well for x = 0% but is not robust to domain rearrangements. In conclusion HiPiX is the only method that remains robust both to sequence divergence and to domain rearrangements. We also performed simulations to test the impact of various factors on HiPiX clustering results (protein sequence length, unbalanced phylogenetic sampling, presence of proteins homologous to the shuffled domains). These analyses showed that HiPiX results are not affected by these factors (see Supplementary Material).

### 3.3 Impact of SiLiX parameters

Steps 2 and 3 of the HiPiX procedure do not require any parameter. However, the sensitivity of HiPiX depends on the parameters that are used by SiLiX in the first step to cluster sequences into families. SiLiX requires two parameters: the minimal percentage of sequence identity x and the alignment coverage c, i.e. the fraction of protein length that is covered by the pairwise sequence alignment. We tested the impact of these parameters on HiPiX results (Table 5 and Supplementary Table S4). The SiLiX step is required to exclude from the similarity network (which will be analyzed by Louvain at the next step) all edges that only correspond to domain homology. We observed that if we do not set any constraint on the alignment coverage (c = 0%), then this leads to the clustering of sequences that are not homologous over their entire length (data not shown). However, HiPiX performances on our simulated dataset remain essentially unchanged for a wide range of relevant c coverage parameter (from c = 60% to c = 90%, Supplementary Table S4). As expected, the use of stringent similarity parameter (x ≥ 40%) leads to a decrease in sensitivity. Indeed, it is necessary to use SiLiX criteria that are permissive enough to avoid the exclusion of relevant homology relationships at the first step. The use of more permissive SiLiX criteria requires more computing time (intrinsically due to the larger number of multiple alignments that have to be computed at step 3) but leads to a strong gain in sensitivity, with only limited loss in specificity for HiPiX. This indicates
While the main advantage of HiFiX is the quality of the clusters it generates, the associated computational expense is affordable even for very large datasets. Indeed the Louvain algorithm can handle huge networks with millions of vertices in reasonable time [see Blondel et al. (2008)]: finding communities can usually be achieved in only a few seconds. Second, mafft performs K multiple alignments on protein subsets and mafft-profile fewer than \(I(K-1)\) alignments of alignments, so the overall alignment task scales linearly with \(K\). Moreover mafft and mafft-profile present an excellent trade-off between speed and accuracy (Katoh et al. 2002). Finally likelihood calculations rely on the very efficient HMMER3 package (Eddy 2007). Altogether this makes HiFiX extremely efficient. Hence, using a single core Intel Xeon 3.07 GHz, we clustered the 536 simulated sequences in 5 s, the 866 enzymes in 80 s, the 8 million edges in the similarity network in \(~75\) min. As a benchmark, we analyzed a very large set of 3 206 033 sequences (extracted from the HOGENOM database version 6) displaying numerous pre-families with a power law distribution of size between 50 and 23 327 families. Taking advantage of multiprocessing to treat them in parallel on 24 cores Opteron 2.2 GHz, HiFiX required \(~13 700\) min of calculation corresponding to \(<10\) h elapsed time.

### 4 DISCUSSION

Proteins are the result of an evolutionary process, and can therefore naturally be classified into families of homologous sequences. We propose here a clustering method (HiFiX) that relies on three steps: (i) permissive clustering of sequences in pre-families; (ii) sub-clustering of pre-families into homogeneous clusters; and (iii) progressive merging of clusters into families, with evaluation of the quality of the multiple alignment at each step. The goal of this strategy is to maximize sensitivity (i.e. families should be as exhaustive as possible), while preserving the quality of the final multiple alignment (i.e. two clusters can be merged, as long as they are homologous over their entire length). The logics behind this strategy is to cluster all homologous sequences that are similar enough to obtain a reliable multiple alignment, from which it will be possible to construct a phylogenetic tree. We evaluated the quality of the clustering with several manually curated or simulated benchmark datasets. As expected, the clustering obtained after step 2 (Louvain) shows a higher specificity but lower sensitivity than after step 1 (SiLiX) (Tables 1 and 2). Interestingly, step 3 leads to recover a sensitivity almost as good as SiLiX, while preserving the specificity.

### Table 2. Average performance of clustering programs on simulated datasets of evolving modular proteins

<table>
<thead>
<tr>
<th>Method</th>
<th>x = 40%</th>
<th></th>
<th>x = 25%</th>
<th></th>
<th>x = 0%</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Low divergence (d = 1.33)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HiFiX</td>
<td>1.00</td>
<td>1.00</td>
<td>0.78</td>
<td>1.00</td>
<td>0.88</td>
<td>1.00</td>
</tr>
<tr>
<td>SiLiX</td>
<td>0.29</td>
<td>1.00</td>
<td>0.29</td>
<td>1.00</td>
<td>0.44</td>
<td>1.00</td>
</tr>
<tr>
<td>Louvain</td>
<td>1.00</td>
<td>0.98</td>
<td>0.91</td>
<td>1.00</td>
<td>0.94</td>
<td>1.00</td>
</tr>
<tr>
<td>MCL</td>
<td>0.29</td>
<td>1.00</td>
<td>0.29</td>
<td>1.00</td>
<td>0.44</td>
<td>1.00</td>
</tr>
<tr>
<td>TransClust</td>
<td>1.00</td>
<td>0.97</td>
<td>0.96</td>
<td>0.98</td>
<td>0.96</td>
<td>1.00</td>
</tr>
<tr>
<td>helcluster</td>
<td>0.29</td>
<td>1.00</td>
<td>0.29</td>
<td>1.00</td>
<td>0.44</td>
<td>1.00</td>
</tr>
<tr>
<td>High divergence (d = 4.5)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HiFiX</td>
<td>1.00</td>
<td>0.95</td>
<td>0.97</td>
<td>0.93</td>
<td>0.94</td>
<td>0.92</td>
</tr>
<tr>
<td>SiLiX</td>
<td>0.95</td>
<td>0.95</td>
<td>0.94</td>
<td>0.62</td>
<td>0.95</td>
<td>0.71</td>
</tr>
<tr>
<td>Louvain</td>
<td>1.00</td>
<td>0.65</td>
<td>0.68</td>
<td>1.00</td>
<td>0.78</td>
<td>0.77</td>
</tr>
<tr>
<td>MCL</td>
<td>0.68</td>
<td>0.98</td>
<td>0.74</td>
<td>0.39</td>
<td>0.99</td>
<td>0.52</td>
</tr>
<tr>
<td>TransClust</td>
<td>0.29</td>
<td>0.44</td>
<td>0.29</td>
<td>1.00</td>
<td>0.42</td>
<td>1.00</td>
</tr>
<tr>
<td>helcluster</td>
<td>0.41</td>
<td>1.00</td>
<td>0.55</td>
<td>0.31</td>
<td>1.00</td>
<td>0.46</td>
</tr>
</tbody>
</table>

Divergence is controlled by the guide tree depth \(d\) which is expressed in number of substitutions per site. \(x\) corresponds to the length fraction of heterologous domains (see text for details). Abbreviations are the same as in Table 1.

### Table 3. Average performance of HiFiX on simulated datasets of modular proteins (x = 40%) with SiLiX parameters set to varying percentages of sequence identity \(x\) and to a fixed alignment coverage \(c = 80\%

<table>
<thead>
<tr>
<th>Method</th>
<th>x = 40%</th>
<th></th>
<th>x = 25%</th>
<th></th>
<th>x = 0%</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Low divergence (d = 1.33)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SiLiX</td>
<td>0.29</td>
<td>1.00</td>
<td>0.43</td>
<td>0.91</td>
<td>0.94</td>
<td>0.94</td>
</tr>
<tr>
<td>HiFiX</td>
<td>0.85</td>
<td>1.00</td>
<td>0.90</td>
<td>0.95</td>
<td>0.94</td>
<td>0.94</td>
</tr>
<tr>
<td>SiLiX</td>
<td>0.45</td>
<td>1.00</td>
<td>0.58</td>
<td>0.95</td>
<td>0.95</td>
<td>0.95</td>
</tr>
<tr>
<td>HiFiX</td>
<td>0.91</td>
<td>1.00</td>
<td>0.94</td>
<td>0.95</td>
<td>0.95</td>
<td>0.95</td>
</tr>
<tr>
<td>SiLiX (default)</td>
<td>0.91</td>
<td>1.00</td>
<td>0.94</td>
<td>0.95</td>
<td>0.95</td>
<td>0.95</td>
</tr>
<tr>
<td>HiFiX</td>
<td>1.00</td>
<td>0.95</td>
<td>0.97</td>
<td>1.00</td>
<td>0.80</td>
<td>0.88</td>
</tr>
<tr>
<td>SiLiX</td>
<td>1.00</td>
<td>0.80</td>
<td>0.88</td>
<td>1.00</td>
<td>0.39</td>
<td>0.54</td>
</tr>
<tr>
<td>HiFiX</td>
<td>1.00</td>
<td>0.80</td>
<td>0.88</td>
<td>1.00</td>
<td>0.39</td>
<td>0.54</td>
</tr>
</tbody>
</table>

Details and abbreviations are the same as in Table 2.

that step 3 is efficient to reject FPs. Interestingly, HiFiX results remain stable for a relatively wide range of SiLiX parameters (in Table 1, from \(x = 25\%\) to \(x = 40\%\)). Thus it appears that it is not necessary to use extremely permissive SiLiX criteria to obtain optimal results. Based on these simulations, we recommend using the default parameters of SiLiX (\(x = 35\%\) and \(c = 80\%\)).

### 3.4 Computational footprint

While the main advantage of HiFiX is the quality of the clusters it generates, the associated computational expense is affordable even for very large datasets. Indeed the Louvain algorithm can handle huge networks with millions of vertices in reasonable time [see Blondel et al. (2008)]: finding communities can usually be achieved in only a few seconds. Second, mafft performs multiple alignments on protein subsets and mafft-profile fewer than \(I(K-1)\) alignments of alignments, so the overall alignment task scales linearly with \(K\). Moreover mafft and mafft-profile present an excellent trade-off between speed and accuracy (Katoh et al. 2002). Finally likelihood calculations rely on the very efficient HMMER3 package (Eddy 2007). Altogether this makes HiFiX extremely efficient. Hence, using a single core Intel Xeon 3.07 GHz, we clustered the 536 simulated sequences in 5 s, the 866 enzymes in 80 s, the 8 million edges in the similarity network in \(~75\) min. As a benchmark, we analyzed a very large set of 3 206 033 sequences (extracted from the HOGENOM database version 6) displaying numerous pre-families with a power law distribution of size between 50 and 23 327 families. Taking advantage of multiprocessing to treat them in parallel on 24 cores Opteron 2.2 GHz, HiFiX required \(~13 700\) min of calculation corresponding to \(<10\) h elapsed time.
the specificity obtained at the second step with Louvain (Tables 4 and 5). The enzyme benchmark set from Brown et al. [1000] and the simulated set without modular evolution (x = 0) correspond to relatively easy cases, where most tested methods, including SiLiX, give very good results. On these families, HiFiX performs as well as SiLiX, which indicates that steps 2 and 3 do not deteriorate the quality of the clustering. The more difficult cases consist of families of multi-domain proteins that are not homologous over their entire length (the TCRR family and the simulated sets with x > 0). HiFiX turned out to be the only method that is robust to such situations of modular evolution. We think that the main reason for these good results is that contrarily to other methods, that are all based on the analyses of pairwise sequence similarities, HiFiX uses information from the multiple alignment to decide whether a sequence should be included in a family or not. This step results in good specificity and therefore allows using more permissive parameters at the first step, which ensures good sensitivity. It should be noted that the results of the other tested methods might have been improved by tuning their parameters (here we used default parameters provided by the authors). However, as mentioned in the introduction, the optimal clustering parameters for a given set of families may in fact not be optimal for other families. Indeed, one important property of the HiFiX method is that it does not require setting a priori the same parameters for all families: the decision to include or not a sequence in a family is based on the analysis of the multiple alignment, and hence takes into account the pattern of evolution specific of that family. Of course, HiFiX sensitivity depends on the parameters used for step 1 (SiLiX). However, as long as SiLiX parameters are permissive enough, HiFiX results remain essentially unaffected.

HiFiX is designed to cluster sequences that are homologous over their entire length, by contrast to modular proteins that only share some homologous domains. HiFiX is therefore not appropriate to detect homology relationships among proteins that are only partially alignable. The choice of this relatively strict clustering criterion is primarily motivated by pragmatic considerations. Indeed, HiFiX was developed to be used for large scale phylogenomic studies, and notably for the construction of the HOGENOM database, which provides phylogenetic trees for thousands of gene families [Penel et al. [2009]]. To be able to reconstruct the evolutionary relationships among homologous sequences, it is necessary to have a multiple alignment of good quality. Allowing the clustering of partially homologous proteins inevitably leads to low quality multiple alignments, which therefore produce erroneous phylogenetic trees. Of course, such cases could be corrected by manual expertise, but this cannot be done systematically for large scale phylogenomic analyses. An alternative approach would be to cluster homologous domains, instead of entire homologous proteins. The problem is that domains are generally short, and this strategy therefore leads to a loss of phylogenetic signal. The HiFiX clustering strategy thus reflects a tradeoff between the quantity and quality of phylogenetic information that can be recovered from large scale comparative sequence analyses. It should be noted however that despite this intrinsic limitation, HiFiX performs at least as well as other methods on manually curated benchmark sets (Table 5). This is most probably because these methods, although they do not set a priori constraints on the alignment length, have to use relatively strict parameters to limit the rate of FPs.

In conclusion, HiFiX provides a significant improvement in clustering quality over other existing methods, notably on multi-domain protein families, which represent a large fraction of all sequences. The HiFiX procedure is fast enough to be used in practice on very large sequence data sets.

ACKNOWLEDGEMENTS

The authors would like to give special thanks to David Robelin for his inspiration, and would like to thank Bruno Spataro and colleagues for the computing facilities, and also Thomas Bernard, Catherine Matias, Bernad Prum and lastly Tobias Witkop for answers about the program. We also thank three anonymous reviewers for their constructive comments.

Funding: French Agence Nationale de la Recherche under grants NeMo ANR-08-BLAN-0304-01 and ANCESTROME ANR-10-BINF-01-01; and European Union under grant IMPACT FP7-RI-213037.

Conflict of Interest: none declared.

REFERENCES


Miele, V. et al. (2011) Ultra-fast sequence clustering from similarity networks with SiLiX. BMC Bioinformatics, 12, 116.


Paul, S. et al. (2009) Databases of homologous gene families for comparative genomics. BMC Bioinformatics, 10 (Suppl. 6), S3.

Picard, F. et al. (2009) Deciphering the connectivity structure of biological networks using MixNet. BMC Bioinformatics, 10 (Suppl. 6), S17.


