GEDEVO: An Evolutionary Graph Edit Distance Algorithm for Biological Network Alignment

Rashid Ibragimov[1], Maximilian Malek[1], Jiong Guo[2], and
Jan Baumbach[1,3]

1 **Computational Systems Biology Group, Max-Planck-Institut für Informatik**
Saarbrücken 66123, Germany
`{ribragim,mmalek}@mpi-inf.mpg.de`

2 **Universität des Saarlandes**
Campus E 1.4, Saarbrücken 66123, Germany
`jguo@mmci.uni-saarland.de`

3 **Computational Biology group, University of Southern Denmark**
Campusvej 5, 5230 Odense M, Denmark
`jan.baumbach@imada.sdu.dk`

Abstract

Introduction: With the so-called OMICS technology the scientific community has generated huge amounts of data that allow us to reconstruct the interplay of all kinds of biological entities. The emerging interaction networks are usually modeled as graphs with thousands of nodes and tens of thousands of edges between them. In addition to sequence alignment, the comparison of biological networks has proven great potential to infer the biological function of proteins and genes. However, the corresponding network alignment problem is computationally hard and theoretically intractable for real world instances.

Results: We therefore developed GEDEVO, a novel tool for efficient graph comparison dedicated to real-world size biological networks. Underlying our approach is the so-called Graph Edit Distance (GED) model, where one graph is to be transferred into another one, with a minimal number of (or more general: minimal costs for) edge insertions and deletions. We present a novel evolutionary algorithm aiming to minimize the GED, and we compare our implementation against state of the art tools: SPINAL, GHOST, C-GRAAL, and MI-GRAAL. On a set of protein-protein interaction networks from different organisms we demonstrate that GEDEVO outperforms the current methods. It thus refines the previously suggested alignments based on topological information only.

Conclusion: With GEDEVO, we account for the constantly exploding number and size of available biological networks. The software as well as all used data sets are publicly available at `http://gedevo.mpi-inf.mpg.de`.

1998 ACM Subject Classification F.2.2 Nonnumerical Algorithms and Problems

Keywords and phrases Network Alignment, Graph Edit Distance, Evolutionary Algorithm, Protein-Protein Interactions

Digital Object Identifier 10.4230/OASIcs.GCB.2013.68

1 Introduction

We have finally arrived in the post-genome era. At the web site of the National Center for Biotechnology Information (NCBI) we find registered sequencing projects for >1,500 eukaryotes, >8,500 prokaryotes and >3,000 viruses with >8,000,000 gene sequences in total [26]. However, the genes' function is often unclear and most-widely deduced from similarities to the

German Conference on Bioinformatics 2013 (GCB'13).
Editors: T. Beißbarth, M. Kollmar, A. Leha, B. Morgenstern, A.-K. Schultz, S. Waack, E. Wingender; pp. 68–79
OpenAccess Series in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Table 1 The highest edge correctnesses (EC) achieved by different tools for aligning two pairs of networks, adopted from [14] and extended by the results of SPINAL and GHOST. Note that GHOST did not terminate for *yeast2* vs. *human1*. Note that GEDEVO obtained better results (refer to Table 3). Table 2 summarizes all data sets.

	IsoRank [29]	GRAAL [13]	H-GRAAL [17]	MI-GRAAL [14]	C-GRAAL [16]	SPINAL [1]	GHOST [20]
yeast2 vs. *human1*	3.89	11.72	10.92	23.26	22.55	19.33	-
Meso vs. *Syne*	5.33	11.25	4.59	41.79	26.02	25.86	41.98

sequences of genes with known functions. Consequently, we still lack fundamental knowledge about crucial genetic programs, the interplay of genes and their products (the proteins), their biochemical regulations and their evolutionary appearance. We know very little about how cells, organs and tissues regulate survival, reproduction, differentiation or movement in response to changing internal and external conditions [24]. Many problems in understanding these issues are concerned with biological networks that model the interplay of all kinds of biological entities [4]. Most widely known are transcriptional gene regulatory networks and protein-protein interaction (PPI) networks. More than 16 million protein-protein interactions available through PSICQUIC [3] may serve as an example for the ongoing “data explosion”.

One of the major computational challenges in systems biology is biological Network Alignment [11], which aims to find a node-to-node mapping between two or more biological networks, optimizing a certain quality criterion. A quality criterion of a mapping usually reflects topological aspects and biological aspects, such as the number of shared interactions induced by a mapping of the nodes from two networks or a similarity of the biological sequences underlying the nodes. Comparing biological networks, particularly protein-protein interaction (PPI) networks, from different organisms has proven very useful for inferring biological function, besides relying on DNA sequence similarity alone [27, 16].

Biological Network Alignment was recently addressed by several tools. IsoRank [29] integrates the nodes’ neighborhoods with sequence information and models the alignment as an eigenvalue problem. C-GRAAL [16], SPINAL [1], GHOST [20], and MI-GRAAL [14] use a similar seed-and-extend approach. While C-GRAAL greedily builds a neighborhood-dependent mapping, SPINAL and MI-GRAAL model (and solve) a weighted bipartite graph problem. IsoRank was shown to be be outperformed by MI-GRAAL [14]. On real PPI networks, SPINAL as well as the GRAAL collection, proved to perform best and to offer biologically meaningful alignments. A brief comparison previously used in [14] is given in Table 1. Instead of replicating the conclusion from the above cited papers that Network Alignment offers biological insights, we concentrate on the methodological problem that the existing tools possess.

All approaches struggle to provide high-quality results on the huge, yet constantly increasing, biological networks that we are confronted with nowadays (Table 2). As we will demonstrate, the existing software cannot cope well with such big networks. This becomes most evident when we see them fail on aligning a network to itself, which should result in 100% node mapping accuracy.

In this paper we present GEDEVO, a novel method for PPI Network Alignment. GEDEVO is an evolutionary algorithm that uses the Graph Edit Distance as optimization model for finding the best alignments. For evaluation, we use a set of high quality PPI networks, including the same networks previously used for C-GRAAL [16] and MI-GRAAL [14] for comparison with existing tools (see Table 1). We will demonstrate that GEDEVO performs comparable or better than recent tools, being at the same time fast and flexible. An implementation of GEDEVO as well as all used data sets are publicly available under `http://gedevo.mpi-inf.mpg.de`.

2 Methods

2.1 Problem definition

Consider a pair of PPI networks modeled as two unlabeled unweighted graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ and a one-to-one mapping f between nodes V_1 and V_2. We define the Graph Edit Distance (GED) between G_1 and G_2 induced by mapping f as follows: $\mathrm{GED}_f(G_1, G_2) = |\{(u,v) \in E_1 : (f(u), f(v)) \notin E_2\} \cup \{(u', v') \in E_2 : (f^{-1}(u'), f^{-1}(v')) \notin E_1\}|$. By definition, $\mathrm{GED}_f(G_1, G_2)$ counts inserted or deleted edges induced by the mapping f, and it can easily be extended to reflect node/edge dissimilarities or any other related information (e.g. protein sequence similarity). Here, for Network Alignment, we aim to find a mapping f that minimizes the $\mathrm{GED}_f(G_1, G_2)$. Graph Edit Distance is a general model for the Graph Matching problem and defined as the minimal amount of modifications required in graph G_1 to make it isomorphic to graph G_2 (see for example [6] for more details).

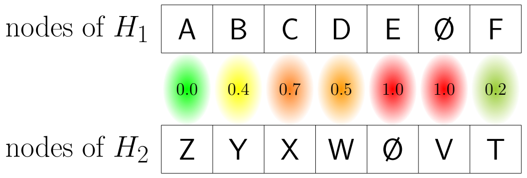
In previous work, the quality of the mapping f of most biological network aligners is assessed by using the number of shared interactions, defined as $|\{(u,v) \in E_1 : (f(u), f(v)) \in E_2\}|$ or the number of conserved interactions, defined as $|\{(u,v) \in E_1 : \mathrm{dist}(v, f(v)) < \Delta, \mathrm{dist}(u, f(u)) < \Delta, (f(u), f(v)) \in E_2\}|$, where $\mathrm{dist}(x, y)$ is a dissimilarity between $x \in V_1$ and $y \in V_2$ (such as BLAST E-value), and Δ is the node dissimilarity threshold. This corresponds to the intuition that the closer two species in the evolutionary tree are, the higher the number of conserved interaction partners they share. Incorporating external biological information, such as sequence similarity, can relax the Network Alignment problem to some extent by significantly reducing the search space by pre-defining preferable sets of nodes to be mapped. Although GEDEVO can include such external information, a "good" method should be able to determine an optimal mapping, by maximizing the number of shared interactions and thus utilizing the graph structure alone. For this reason and to assure comparability between the existing biological Network Alignment tools we focus on topological criteria only in this paper.

2.2 Evolutionary algorithm for Graph Edit Distance

Evolutionary Algorithms (EAs) are nature inspired heuristics, which are widely used to tackle many NP-hard problems (see for example [8]). The key idea behind EAs is mimicking the rule "survival-of-the-fittest" on a population of different individuals. Note that in [15] an EA was suggested for the Graph Matching problem. However, the major hindrance for efficient EA-based combinatorial optimization remained unsolved: the generation of new individuals. In the following we briefly introduce a general scheme of an EA and describe how we modified it with partial adoptions from [15] to Network Alignment.

In an EA an individual represents a solution for the problem, i.e. a mapping of nodes between two networks (see Figure 1). The state of an individual determines how well the individual fits to the requirements of the environment it populates. New individuals result from inheriting parts of solutions from its parents; the better an individual fits the requirements, the higher are its chances to survive and to pass its solution to future generations. Mutations of the solutions exposed to a new individual are another way of mimicking nature in EAs. The requirements of the environment are related to the fitness function, for which we utilize the above defined GED. Starting with generating a (quasi)

Figure 1 A mapping between networks H_1 (nodes A, B, C, D, E, and F) and H_2 (nodes T, V, W, X, Y, and Z) with arbitrary pair scores (for illustration only). In this mapping, node A fits perfectly to node Z, node C corresponds quite poorly to node X, node E is deleted and node V is inserted and both have worst pair scores. One major principle behind GEDOVO's generation of new individuals is to swap pairs of nodes with bad pair scores.



random initial population, an EA repeats the following three steps (individual evaluation, offspring generation, survival function application) until a termination criterion is met.

2.2.1 Initial Population Generation and Evaluation of an Individual

An individual represents a mapping f. Individuals in the initial population are created with random permutations. However, initialization in a more sophisticated manner, which, as a consequence, will require more time, may reduce the convergence time of the algorithm. Here, we may use protein sequence similarities, acquired by BLAST [2], for instance.

Evaluating individuals, for every pair $u \in V_1$ and $v \in V_2$ with $v = f(u)$, we define a *pair score* that reflects how well node u corresponds to node v given a mapping f:

$$\text{pairScore}_f(u, v) = (\text{pairGED}_f(u, v) + \text{grlets}(u, v))/2$$

where $\text{pairGED}_f(u, v)$ is the relative number of deleted and inserted edges induced by mapping node u to node v given mapping f, and $\text{grlets}(u, v)$ is the graphlet degree signature distance introduced in [18]. The graphlet signature distance (GSD) can be interpreted as the difference in neighboring topologies (within distance 4) of two nodes. GSD is computed from two graphlet degree vectors (GDVs); a GDV of a node counts graphlet orbits, which are topologically distinct induced subgraphs (with up to five nodes) the node touches. Note that although computing GDVs for a network with $|V|$ nodes requires $O(|V|^5)$ time it is still practically feasible for graphs as sparse as PPI networks. In addition, GDVs can be precomputed for each network and stored with the graph itself on the hard disk.

The pairScore is mainly used as local optimization guideline. The Graph Edit Distance (GED) is the final, global fitness score of an individual that is to be optimized. GEDEVO exploits the graphlet degree signature distance (GSD) only to accelerate convergence of the algorithm. It is not bound to it; and other external data, such as sequence similarities, may be introduced as additional (weighted or unweighted) terms to the pairScore formula. Computing $\text{pairGED}_f(u, v)$ requires not more than $O(d_1 + d_2) = O(d)$ time, where d_1, d_2 are the maximal node degrees in G_1 and G_2 and $d := \max(d_1, d_2)$. Given precomputed GSDs, a look-up for $\text{grlets}(u, v)$ needs constant time. Thus, computing pair scores for a mapping takes $O(n \cdot (d + s)) = O(n \cdot d)$, where $n = |V_1| + |V_2|$, and s is the number of precomputed terms on the right side of the $\text{pairScore}_f(u, v)$ formula.

The score of an individual together with its *health*, a non-increasing function of the number of iterations and GED of the individual, defines its fitness. The introduction of health allows keeping individuals with a "bad" GED for a number of iterations instead of simply discarding them immediately. This introduces some divergence and contributes to avoiding local optima.

2.2.2 Offspring generation

To generate new individuals we combine a set of different operations to balance between a reasonably high population diversity to avoid local optima and a high and fast convergence towards optimal solutions. The operations are as follows:

- *Random generation* creates an individual by relating it to a mapping based on a random permutation; it requires $O(n)$ time.
- In *PMX-like mutation* we adopt the idea of partially-mapped crossover (PMX), initially introduced in [10]. We partition a mapping into two sets of pairs, low scores and high scores, by using the average over all pair scores in the mapping as a threshold. Afterwards, the high scoring pairs are swapped randomly. To avoid local minima, however, we also swap low scoring pairs with a low probability (of 1% for GEDEVO and PPI networks). PMX-like mutation evaluates each pair by using the pairScore, which requires at most $O(n \cdot d)$ time.
- A so-called *crossover* results in an individual that in the first place preserves pairs with low pair scores from two or more parents. Ties are resolved randomly. Crossover is similar to the previous operation with a term responsible for sorting n pairs from a constant number of parents $p \leq 8$, which results in $O(p \cdot (n \cdot d) + p \cdot n \cdot \log(p \cdot n)) = O(n \cdot (\log n + d))$ time.
- With *directed mutations* we swap of a number $r \leq 20$ randomly chosen "bad" pairs in the mapping of an individual. At the end, the one swap that induces the best score is kept. One swap requires recomputing two pair scores. Thus, the running time of the operation is bound by $O(r \cdot 2 \cdot n \cdot (d + s)) = O(n \cdot d)$.

These operations are GEDEVO's strategies to find and keep "good" pairs while a "bad" pair is swapped more often with another "bad" pair, in this way improving the final score of the mapping. Over a number of iterations, many individuals are exposed to these operations by GEDEVO to traverses the search space and optimizes the final score.

2.2.3 Termination

No practical exact algorithm for the Graph Edit Distance computation on large graphs exists. Consequently, it is hard to theoretically estimate the number of necessary iterations until a "good" solution can be achieved. Convergence time mainly depends on the population size as well as on the input graphs' topological properties. Our implementation of GEDEVO can be set to execute (1) a specified number of iterations, (2) a pre-specified running time, or (3) a fixed number of iterations of no significant changes in the mapping scores of the best individuals (such that convergence was probably reached).

The total theoretical running time of GEDEVO is based on the run times of the individual steps. The evaluation step is performed in $O(N \cdot n \cdot d)$, where N is the population size. The offspring generation step requires $O(N \cdot (n + n \cdot d + n \cdot (\log n + d) + n \cdot d)) = O(N \cdot n \cdot (\log n + d))$ time. The selection step sorts the individuals from the older and new generations in $O(2 \cdot N \cdot \log(2 \cdot N))$ time. Given that GEDEVO runs I iterations, its total running time sums up to $O(I \cdot N \cdot n \cdot (\log n + d))$.

3 Data

For the evaluation of GEDEVO with existing tools we used several PPI networks (see Table 2). The following six networks were previously used for evaluating C-GRAAL and

Table 2 Summary of PPI networks used for evaluations.

Short name	Species	Source	Proteins	Interactions
cjejuni	*Campylobacter jejuni*	[19]	1,095	2,988
Meso	*Mesorhizobium loti*	[28]	1,803	3,094
Syne	*Synechocystis sp.(PCC6803)*	[25]	1,908	3,102
ecoli_fi	*Escherichia coli*	[21]	1,941	3,989
yeast2	*Saccharomyces cerevisiae*	[7]	2,390	16,127
SC	*Saccharomyces cerevisiae*	[31]	5,152	24,847
HS	*Homo Sapiens*	[31]	5,878	14,015
DM	*Drosophila Melanogaster*	[31]	7,533	22,477
ulitsky	*Homo Sapiens*	[30]	7,384	23,462
human1	*Homo Sapiens*	[23]	9,141	41,456
hprd	*Homo Sapiens*	[22]	9,672	3,7047

MI-GRAAL. The two bacterial networks *cjejuni* and *ecoli_fi* are well-studied high-confidence networks: The first network resulted from high-throughput yeast two-hybrid screens; the second network was constructed using experimental and computational data (see [21]). The *Syne* network was obtained through a modified high-throughput yeast two-hybrid assay and covers around half (52%) of the total protein coding genes; similarly for network *Meso* that involves 24% of the protein coding genes. The high-confidence network *human1* was created by combining data from multiple sources including HPRD [22]. The network from [7] is based on (post-processed) data from high throughput experiments.

In addition, we obtained the networks *DM*, *SC*, and *HS* from the DIP database, which contains experimentally determined and manually curated protein interactions. The *hprd* network is a PPI network obtained from the Human Protein Reference Database (HPRD), which is a repository storing high-quality manually curated human interaction data. The human interactome network *ulitsky* is a compilation of protein-protein interactions, based mostly on small-scale experiments, from several interaction databases, including the HPRD database. Refer to Table 2 for a summary and citations.

4 Result and Discussions

Here, we evaluate GEDEVO against the four tools GHOST, SPINAL, C-GRAAL and MI-GRAAL, which form the current state of the art and have been shown to outperform other existing tools [1, 14, 16].

All tools were executed on a 64 bit Linux 2.6.32 kernel, running on an Intel Xeon CPU W3550 @ 3.07GHz and 12 GB RAM. SPINAL is deterministic and was thus executed only one time for each pair of the input networks, while MI-GRAAL, GHOST, C-GRAAL and GEDEVO, as randomized algorithms, we executed 10 times for each pair. The execution of all tools was interrupted after 24 hours of runtime without termination. MI-GRAAL and C-GRAAL, similarly to GHOST, require graphlet degree signatures as preliminary node similarity measures, which were precomputed and used as input (precomputation time not taken into account for evaluation). The termination criterion for GEDEVO was set to stop after 3,000 iterations of no significant improvement of the GED score amongst the best solutions (individuals).

The performance of all methods, similarly to [14] and [16], is assessed with the so-called

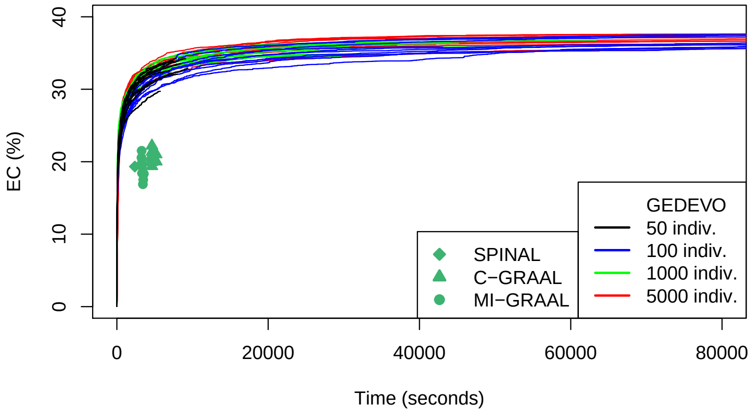


Figure 2 The influence of the population size to the performance of GEDEVO aligning *yeast2* vs. *human1* in comparison to SPINAL, C-GRAAL and MI-GRAAL. Each line/symbol represents one run.

Edge Correctness, which is particularly useful when comparing many networks with different numbers of nodes and edges. Defined as EC $= \frac{\#\text{sharedInteractions}}{\min(|E_1|,|E_2|)} \times 100$ [%], its highest value is 100% and occurs if one input network is isomorphic (or sub-isomorphic) to the other.

Note that GEDEVO internally utilizes the Graph Edit Distance for optimization, not the EC. This makes GEDEVO more applicable to general graph comparison problems outside computational biology. However, if we set the costs for node deletions/insertions/substitutions and edge substitution to zero but only the cost for edge deletions/insertions to one, the EC will be related to GED as EC $= (|E_1| + |E_2| - \text{GED})/(2 \cdot \min(|E_1|, |E_2|)) \times 100$ [%]. This allows us to compare GEDEVO to existing approaches on protein-protein interaction Network Alignment based on the EC criterion, as in previous work [14, 16], which is particularly useful for networks where differences between $|V_1|$ and $|V_2|$ are common (as in PPI networks from different organisms).

In Figure 2 we depict the influence of the population size to the progression of the EC (convergence). Runs with 50 individuals (black line) converged earlier to the final solution, still providing quite high values of EC. With larger population sizes the runs obtained slightly better alignments with higher EC and reached them slightly faster. This indicates that GEDEVO is quite robust to different population sizes, given that they are reasonably large. In the remaining (below described) evaluations we used 500 individuals per run.

We executed GEDEVO and the four competing tools on multiple pairs of networks from Table 2. The resulting edge correctnesses and the according run times for all tools are depicted in Figure 3. Since SPINAL, C-GRAAL and MI-GRAAL do not provide intermediate results, the final values are shown point-wise (diamonds, triangles, and circles); for GEDEVO the progression of EC is depicted with lines. The plot illustrates that GEDEVO can provide a "good" solution comparably fast. A summary of the maximal EC values from the plot is given in Table 3. Unsuccessful runs (no termination after 24 hours) of SPINAL and MI-GRAAL are marked with an "x"). Note: Since GHOST only terminated for the alignment of the two small networks *Meso* and *Syne* (with best EC: 41.98, runtime: 140 sec) we did not add it to Table 2 and Figure 3.

The networks *human1*, *hprd* and *ulitsky* are all human PPI networks, therefore the EC scores for GEDEVO are comparably high. The results for aligning *ulitsky* with *human1* or *hprd* are rather "poor" since *ulitsky* is a compilation of data from different databases. The known overlap of *ulitsky* with *hprd* is only 649 nodes and 15, 305 interactions, from which GEDEVO aligned 11, 800.

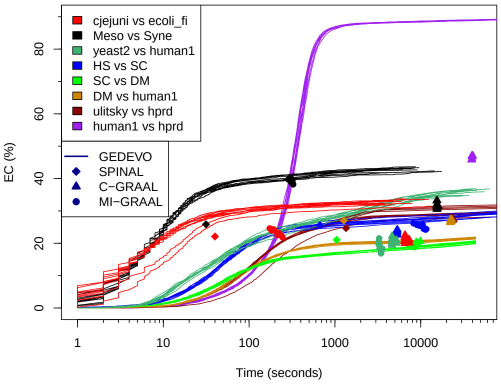


Figure 3 Convergence and Edge Correctness vs. run time for aligning different PPI networks, 10 runs for each tool. Each line/symbol represents one run.

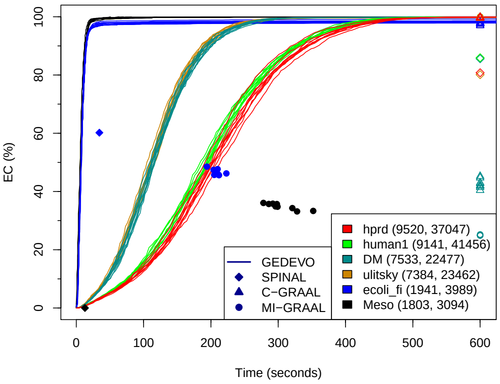


Figure 4 Aligning a network against itself should result in an Edge Correctness of 100%, which is achieved with GEDEVO and C-GRAAL in most cases (see text). Each line/symbol represents one run. Unfilled symbols (at the right side of the plot) mean that it took more than 10 minutes to achieve the corresponding EC values.

To further investigate the robustness of the four methods on networks where we definitely know the correct solution, we aligned some PPI networks against themselves. Naturally, this should result in an EC of 100%. In Figure 4, we plot the EC vs. run time for the following data sets: *Meso*, *ecoli_fi*, *ulitsky*, *DM*, and *human1*. Note that GHOST only terminated for the self-alignment of the two smallest networks *Meso* (with best EC: 100%, runtime: 197 sec) and *ecoli_fi* (with best EC: 100%, runtime: 173 sec). We also downloaded and tested Natalie 2.0 [9] on our servers. It terminated with memory faults for all network pairs but the two smallest ones: For *cjejuni* vs. *ecoli_fi* (runtime: 7 hours) and self-alignment of *Meso* (runtime: 11 hours) the tool resulted with edge correctnesses of 97.64% and 20.38% respectively. Hence, we did not include GHOST and Natalie 2.0 with Figure 4. Further note that a set of methods exists that restrict alignment candidates to a set of pre-mapped nodes (limited search space), see for example [12] and [5]. GEDEVO can be restricted to such pre-mappings (e.g. with BLAST as preprocessing) but it does not rely on it.

In conclusion, GEDEVO, in contrast to the other approaches, was able to achieve the expected 100% EC in all cases, often even faster than the existing tools. C-GRAAL reached around 97-98% of EC in most cases, but required up to 11 hours for the biggest networks (*hprd*, *human*), for which GEDEVO needed only approx. 10 minutes.

To sum up, in almost all cases, GEDEVO outperformed SPINAL, GHOST, C-GRAAL and MI-GRAAL in terms of quality and run time. Moreover, GEDEVO in contrast to the other methods was able to recognize the high similarity (*human1* vs. *hprd*) and composition (*ulitsky* vs. *hprd*) between the human PPI networks using topological information only. In addition, we wish to emphasize that GEDEVO provides intermediate results that allow for a manual termination of the software at earlier iterations when a high EC score (or a corresponding low Graph Edit Distance solution) has been found and convergence seems to be reached.

Table 3 The highest achieved Edge Correctness (EC) quality scores for alignments of different PPI networks from Figure 3.

Network 1	Network 2	EC (%)			
		GEDEVO	MI-GRAAL	C-GRAAL	SPINAL
cjejuni	*ecoli_fi*	**33.70**	*24.60*	22.56	22.09
Meso	*Syne*	**43.60**	*39.88*	33.19	25.86
yeast2	*human1*	**38.14**	21.38	*22.20*	19.33
HS	*SC*	**30.40**	*26.15*	24.15	25.59
SC	*DM*	*20.79*	17.73	20.59	**21.07**
DM	*human1*	21.88	x	**27.36**	*27.04*
ulitsky	*hprd*	**32.00**	x	*27.56*	24.68
human1	*hprd*	**89.37**	x	*47.07*	x

5 Conclusion

We presented GEDEVO, a novel Network Alignment algorithm, and evaluated it on protein-protein interaction networks. GEDEVO uses an evolutionary algorithm to heuristically approximate the Graph Edit Distance optimization problem. On a wide range of real PPI networks our approach outperforms state-of-the-art methods in terms of speed and quality, and provides intermediate alignment results on the fly. GEDEVO is robust and not limited to PPI networks (unlabeled, undirected, and unweighted graphs), but flexible enough to be applicable to other types of networks as well, biological and non-biological.

In the future we will speed-up the convergence of our algorithm by improving the population initialization by complementing the random permutations in this step with an assignment function that depends, for instance, on the degree differences between candidate node pairs (the less the difference, the higher the chance to contribute to low GED and high EC in the final solution). While in this paper, the quality measures were derived from purely topological alignments, in the future we will experiment with integrating additional external node-to-node scoring functions, such as BLAST.

GEDEVO as well as all used data sets are publicly available at `http://gedevo.mpi-inf.mpg.de`.

Acknowledgements We wish to thank Nataša Pržulj from Imperial College London for interesting discussions and her help with the data sets used in her C-GRAAL and MI-GRAAL papers. RI is grateful for financial support from the International Max Planck Research School (IMPRS). All authors with to thank the DFG Cluster of Excellence for Multimodal Computing and Interaction (MMCI) for support. JB is also grateful for funding from the Villum Foundation.

References

1 Ahmet E. Aladag and Cesim Erten. SPINAL: scalable protein interaction network alignment. *Bioinformatics*, 29(7):917–924, 2013.

2 Stephen F. Altschul, Warren Gish, Webb Miller, Eugene W. Myers, and David J. Lipman. Basic local alignment search tool. *Journal of molecular biology*, 215(3):403–410, October 1990.

3 Bruno Aranda, Hagen Blankenburg, Samuel Kerrien, Fiona S L Brinkman, Arnaud Ceol, Emilie Chautard, Jose M Dana, Javier De Las Rivas, Marine Dumousseau, Eugenia Ga-

leota, Anna Gaulton, Johannes Goll, Robert E W Hancock, Ruth Isserlin, Rafael C Jimenez, Jules Kerssemakers, Jyoti Khadake, David J Lynn, Magali Michaut, Gavin O'Kelly, Keiichiro Ono, Sandra Orchard, Carlos Prieto, Sabry Razick, Olga Rigina, Lukasz Salwinski, Milan Simonovic, Sameer Velankar, Andrew Winter, Guanming Wu, Gary D Bader, Gianni Cesareni, Ian M Donaldson, David Eisenberg, Gerard J Kleywegt, John Overington, Sylvie Ricard-Blum, Mike Tyers, Mario Albrecht, and Henning Hermjakob. PSICQUIC and PSISCORE: accessing and scoring molecular interactions. *Nature Methods*, 8(7):528–9, Jul 2011.

4 Jan Baumbach. On the power and limits of evolutionary conservation–unraveling bacterial gene regulatory networks. *Nucleic Acids Res*, 38(22):7877–84, Dec 2010.

5 Mohsen Bayati, David F. Gleich, Amin Saberi, and Ying Wang. Message-passing algorithms for sparse network alignment. *ACM Trans. Knowl. Discov. Data*, 7(1):3:1–3:31, March 2013.

6 Horst Bunke and Kaspar Riesen. *Graph Edit Distance – Optimal and Suboptimal Algorithms with Applications*, pages 113–143. Wiley-VCH Verlag GmbH & Co. KGaA, 2009.

7 Sean R. Collins, Patrick Kemmeren, Xue-Chu Zhao, Jack F. Greenblatt, Forrest Spencer, Frank C. P. Holstege, Jonathan S. Weissman, and Nevan J. Krogan. Toward a comprehensive atlas of the physical interactome of saccharomyces cerevisiae. *Molecular and Cellular Proteomics*, 6(3):439–450, 2007.

8 Agoston E. Eiben and J. E. Smith. *Introduction to evolutionary computing.* Natural Computing Series. Springer, December 2010.

9 Mohammed El-Kebir, Jaap Heringa, and GunnarW. Klau. Lagrangian relaxation applied to sparse global network alignment. In Marco Loog, Lodewyk Wessels, MarcelJ.T. Reinders, and Dick Ridder, editors, *Pattern Recognition in Bioinformatics*, volume 7036 of *Lecture Notes in Computer Science*, pages 225–236. Springer Berlin Heidelberg, 2011.

10 David E. Goldberg and Robert Jr. Linge. Alleles, loci, and the traveling salesman problem. In John J. Grefenstette, editor, *Proceedings of the First International Conference on Genetic Algorithms and Their Applications.* Lawrence Erlbaum Associates, Publishers, 1985.

11 Allison P Heath and Lydia E Kavraki. Computational challenges in systems biology. *Computer Science Review*, 3(1):1–17, 2009.

12 Gunnar Klau. A new graph-based method for pairwise global network alignment. *BMC Bioinformatics*, 10(Suppl 1):S59, 2009.

13 O. Kuchaiev, T. Milenković, V. Memišević, W. Hayes, and N. Pržulj. Topological network alignment uncovers biological function and phylogeny. *Journal of the Royal Society Interface*, 7(50):1341–1354, 2010.

14 Oleksii Kuchaiev and Nataša Pržulj. Integrative Network Alignment Reveals Large Regions of Global Network Similarity in Yeast and Human. *Bioinformatics*, 27(10):1390–1396, March 2011.

15 Cheng-Wen Liu, Kuo-Chin Fan, Jorng-Tzong Horng, and Yuan-Kai Wang. Solving weighted graph matching problem by modified microgenetic algorithm. In *Systems, Man and Cybernetics, 1995. Intelligent Systems for the 21st Century., IEEE International Conference on*, volume 1, pages 638–643. IEEE, 1995.

16 Vesna Memišević and Nataša Pržulj. C-GRAAL: Common-neighbors-based global GRAph ALignment of biological networks. *Integr. Biol.*, 4:734–743, 2012.

17 T. Milenković, W.L. Ng, W. Hayes, and N. Pržulj. Optimal network alignment with graphlet degree vectors. *Cancer informatics*, 9:121, 2010.

18 Tijana Milenković and Nataša Pržulj. Uncovering biological network function via graphlet degree signatures. *Cancer Informatics*, 6:0–0, 04 2008.

19 Jodi Parrish, Jingkai Yu, Guozhen Liu, Julie Hines, Jason Chan, Bernie Mangiola, Huamei Zhang, Svetlana Pacifico, Farshad Fotouhi, Victor DiRita, Trey Ideker, Phillip Andrews,

and Russell Finley. A proteome-wide protein interaction map for campylobacter jejuni. *Genome Biology*, 8(7):R130, 2007.

20 Rob Patro and Carl Kingsford. Global network alignment using multiscale spectral signatures. *Bioinformatics*, 28(23):3105–3114, 2012.

21 Jose M. Peregrin-Alvarez, Xuejian Xiong, Chong Su, and John Parkinson. The modular organization of protein interactions in *Escherichia coli. PLoS Comput Biol*, 5(10):e1000523, 10 2009.

22 Keshava T.S. Prasad, Renu Goel, Kumaran Kandasamy, Shivakumar Keerthikumar, Sameer Kumar, Suresh Mathivanan, Deepthi Telikicherla, Rajesh Raju, Beema Shafreen, Abhilash Venugopal, Lavanya Balakrishnan, Arivusudar Marimuthu, Sutopa Banerjee, Devi S. Somanathan, Aimy Sebastian, Sandhya Rani, Somak Ray, Harrys C.J. Kishore, Sashi Kanth, Mukhtar Ahmed, Manoj K. Kashyap, Riaz Mohmood, Y.L. Ramachandra, V. Krishna, Abdul B. Rahiman, Sujatha Mohan, Prathibha Ranganathan, Subhashri Ramabadran, Raghothama Chaerkady, and Akhilesh Pandey. Human Protein Reference Database–2009 update. *Nucleic acids research*, 37(Database issue):D767–D772, January 2009.

23 Predrag Radivojac, Kang Peng, Wyatt T. Clark, Brandon J. Peters, Amrita Mohan, Sean M. Boyle, and Sean D. Mooney. An integrated approach to inferring gene-disease associations in humans. *Proteins: Structure, Function, and Bioinformatics*, 72(3):1030–1037, 2008.

24 Richard Röttger, Ulrich Rückert, Jan Taubert, and Jan Baumbach. How little do we actually know? On the size of gene regulatory networks. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 9:1293–1300, 2012.

25 Shusei Sato, Yoshikazu Shimoda, Akiko Muraki, Mitsuyo Kohara, Yasukazu Nakamura, and Satoshi Tabata. A large-scale protein-protein interaction analysis in synechocystis sp. pcc6803. *DNA Research*, 14(5):207–216, 2007.

26 Eric W Sayers, Tanya Barrett, Dennis A Benson, Evan Bolton, Stephen H Bryant, Kathi Canese, Vyacheslav Chetvernin, Deanna M Church, Michael DiCuccio, Scott Federhen, Michael Feolo, Ian M Fingerman, Lewis Y Geer, Wolfgang Helmberg, Yuri Kapustin, David Landsman, David J Lipman, Zhiyong Lu, Thomas L Madden, Tom Madej, Donna R Maglott, Aron Marchler-Bauer, Vadim Miller, Ilene Mizrachi, James Ostell, Anna Panchenko, Lon Phan, Kim D Pruitt, Gregory D Schuler, Edwin Sequeira, Stephen T Sherry, Martin Shumway, Karl Sirotkin, Douglas Slotta, Alexandre Souvorov, Grigory Starchenko, Tatiana A Tatusova, Lukas Wagner, Yanli Wang, W John Wilbur, Eugene Yaschenko, and Jian Ye. Database resources of the National Center for Biotechnology Information. *Nucleic Acids Res*, 39(Database issue):D38–51, Jan 2011.

27 Roded Sharan, Silpa Suthram, Ryan M Kelley, Tanja Kuhn, Scott McCuine, Peter Uetz, Taylor Sittler, Richard M Karp, and Trey Ideker. Conserved patterns of protein interaction in multiple species. *PNAS*, 102(6):1974–1979, Feb 2005.

28 Yoshikazu Shimoda, Sayaka Shinpo, Mitsuyo Kohara, Yasukazu Nakamura, Satoshi Tabata, and Shusei Sato. A large scale analysis of protein-protein interactions in the nitrogen-fixing bacterium mesorhizobium loti. *DNA Research*, 15(1):13–23, 2008.

29 Rohit Singh, Jinbo Xu, and Bonnie Berger. Pairwise global alignment of protein interaction networks by matching neighborhood topology. In *Proc. of the 11th annual international conference on Research in computational molecular biology*, RECOMB'07, pages 16–31, Berlin, Heidelberg, 2007. Springer-Verlag.

30 Igor Ulitsky, Richard M. Karp, and Ron Shamir. Detecting Disease-Specific Dysregulated Pathways Via Analysis of Clinical Expression Profiles Research in Computational Molecular Biology. In Martin Vingron and Limsoon Wong, editors, *Research in Computational Molecular Biology*, volume 4955 of *Lecture Notes in Computer Science*, chapter 30, pages 347–359. Springer Berlin / Heidelberg, Berlin, Heidelberg, 2008.

31 Ioannis Xenarios, Lukasz Salwínski, Xiaoqun Joyce, Patrick Higney, Sul-Min M. Kim, and David Eisenberg. DIP, the Database of Interacting Proteins: a research tool for studying cellular networks of protein interactions. *Nucleic acids research*, 30(1):303–305, January 2002.