Supplementary Material to

Evolutionary Construction of Multiple Graph Alignments for the Structural Analysis of Biomolecules

Thomas Fober, Marco Mernberger, Gerhard Klebe, Eyke Hüllermeier
University of Marburg, Germany

Contents

SOFTWARE: An implementation of the GAVEO algorithm, along with a user's guide, can be downloaded on the following website:

`www.uni-marburg.de/fb12/kebi/research/`

1 Related Work

The problem of identifying common structural similarities among structured data has simultaneously arisen from many fields of research, including pattern recognition, data mining, structural bio- and chemoinformatics, database systems and many more. In particular, the use of graphs as a modeling concept for structured data has been proposed by several authors. Since graphs are a rather general data structure, graph-based methods are very flexible and widely applicable. Especially in life sciences, the analysis of structured data has gained increased attention in recent years. In the field of bioinformatics, for example, graphs have not only been used for modeling molecular structures, but also for modeling biological networks, such as regulatory networks [16], interaction networks [5, 88], metabolic networks [36], or phylogenetic networks [31]. Moreover, graph-based models also play an important role beyond the bioinformatics domain. For example, graphs can be used to model other kinds of networks, such as social networks [87], HTML/XML documents [57], or the Internet itself [9]. Hence it is not surprising, that a plethora of methods exist to tackle this problem.

Roughly speaking, one can distinguish a couple of generic principles of graph similarity on which most of the existing approaches are based. A first concept that has been widely used in chemoinformatics, pattern matching and computer vision considers two graphs similar if they are isomorphic or share at least a common subgraph which leads to the (sub)graph isomorphism problem, and closely related to this, the concept of the maximum common subgraph. Instead of looking for a single, as large as possible compliance, one may also look for many small compliances. Thus, it might be more reasonable to look for a large number of smaller common substructures and define similarity accordingly which is the basic idea of frequent subgraph mining. A third principle is based on the generic concept of an *edit distance*. According to this principle, two graphs are similar if a few modifications (edit operations) are sufficient to make the first one isomorphic to the second one. As mentioned previously, our idea of *aligning* (multiple) graphs is actually closely related to this conception of similarity. Note that this approach naturally supports the idea of an *approximate* match between graphs and, thereby, of approximately conserved patters, which makes it especially interesting for our purpose. In contrast, the first two approaches focus primarily on exact matches between graphs. Although it is possible to extend these concepts to approximate similarity, such extensions are often difficult to realize algorithmically. Other approaches aim at representing graphs or structures by defining certain representative features and calculate similarity accordingly.

Subsequently, we give a brief review of general approaches to graph comparison (cf. Section 1.1) and then focus on those commonly used in bioinformatics that are most relevant for protein structure analysis (cf. Section 1.2). Finally we give a short overview on alternative non-graph-based methods in the field of protein comparison (cf. Section 1.3).

1.1 General approaches to graph matching and comparison

Graph isomorphism and subgraph isomorphism are standard concepts for determining the similarity of graphs in the field of pattern matching, for which standard algorithms have long been known [79, 63]. Closely related to these concepts is the principle of common subgraphs. In chemoinformatics, the concepts of maximum common subgraph (MCS) [14] and minimum common supergraph (mcs) [12] have been widely used for the comparison of chemical compounds [61]. Obviously, both measures are related and can be used as distance metrics on graphs, either separately, for example based on the MCS [13], or combined into a single distance measure [20]. A variety of algorithms have been proposed for the calculation of the MCS, some of them being exact algorithms using clique-detection [10, 59] and, to a lesser extend, also backtracking algorithms [49, 66]. Other approaches approximate the MCS, often based on combinatorial optimization techniques [83, 62] as the problem is provably NP-hard, which in fact is a major problem for all these methods. As a major disadvantage of isomorphism-based similarity is the requirement of exact node matches between graphs, some approaches exist that extend the concept to approximate graph matching techniques [15, 75, 89, 86]. However, these methods are likely to get stuck in local optima.

Furthermore, the methods mentioned so far are generally limited to simple pairwise comparisons and concentrate on one large common subgraph. As pointed out earlier, frequent subgraph mining aims at identifiying a large set of smaller common substructures instead of concentrating on single large subgraph to define similarity on graphs, while offering the opportunity to incorporate multiple graphs into the analysis. Early contributions in this area employed computationally expensive inductive logic programming (ILP) [17, 74]. As this is unfeasible for larger or a greater number of graphs approximate algorithms were also proposed, but these early approaches could not guarantee to find all common substructures [94, 27]. More advanced methods extend the well-known apriori algorithm [1] for mining frequent item sets to this problem [32, 44, 45].

Faster approaches have also been proposed. Borgelt and Berthold developed an algorithm that employs a depth-first tree search with structural pruning [7, 8]. ClosedGraph is an approach that constraints itself by looking for connected closed subgraphs [90] and FFSM utilizes efficient subgraph enumeration operations [30]. However, they all constrain the patterns they allow to connected subgraphs. Although these approaches were successfully employed in chemoinformatics, they are generally not applicable for larger graph structures that may arise when analyzing protein structure data due to their complexity.

The above mentioned methods are in most cases dependent on exact matches, although approximations have also been considered to a certain degree. However, as especially in life sciences one has to deal with inconsistencies and noisy data, more powerful approximate and error-tolerant graph matching techniques are required. As pointed out earlier, a powerful alternative to subgraph isomorphism is given by the concept of graph edit distances as a similarity measure between graphs, originally introduced by Sanfeliu et al. [65]. The distance or similarity between two graphs is given by the minimal sequence of edit operations needed to transform one graph into the other. Edit operations are typically insertions, deletions or label/weight changes of nodes, respectively edges. This is a more general approach to graph matching than the subgraph methods mentioned above, in fact it could be shown that graph and subgraph isomorphism as well as the MCS problem are special instances of graph edit computations [11]. Our proposed method is also based on this concept.

Most algorithms that utilize graph edit distances stem from the field of computer vision. Here graph edit distances were used in combination with enumeration techniques and indexing methods [51, 50], probabilistic edit models [54, 64, 6] or hill climbing heuristics [84]. A more recent approach uses binary linear programming to calculate graph matchings based on graph edit distances [34]. A prominent approach that has been motivated by the task of parsing three-dimensional structure databases of chemical compounds employs a geometric hashing method introduced by [46] to find approximate matching counterparts to a query molecule as indicated by the graph edit distance [85]. However, note that the usefulness of graph edit distances strongly depend on the underlying cost function, more specifically on the costs assigned to a certain edit operation [11].

Another approach builds upon graph edit distances as well but proposes the utilization of a vector representation of graphs [58]. This is equivalent to the fourth principle mentioned above, the feature based representation of structured objects. Of course many possibilities to define features on graphs exist. The main challenge is here of course to cover all aspects that might be relevant to the underlying problem. One approach to feature definition is to look at local features in a graph, which leads to similarity measures based on local rather than global similarity. Local approaches to graph comparison generally look for the compliance of properties that refer to substructures or local components of a graph, such as subgraphs, paths or walks. In contrast to subgraph isomorphism approaches, local methods typically aim at the identification of a set of characteristic substructures for a given group of graphs rather than the calculation of a single maximum common subgraph.

Main contributions to local similarity measures have recently been made in the field of kernel-based machine learning [70]. A kernel function defined on a set X is an $X \times X \rightarrow \mathbb{R}$ mapping satisfying certain formal properties (including symmetry and positive semi-definiteness), which makes them appealing both from a mathematical and algorithmic point of view. Generally, a kernel function can also be viewed as a similarity function. Several kernel functions on graphs have already been proposed, some of which are based on walks or, more precisely, random walks [22]. Here, walks are generated in one graph at random and then searched in the second graph. The number of random walks present in both graphs can be used to define a similarity metric. Other kernels build upon random walks [37] or are closely related to the concept, for example diffusion kernels [42]. Since the number of possible random walks can become extraordinary large, the use of shortest paths has been proposed as an alternative [9]. Other graph kernels are based on graph edit distances [55]. A number of kernels exist that are deliberately tailored towards chemoinformatics, namely the Tanimoto kernel, the min-max kernel and the hybrid kernel[60].

An interesting approach was suggested by Neuhaus and Bunke. They stated that graph kernels and edit distance based matching algorithms tackle the problem of graph similarity in a complementary way, and for given applications, either the first or the second approach is superior to the other. They combined both priciples by enhancing a random walk kernel by adding information based on graph edit distances [55]. Another kernel is the graphlet kernel [9] that also makes use of substructures (a graphlet is a subgraph consisting of four nodes) to calculate the similarity between two graphs. The related concept of an optimal assignment kernel has recently been introduced in [21]. Here, the idea is to search for an assignment of subcomponents of the graphs so that, for a given kernel function on the subcomponents, the sum over all mutually assigned pairs becomes maximal. Strictly speaking, the term ‘kernel’ is misleading here, since this measure does actually not fulfill the kernel properties [81].

Aside from kernel functions there exist alternative approaches that build upon different feature representations of graphs. One line of works is focused on graph decomposition methods. As every graph can be represented by its adjacency matrix, a number of decomposition methods have been employed to solve the mapping problem of graphs. Several approaches utilize an eigenvalue decomposition of the adjacency matrix [80]. Recently, Kondor and Borgwardt introduced a set of invariant matrices derived from graphs by Fourier transformation called the skew spectrum [43]. They showed that it could compete with state-of-the-art graph kernels.

1.2 Graph methods in protein structure comparisons

The comparative analysis of protein structures is more challenging, since these structures can have a considerably larger size which renders many of the above mentioned approaches useless for this task. Thus, a variety of specialized approaches exist that aim at the identification of common three-dimensional patterns and substructures in proteins, corresponding to relevant sites for the protein function, such as catalytic triads or protein binding sites. Some of those approaches resort to principles and methods from graph theory. Among these are the ASSAM algorithm by [4, 73], a method that exploits a protein surface database (ef-site) [39, 40], and the approach of [33], in which the amino acid structure of a protein is represented as a set of chemical groups. Those approaches mostly utilize clique detection algorithms to discover similar substructures in graphs.

A variety of algorithms exist that build upon a higher-level representation of proteins than atom coordinates. Artymiuk et al. first proposed the use of secondary structure elements, such as alpha helices and beta sheets in conjunction with additional information such as orientation of these elements, to tackle the problem of aligning similar protein structures [26, 52] by employing clique-detection algorithms. Based upon this idea, several other algorithms have evolved that employ this strategy [48, 3].

Other approaches, originating from the database field, aim at the exploration of (potentially very large) graph databases [67, 91, 95]. These methods usually employ exact matching techniques which is problematic in life sciences, where structured data is usually noisy and incomplete. Therefore, query algorithms for the approximate matching of graphs have been developed as well [92, 93]. Yet, these approaches still are not very flexible, as they do not allow insertions or deletions of nodes. SAGA is a more versatile approach that uses a flexible graph similarity model [77]. Although SAGA is very efficient on small graphs, it is computationally expensive for large graphs. The recently proposed TALE algorithm instead allows for the matching of even large graphs by using a novel sophisticated indexing method [78].

A fundamental limitation of the previous approaches is their restriction to pairwise comparisons, i.e., the comparison of two graphs. Even though pairwise comparisons are sufficient to define a similarity measure and, hence, to apply similarity- or distance-based data analysis tools such as cluster analysis, they are not fully adequate for certain biological applications. In the functional analysis of proteins, for example, it is important to find those features that are conserved across a whole family of related structures. This is why methods for multiple sequence alignment are of major importance in bioinformatics. Obviously, high scoring pairwise alignments do in general not correspond to high scoring alignments of multiple molecules [2].

Only a few approaches exist that are able to compare multiple protein structures or protein binding sites. These approaches employ heuristics to overcome the inherent computational complexity of the problem and make additional assumptions on the structure of the proteins to simplify the problem [18, 68, 47]. However, as proteins can share similar functions without common backbone folds, in conjunction with the fact that the heuristics can easily miss small common substructures, these methods cannot guarantee to retrieve all important features. Recently, Shatzky et al. proposed the MultiBind algorithm [69] that defines the multiple alignment of protein substructures such as protein binding sites to the problem of finding a multiple common point set of 3D points that does not rely on additional information.

We like to emphasize that, despite their superficial resemblance, most of the methods and algorithms for analyzing graphs are quite specialized and not universally applicable. In fact, given the existence of many types of graphs (directed vs. undirected, labeled vs. unlabeled, etc.), it is clear that a method suitable for one problem class might not be useful for (and perhaps not even applicable to) another one. And even within a single class of graphs, the suitability of an algorithm may strongly depend on concrete properties of the problems given as input. For example, an algorithm working effectively on graphs with a small number of different node labels may become ineffective if this number is too high. This observation is quite important against the background of biological applications, since these applications have special requirements. For example, our method of multiple graph alignment is tailored toward the comparative analysis of a special type of graph structure for modeling protein binding sites, and we are not aware of any other method equally useful for this purpose.

1.3 Protein structure and fold comparison

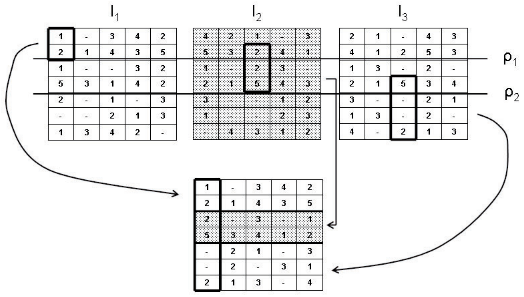
The structural comparison of proteins and protein substructures is not limited to graph-based approaches alone. We now give a short review of relevant approaches in this field of application beyond the scope of graph-based models. Several approaches in this field of research exist that are not based on graph representations of proteins (cf. Section 1.2) but focus on alternative protein representations. A general goal of these algorithms is often to derive an alignment that can be used to superimpose structures.

Some approaches focus on deriving suitable sequence alignments backed up by structure information, such as the Euclidian distance between Cα-Atoms. Such alignments can be used to generate a superposition of structures that can be evaluated by the root mean square deviation (RMSD), which is basically a measure for the structural overlap of two superimposed structures. It can be easily calculated by using the Kabsch algorithm [35] and is often used as a quality measure for structural superpositions. The DALI method for instance uses distance matrices of inter-residue distances based on the corresponding Cα-Atoms to represent proteins and calculates an alignmnet of structural equivalent residues by using a Monte Carlo approach [29, 28]. Other approaches (SSAP, MUTAL) compare inter-atomic distance vectors for a certain residue [56, 76] or focus on pairwise residue distances [23, 24]. MUTAL even allows for the comparison of multiple structures. Shindyalov et al. proposed an approach based on a combinatorial extension technique (CE) [71]. The main drawback of these methods is their relatively huge computational cost which renders them less suited for large-scale analyses.

Yet other approaches employ a higher level representation of protein structure and compare secondary structure elements (SSE) of proteins. Those methods employ dynamic programming, depth-first search, three-dimensional clustering or a Markov Transition Model to align similar SSEs [72, 25, 41, 53, 82, 38]. Since there are only small numbers of SSE present in a protein structure, algorithms that focus on this representation are usually faster than those building upon more detailed representation. The concept of SSEs gives also rise to a variety of graph based methods, as was mentioned above (cf. Section 1.2). However, these methods resemble a fold-based similarity, as SSE are rather coarse features that are not able to capture finer details. Other approaches aim to find the best superposition of two proteins by minimizing the surface between virtual protein backbones [19]. The metric of choice for this task is the RMSD-value.

2 Illustration of the Recombination Operator

The recombination operator is illustrated in Fig. 1 for the case $\rho = 3$. Three individuals I_1, I_2, and I_3 and two integers ρ_1 and ρ_2 in the range $\{1 \dots m\}$ are chosen at random. All individuals are split at the rows ρ_1 and ρ_2. The resulting blocks are merged into a new individual (offspring). To preserve the ordering, columns are rearranged according to the rows ρ_1 and ρ_2, respectively, whose indexes serve as pivot elements: For example, the first framed subcolumn in I_1 is copied to the offspring, and since the index in the pivot row ρ_1 is 2, we have to search for the same index in this row in I_2. This subcolumn (framed) is also copied into the offspring. This procedure is repeated for all individuals and columns.



1	-	3	4	2
2	1	4	3	5
1	-	-	3	2
5	3	1	4	2
2	-	1	-	3
-	-	2	1	3
1	3	4	2	-

4	2	1	-	3
5	3	2	4	1
1	-	2	3	-
2	1	5	4	3
3	-	-	1	2
1	-	-	2	3
-	4	3	1	2

2	1	-	4	3
4	1	2	5	3
1	3	-	2	-
2	1	5	3	4
3	-	-	2	1
1	3	-	2	-
4	-	2	1	3

1	-	3	4	2
2	1	4	3	5
2	-	3	-	1
5	3	4	1	2
-	2	1	-	3
-	2	-	3	1
2	1	3	-	4

Figure 1: Recombination of $\rho = 3$ individuals

3 Comparing GAVEO and Hill-Climbing

Average scores and standard deviations for GAVEO and a simple hill climber (HC) are shown in the following table:

# graphs	GAVEO	HC
2	-22.2 ± 74.7	-191.4 ± 47.6
4	-98.4 ± 144.6	-1152.4 ± 87.4
8	-539.3 ± 352.2	-6484.2 ± 1156.4
16	-5888.0 ± 1626.4	-33004.0 ± 3249.4

4 Runtimes of GAVEO, GAVEO*, and the greedy approach on the thermolysin data set

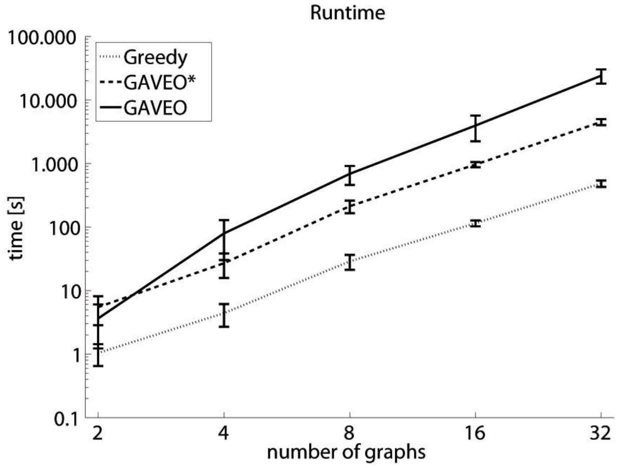


Figure 2: Runtimes in seconds (mean and standard deviation) of Greedy, GAVEO, and GAVEO*

5 Depiction of a pairwise Graph Alignment

Naturally, the depiction of a multiple graph alignment showing the real underlying structures, that were used in the experiments is difficult and prone to be overloaded. Hence we only show a subpart of a multiple alignment consisting of two cavities and the mapping that was calculated during the alignment process as an example in Fig. 3. The two structures resemble protein binding pockets of members of the thermolysin family, which we used during our experiments. Shown are the amino acids bordering the cavities, as well as the corresponding pseudocenters, which are depicted as spheres. Dotted lines indicate the mapping of the graph alignments. The color of the lines indicates a node match (green) or a node mismatch(red). Mappings of a node onto a dummy are omitted to improve the clarity of the visualization.

6 Visualization of Multiple Graph Alignments

In Fig. 4, each tuple of an alignment corresponds to a segment of a circle. The outer part of a segment provides information about the mutually assigned nodes: The nodes are

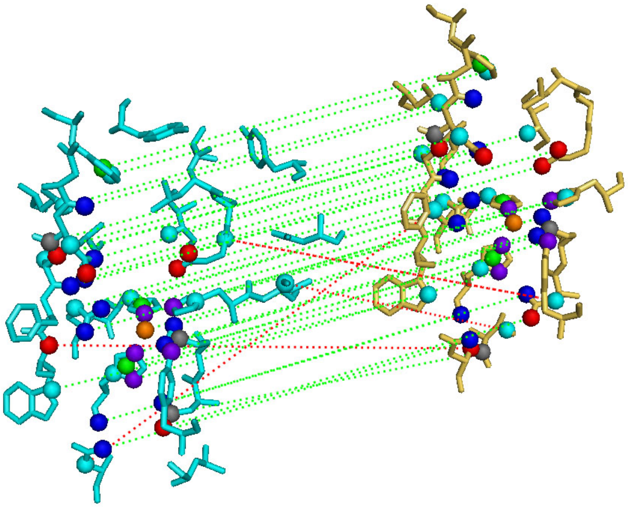


Figure 3: Depiction of a part of an alignment of eight thermolysin structures calculated by GAVEO. Only two of the eight cavities are shown. Spheres represent pseudocenters with colors indicating the type of the pseudocenter (donor(red), acceptor(blue), donor/acceptor(purple), aliphatic(cyan), aromatic(green), metal(orange), PI(grey)). Dotted lines indicate mappings of nodes onto each other as calculated by GAVEO. Red indicates a node mismatch, green indicates a node match.

sorted according to their frequency, and the list of frequencies thus obtained is shown as a kind of color histogram, using a fixed order of colors (the frequency of dummy nodes is shown in white). Thus, the more "pure" a segment is, i.e., the less colors it contains and the more dominant a single color is, the better is the alignment. The lines in the inner part of the circle provide information about the matches between edges. The length of a line is proportional to the average fraction of mismatches (defined by a threshold) between the edges that emerge from the nodes in this segment and their corresponding match partners. Thus, a good alignment is almost unicolored at the outside and mostly uncolored in the middle.

7 Illustration of Conserved Patterns

In the experiments on mining protein binding pockets, the GAVEO algorithm was applied to a data set consisting of 74 structures derived from the Cavbase database. Each structure represents a protein cavity belonging to the protein family of thermolysin, bacterial proteases frequently used in structural protein analysis and annotated with the E.C. number 3.4.24.27 in the ENZYME database.

Fig. 5 shows an example of a conserved pattern that was discovered by GAVEO (parameters $\alpha = 1$, $\beta = 0.9$, as explained in the paper). The pattern includes a metal pseudocenter surrounded by several acceptor and donor/acceptor centers. As thermolysin is a bacterial metalloprotease, it obviously captures the subpart of the cavity hosting the zinc ion of thermolysin. The surrounding acceptor pseudocenters probably correspond to residues

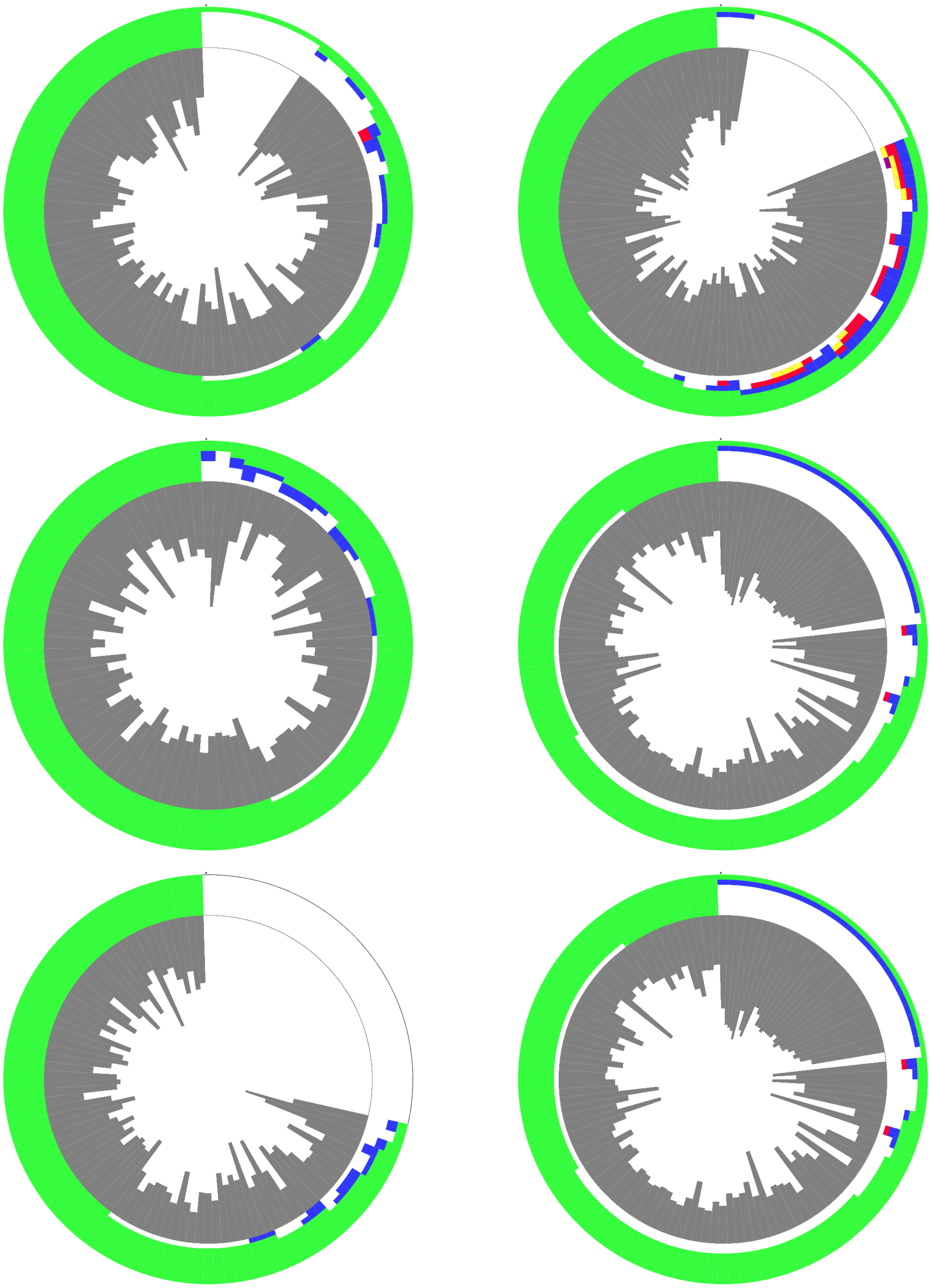


Figure 4: Visual representation of exemplary MGAs calculated by GAVEO (left) and Greedy (right) for benzamidine structures. The bottom pictures show the same alignments as the pictures in the middle, but the length of the left alignment (GAVEO) is adapted to the length of the right one (Greedy) to increase comparability.

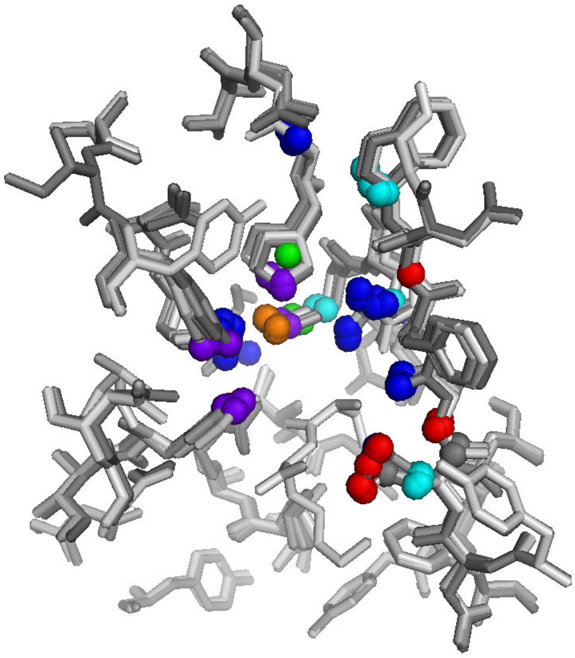


Figure 5: Superposition of six thermolysin cavities showing conserved pseudocenters ($\alpha = 1$, $\beta = 0.9$). Pseudocenter types: donor (red), acceptor (blue), donor/acceptor (purple), aliphatic (cyan), aromatic (green), metal (orange), PI (grey).

interacting with the ion.

References

[1] R. Agrawal and R. Srikant. Fast Algorithms for Mining Association Rules. In *Conference on Very Large Data Bases*, volume 1215, pages 487–499, 1994.

[2] T. Akutsu. Protein Structure Alignment using Dynamic Programing and Iterative Improvement. *IEICE Transactions on Information and Systems*, 79(12):1629–1636, 1996.

[3] N.N. Alexandrov and D. Fischer. Analysis of Topological and Nontopological Structural Similarities in the PDB: New Examples With Old Structures. *Proteins*, 25(3):354–365, 1996.

[4] P.J. Artymiuk, A.R. Poirrette, H.M. Grindley, D.W. Rice, and P. Willett. A Graph-theoretic Approach to the Identification of Three-dimensional Patterns of Amino Acid Side-chains in Protein Structures. *Journal of Molecular Biology*, 243(2):327–344, 1994.

[5] J. Berg and M. Lässig. Local graph alignment and motif search in biological networks. *Proceedings of the National Academy of Sciences of the United States of America*, 101(41):14689–14694, 2004.

[6] P. Bergamini, L. Cinque, A.D.J. Cross, E.R. Hancock, S. Levialdi, and R: Myers. Efficient Alignment and Correspondence using Edit Distance. In *IAPR International Workshops on Advances in Pattern Recognition*, pages 246–255, 2000.

[7] C. Borgelt and M.R. Berthold. Mining Molecular Fragments: Finding Relevant Substructures of Molecules. In *International Conference on Data Mining.* IEEE Computer Society Washington, DC, USA, 2002.

[8] C. Borgelt, T. Meinl, and M. Berthold. MoSS: A Program for Molecular Substructure Mining. In *KDD international workshop on open source data mining: frequent pattern mining implementations*, pages 6–15. ACM Press New York, NY, USA, 2005.

[9] K. M. Borgwardt. *Graph Kernels.* PhD thesis, Ludwig-Maximilians-Universität München, Germany, 2007.

[10] C. Bron and J. Kerbosch. Finding all cliques of an undirected graph. *Communications of the ACM*, 16(9):575 – 577, 1973.

[11] H. Bunke. Error Correcting Graph Matching: On the Influence of the Underlying Cost Function. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(9):917–922, 1999.

[12] H. Bunke, X. Jiang, and A. Kandel. On the Minimum Common Supergraph of two Graphs. *Computing*, 65(1):13–25, 2000.

[13] H. Bunke and K. Shearer. A Graph Distance Metric Based on the Maximal Common Subgraph. *Pattern Recognition Letters*, 19(3-4):255–259, 1998.

[14] Horst Bunke and Xiaoyi Jiang. Graph matching and similarity. *Intelligent systems and interfaces*, 15:281 – 304, 2000.

[15] W.J. Christmas, J. Kittler, and M. Petrou. Structural Matching in Computer Vision using Probabilistic Relaxation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(8):749–764, 1995.

[16] Eric H. Davidson, Jonathan P. Rast, Paola Oliveri, Andrew Ransick, Cristina Calestani, Chiou-Hwa Yuh, Takuya Minokawa, Gabriele Amore, Veronica Hinman, Cesar Arenas-Mena, Ochan Otim, C. Titus Brown, Carolina B. Livi, Pei Yun Lee, Roger Revilla, Alistair G. Rust, Zheng jun Pan, Maria J. Schilstra, Peter J. C. Clarke, Maria I. Arnone, Lee Rowen, R. Andrew Cameron, David R. McClay, Leroy Hood, and Hamid Bolouri. A genomic regulatory network for development. *Science*, 295(5560):1669–1678, 2002.

[17] L. Dehaspe, H. Toivonen, and R.D. King. Finding Frequent Substructures in Chemical Compounds. In *4th International Conference on Knowledge Discovery and Data Mining*, pages 30–36. AAAI Press., 1998.

[18] O. Dror, H. Benyamini, R. Nussinov, and H. Wolfson. MASS: Multiple Structural Alignment by Secondary Structures. *Bioinformatics*, 19(1):i95–104, 2003.

[19] A. Falicov and F.E. Cohen. A Surface of Minimum Area Metric for the Structural Comparison of Proteins. *Journal of Molecular Biology*, 258(5):871–892, 1996.

[20] M. L. Fernández and G. Valiente. A graph distance metric combining maximum common subgraph and minimum common supergraph. *Pattern Recognition Letters*, 22(6-7):753 – 758, 2001.

[21] Holger Fröhlich, , Jörg K. Wegner, Florian Sieker, and Andreas Zell. Optimal assignment kernels for attributed molecular graphs. In *International conference on Machine learning*, pages 225 – 232, Bonn, Germany, 2005.

[22] Thomas Gärtner. A survey of kernels for structured data. *SIGKKD Explorations*, 5(1):49 – 58, 2003.

[23] M. Gerstein and M. Levitt. Using iterative dynamic programming to obtain accurate pairwise and multiple alignments of protein structures. In *International Conference on Intelligent Systems for Molecular Biology*, volume 4, pages 59–67, 1996.

[24] M. Gerstein and M. Levitt. Comprehensive Assessment of Automatic Structural Alignment Against a Manual s´Standard, the Scop Classification of Proteins. *Protein Science*, 7(2):445–456, 1998.

[25] J. F. Gibrat, T. Madej, and S. H. Bryant. Surprising similarities in structure comparison. *Current Opinion in Structural Biology*, 6(3):377–385, 1996.

[26] H.M. Grindley, P.J. Artymiuk, D.W. Rice, and P. Willett. Identification of Tertiary Structure Resemblance in Proteins using a Maximal Common Subgraph Isomorphism Algorithm. *Journal of Molecular Biology*, 229(3):707–721, 1993.

[27] L. Holder, D. Cook, and S. Djoko. Substructure Discovery in the Subdue System. In *AAAI Workshop on Knowledge Discovery in Databases*, pages 169–180, 1994.

[28] L. Holm and J. Park. DaliLite Workbench for Protein Structure Comparison, 2000.

[29] L. Holm and C. Sander. Protein Structure Comparison by Alignment of Distance Matrices. *Journal of Molecular Biology*, 233(1):123–138, 1993.

[30] J. Huan, W. Wang, J. Prins, et al. Efficient Mining of Frequent Subgraphs in the Presence of Isomorphism. In *Third IEEE International Conference on Data Mining*, pages 549–561. IEEE Computer Society Washington, DC, USA, 2003.

[31] Daniel H. Huson and David Bryant. Application of phylogenetic networks in evolutionary studies. *Molecular Biology and Evolution*, 23(2):254–267, 2006.

[32] A. Inokuchi, T. Washio, and H. Motoda. An Apriori-Based Algorithm for Mining Frequent Substructures from Graph Data. In *Fourth European Conference on Principles of Data Mining and Knowledge Discovery*, pages 13–23. Springer-Verlag London, UK, 2000.

[33] M. Jambon, A. Imberty, G. Deleage, and C. Geourjon. A New Bioinformatic Approach to Detect Common 3 D Sites in Protein Structures. *Proteins Structure Function and Genetics*, 52(2):137–145, 2003.

[34] D. Justice and A. Hero. A Binary Linear Programming Formulation of the Graph Edit Distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(8):1200–1214, 2006.

[35] Wolfgang Kabsch. A solution of the best rotation to relate two sets of vectors. *Acta Crystallographica*, 32:922–923, 1976.

[36] M. Kanehisa, S. Goto, S. Kawashima, Y. Okuno, and M. Hattori. The KEGG resource for deciphering the genome. *Nucleic Acids Research*, 32:D277 – D280, 2004.

[37] H. Kashima, K. Tsuda, and A Inokuchi. Marginalized Kernels Between Labeled Graphs. In *20th International Conference on Machine Learning*, pages 321–328, 2003.

[38] T. Kawabata and K. Nishikawa. Protein Structure Comparison Using the Markov Transition Model of Evolution. *Proteins*, 41(1):108–122, 2000.

[39] K. Kinoshita and H. Nakamura. Identification of Protein Biochemical Functions by Similarity Search using the Molecular Surface Database eF-site. *Protein Science*, 12(8):1589–1595, 2003.

[40] K. Kinoshita and H. Nakamura. Identification of the Ligand Binding Sites on the Molecular Surface of Proteins. *Protein Science*, 14(3):711–718, 2005.

[41] G.J. Kleywegt and T.A. Jones. Detecting Folding Motifs and Similarities in Protein Structures. *Methods in Enzymology*, 277:525–545, 1997.

[42] R.I. Kondor and J. Lafferty. Diffusion Kernels on Graphs and Other Discrete Structures. In *19th International Conference on Machine Learning*, pages 315–322, 2002.

[43] Risi Kondor and Karsten M. Borgwardt. The skew spectrum of graphs. In *International Conference on Machine Learning*, pages 496–503, 2008.

[44] M. Kuramochi and G. Karypis. Frequent Subgraph Discovery. In *IEEE International Conference on Data Mining*, pages 313–320, 2001.

[45] M. Kuramochi and G. Karypis. Discovering Frequent Geometric Subgraphs. *Information Systems*, 32(8):1101–1120, 2007.

[46] Y. Lamdan and HJ Wolfson. Geometric Hashing: A General And Efficient Model-based Recognition Scheme. In *Second International Conference on Computer Vision*, pages 238–249, 1988.

[47] N. Leibowitz, R. Nussinov, and H.J. Wolfson. MUSTA-A General, Efficient, Automated Method for Multiple Structure Alignment and Detection of Common Motifs: Application to Proteins. *Journal of Computational Biology*, 8(2):93–121, 2001.

[48] T. Madej, J.F. Gibrat, and S.H. Bryant. Threading a Database of Protein Cores. *Proteins*, 23(3):356–369, 1995.

[49] J. J. McGregor. Backtrack search algorithms and the maximal common subgraph problem. *Software - Practice and Experience*, 12(1):23–34, 1982.

[50] B.T. Messmer and H. Bunke. A new Algorithm for Error-tolerant Subgraph Isomorphism Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(5):493–504, 1998.

[51] BT Messmer and H. Bunke. Error-Correcting Graph Isomorphism using Decision Trees. *International Journal of Pattern Recognition and Artificial Intelligence*, 12:721–742, 1998.

[52] E.M. Mitchell, P.J. Artymiuk, D.W. Rice, and P. Willett. Use of Techniques Derived from Graph Theory to Compare Secondary Structure Motifs in Proteins. *Journal of Molecular Biology*, 212(1):151–166, 1990.

[53] K. Mizuguchi and N. Go. Comparison of Spatial Arrangements of Secondary Structural Elements in Proteins. *Protein Engineering Design and Selection*, 8(4):353–362, 1995.

[54] R. Myers, R.C. Wilson, and E.R. Hancock. Bayesian Graph Edit Distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(6):628–635, 2000.

[55] M. Neuhaus and H. Bunke. A Convolution Edit Kernel for Error-tolerant Graph Matching. In *18th International Conference on Pattern Recognition*, volume 4, pages 220–223, 2006.

[56] C.A. Orengo and W.R. Taylor. SSAP: Sequential Structure Alignment Program for Protein Structure Comparison. *Methods in Enzymology*, 266:617–35, 1996.

[57] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.

[58] A.N. Papadopoulos and Y. Manolopoulos. Structure-based Similarity Search with Graph Histograms. In *10th International Workshop on Database and Expert Systems Applications*, pages 174–178, 1999.

[59] M. Pelillo. A Unifying Framework for Relational Structure Matching. In *14th International Conference on Pattern Recognition*, volume 2, 1998.

[60] L. Ralaivola, S.J. Swamidass, H. Saigo, and P. Baldi. Graph Kernels for Chemical Informatics. *Neural Networks*, 18(8):1093–1110, 2005.

[61] J. Raymond and P. Willett. Maximum common subgraph isomorphism algorithms for the matching of chemical structures. *Journal of Computer-Aided Molecular Design*, 16(7):521–533, 2002.

[62] J.W. Raymond, E.J. Gardiner, and P. Willett. Heuristics for Similarity Searching of Chemical Graphs Using a Maximum Common Edge Subgraph Algorithm. *Jorunal of Chemical Information and Computer Sciences*, 42(2):305–316, 2002.

[63] R.C. Read and D.G. Corneil. The Graph Isomorphism Disease. *Journal of Graph Theory*, 1(1):339–363, 1977.

[64] A. Robles-Kelly and E.R. Hancock. Graph Edit Distance from Spectral Seriation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(3):365–378, 2005.

[65] A. Sanfeliu and K.S. Fu. A Distance Measure Between Attributed Relational Graphs for Pattern Recognition. *IEEE Transactions on Systems, Man and Cybernetics*, 13(3):353–362, 1983.

[66] D.C. Schmidt and L.E. Druffel. A Fast Backtracking Algorithm to Test Directed Graphs for Isomorphism Using Distance Matrices. *Journal of the ACM*, 23(3):433–445, 1976.

[67] D. Shasha, J.T.L. Wang, and R. Giugno. Algorithmics and Applications of Tree and Graph Searching. In *Proc. 21th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, pages 39–52. ACM Press New York, USA, 2002.

[68] M. Shatsky, R. Nussinov, and H.J. Wolfson. A Method for Simultaneous Alignment of Multiple Protein Structures. *Proteins Structure Function and Bioinformatics*, 56(1):143–156, 2004.

[69] M. Shatsky, A. Shulman-Peleg, R. Nussinov, and H. J. Wolfson. The multiple common point set problem and its application to molecule binding pattern detection. *Journal of Computational Biology*, 13(2):407–428, 2006.

[70] John Shawe-Taylor and Nello Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, Cambrigde, 2003.

[71] I.N. Shindyalov and P.E. Bourne. Protein Structure Alignment by Incremental Combinatorial Extension (CE) of the Optimal Path. *Protein Engineering Design and Selection*, 11(9):739–747, 1998.

[72] A.P. Singh and D.L. Brutlag. Hierarchical protein Structure Superposition Using Both Secondary Structure and Atomic Representations. In *International Conference on Intelligence Systems for Molecular Biology*, volume 5, pages 284–293, 1997.

[73] R.V. Spriggs, P.J. Artymiuk, and P. Willett. Searching for Patterns of Amino Acids in 3D Protein Structures. *J. of Chem. Inform. and Comp. Sciences*, 43(2):412–421, 2003.

[74] A. Srinivasan, R.D. King, S. Muggleton, and M.J.E. Sternberg. Carcinogenesis Predictions Using ILP. In *7th International Workshop on Inductive Logic Programming*, pages 273–287. Springer-Verlag London, UK, 1997.

[75] P. N. Suganthan, E. Teoh, and D. Mital. Pattern recognition by graph matching using the potts MFT neural networks. *Pattern Recognition*, 28(7):997–1009, 1995.

[76] W.R. Taylor, T.P. Flores, and C.A. Orengo. Multiple Protein Structure Alignment. *Protein Science*, 3(10):1858–1870, 1994.

[77] Y. Tian, R. C. McEachin, C. Santos, D. J. States, and J. M. Patel. SAGA: A subgraph matching tool for biological graphs. *Bioinformatics*, 23(2):232–239, 2007.

[78] Y. Tian and J. M. Patel. TALE: A tool for approximate large graph matching. In *International Conference on Data Engineering*, pages 963–972, Cancun, Mexico, 2008.

[79] J.R. Ullmann. An Algorithm for Subgraph Isomorphism. *Journal of the ACM*, 23(1):31–42, 1976.

[80] S. Umeyama. An Eigendecomposition Approach to Weighted Graph Matching Problems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(5):695–703, 1988.

[81] Jean-Philippe Vert. The optimal assignment kernel is not positive definite. Technical report, Centre for Computational Biology, Mines Paris Tech, 2008.

[82] G. Vriend and C. Sander. Detection of Common Three-dimensional Substructures in Proteins. *Proteins: Structure, Function and Genetics*, 11(1):52–58, 1991.

[83] Robert A. Wagner and Michael J. Fischer. The string-to-string correction problem. *Journal of the ACM*, 21(1):168 – 173, 1974.

[84] J.T.L. Wang, K. Zhang, and G.W. Chirn. Approximate Graph Matching using Probabilistic Hill Climbing algorithms. In *6th International Conference on Tools with Artificial Intelligence*, pages 390–396, 1994.

[85] X. Wang and JT Wang. Fast Similarity Search in Three-Dimensional Structure Databases. *Journal of Chemical Information and Computer Sciences*, 40(2):442–451, 2000.

[86] Y.K. Wang, K.C. Fan, and J.T. Horng. Genetic-based Search for Error-correcting Graph Isomorphism. *IEEE Transactions on Systems, Man and Cybernetics*, 27(4):588–597, 1997.

[87] S. Wassermann and K. Faust. *Social Network Analysis: Methods and Applications.* Cambridge University Press, 1994.

[88] I. Xenarios, L. Salwinski, X. Duan, P. Higney, S. Kim, and D. Eisenberg. DIP, the database for interacting proteins: A research tool for studying cellular networks of protein interactions. *Nucleic Acids Research*, 30:303–305, 2002.

[89] L. Xu and E. Oja. Improved Simulated Annealing, Boltzmann Machine, and Attributed Graph Matching. In *EURASIP Workshop on Neural Networks*, pages 151–160. Springer-Verlag London, UK, 1990.

[90] X. Yan and J. Han. CloseGraph: Mining Closed Frequent Graph Patterns. In *9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 286–295. ACM Press New York, NY, USA, 2003.

[91] X. Yan, P.S. Yu, and J. Han. Graph Indexing: A Frequent Structure-based Approach. In *ACM SIGMOD International Conference on Management of Data*, pages 335–346. ACM New York, NY, USA, 2004.

[92] X. Yan, P.S. Yu, and J. Han. Substructure Similarity Search in Graph Databases. In *ACM SIGMOD Int. Conf. on Management of Data*, pages 766–777, New York, 2005.

[93] X. Yan, F. Zhu, J. Han, and P.S. Yu. Searching Substructures with Superimposed Distance. In *International Conference on Data Engineering*, volume 88, 2006.

[94] K. Yoshida and H. Motoda. CLIP: Concept Learning from Inference Patterns. *Artificial Intelligence*, 75(1):63–92, 1995.

[95] S. Zhang, M. Hu, and J. Yang. Treepi: A novel graph indexing method. In *23th International Conference on Data Engineering*, pages 966–975, 2007.