

Clique-detection Models in Computational Biochemistry and Genomics

S. Butenko and W. E. Wilhelm

*Department of Industrial Engineering
Texas A&M University
TAMUS 3131
College Station, TX 77843-3131
{butenko,wilhelm}@tamu.edu*

Abstract

Many important problems arising in computational biochemistry and genomics have been formulated in terms of underlying combinatorial optimization models. In particular, a number have been formulated as clique-detection models. The proposed article includes an introduction to the underlying biochemistry and genomic aspects of the problems as well as to the graph-theoretic aspects of the solution approaches. Each subsequent section describes a particular type of problem, gives an example to show how the graph model can be derived, summarizes recent progress, and discusses challenges associated with solving the associated graph-theoretic models.

Clique detection models include prescribing (a) a maximal clique, (b) a maximum clique, (c) a maximum weighted clique, or (d) all maximal cliques in a graph. The particular types of biochemistry and genomics problems that can be represented by a clique detection model include integration of genome mapping data, nonoverlapping local alignments, matching and comparing molecular structures, and protein docking.

Key words: computational biochemistry, genomics, optimization, clique-detection

1 Introduction

Recent progress in biochemistry and genome studies has opened up a window of new opportunities for interdisciplinary research related to biotechnology. Massive data sets generated in biochemistry and genome sequencing research entail a number of computational challenges, some of which can be approached using operations research and optimization techniques. In particular, many important problems arising in computational biochemistry and genomic analysis

can be formulated in terms of certain combinatorial optimization problems in specially constructed graphs. This paper discusses several problems of this type.

1.1 Recent reviews

Abbas and Holmes (2004) provided an overview of methods that are common to management science and bioinformatics, which they defined as “the application of mathematical, statistical, and computational tools in the analysis of biological data.” They gave a “crash course” on molecular biology and reviewed (selected) quantitative methods that had been applied to three types of problems in the realm of molecular biology: sequence alignment (for a pair of either DNA or amino acid sequences); phylogenetic trees; and protein folding, simulation, and structure prediction. In another survey, Greenberg et al. (2004) discussed combinatorial optimization models arising in sequencing, evolutionary explanations, structure prediction and recognition. Finally, Blazewicz et al. (2004) gave a brief primer on the biology of DNA and reviewed (selected) research in the field of computational biology related to DNA studies: sequencing DNA chains, assembling DNA chains, mapping genomes, comparing sequences, and analyzing phylogenetic relationships.

This paper complements these recent reviews; it deals with a different set of models and specific problem types. It is somewhat more focused in the sense that it concentrates on similar types of models, but, at the same time, these models are applied to a diverse set of problems arising in an important research area. The primary overlap between this paper and the three review papers is that each includes a primer on the underlying biochemistry and genomic aspects of the problems it addresses, so that each paper is self-contained.

1.2 Organization of the paper

The remainder of this paper is organized as follows. Section 2 contains a brief tutorial on molecular biology. Section 3 gives the necessary background information from graph theory. Section 4 discusses the integration of genome mapping data. Section 5 deals with the problem of finding nonoverlapping local alignments arising in the study of genome rearrangements. Section 6 addresses mapping three-dimensional molecular structures. Finally, Section 7 relates conclusions.

2 Background in molecular biology

2.1 Deoxyribonucleic acid (DNA)

Some of the most important and spectacular advances of biology in the last century are associated with molecular genetics and, in particular, with the discovery of the nature of genetic material. In 1868, a Swiss physician, Fritz Miescher, first discovered *deoxyribonucleic acid* (DNA) in cell nuclei. It was only eighty five years later when Watson and Crick (1953) published their model of the DNA molecule in *Nature*. Their model, which has been validated by subsequent experiments, is that the DNA molecule comprises two polymer chains, each made of four types of residues (bases) called *nucleotides*: A (adenine), G (guanine), T (thymine), and C (cytosine). A (T) in one chain is always opposite T (A) in the other. Similarly, C (G) is always opposite G (C). Strong covalent bonds link atoms inside a polymer chain, and the two complementary strands interact through weak (intermolecular) forces. Fig. 1 sketches a thymine nucleotide and the two DNA base pairs. Each DNA strand forms a helical line twisted right-handedly and a DNA molecule forms a double helix line twisted right-handedly. Nitrogenous bases form the core of the molecule, which has a coating comprising negatively charged phosphate groups. Base pairs are perpendicular to the axis of the double helix, like rungs on a helically twisted ladder.

The distance between adjacent base pairs along the helix is 0.34 nanometers (nm) and the diameter of the double helix is 2 nm. The double helix makes a complete turn every 10 base pairs. If a DNA molecule were straightened out, it would be about seven feet long! Thus, packing a DNA molecule into a cell is no small feat. To accomplish this, the double helix wraps around a set of nuclear proteins (histones). A molecule winds around a “spool” (nucleosome) twice, then around another spool, and another, and so on, forming a “necklace” of nucleosomes. Necklaces are packed into small bodies called *chromosomes*, which comprise DNA and the proteins. Human genome comprises 24 chromosomes.

The DNA of an individual is formed when the *gametes* of father and mother merge to form a *zygote* (fertilized egg cell). This single DNA molecule replicates at each cell division so that each cell includes a DNA molecule that is identical to those in all other cells of the individual. The Watson-Crick model is used to explain the mechanism of DNA replication. Given one strand of a DNA helix, it allows for an easy reconstruction of the *complementary* strand. Namely, DNA is replicated by separating its two strands and supplementing each with a complementary strand with the correct code. This results in two, identical DNA molecules. Figure 2 illustrates the double helix structure of

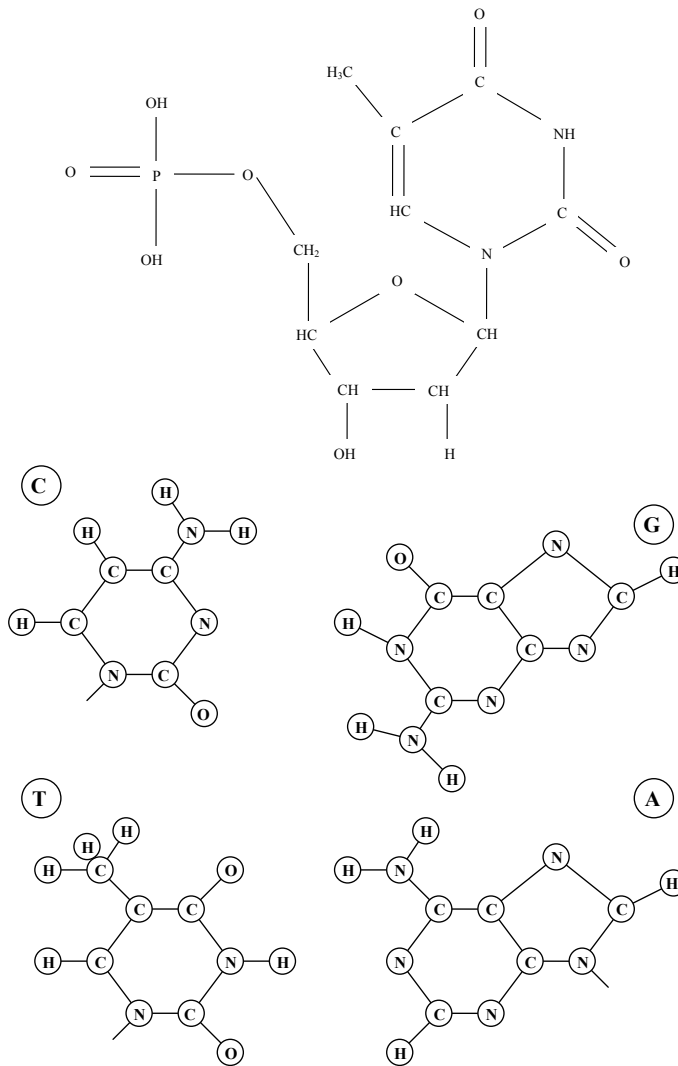


Fig. 1. The top part of the figure illustrates a thymine nucleotide (thymidine monophosphate). The other three nucleotides have a similar structure, but each has a different nitrogenous base (the top group). These differences can be observed in the bottom part of the figure illustrating the DNA base pairs (Frank-Kamenetskii (1997)).

DNA and the mechanism of its replication.

The DNA molecule is central to a broad spectrum of modern techniques for medical analysis and testing. Recent breakthroughs in biotechnology make it possible to reproduce DNA, fragment it, determine its composition, and experiment with its structure. A technology called *polymerase chain reaction (PCR)* plays an essential role in DNA analysis. PCR is a chain of chemical reactions that can generate millions of copies of a DNA fragment in just a few hours. A relatively small (a few thousand bases in length), single-stranded DNA molecule called a *DNA probe* is used to detect a target, complementary DNA strand in a large DNA molecule. The presence of multiple copies of the

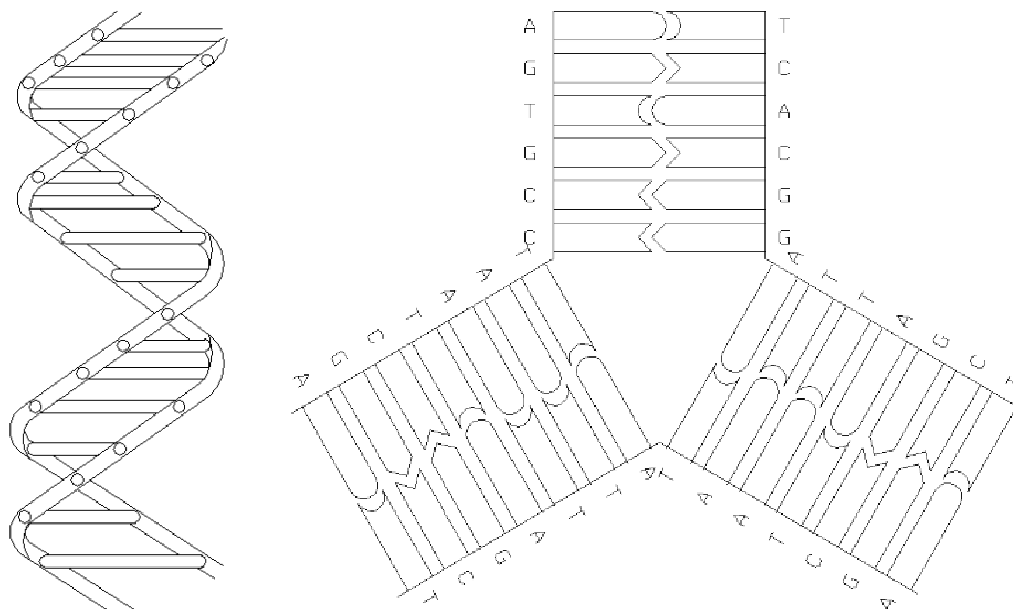


Fig. 2. DNA ladder and its replication (Frank-Kamenetskii (1997)).

DNA obtained from PCR ensures the reliability of the DNA probe test.

The analysis of DNA starts with reading the DNA sequence. This process usually comprises three stages: *mapping*, *assembling*, and *sequencing*. Since the entire DNA of a human organism (also known as *genome*) consists of billions of nucleotides, it cannot be read directly. Therefore, the long DNA sequence is cut using restriction enzymes into smaller pieces of length $\approx 10^5 - 10^6$ nucleotides. When DNA is cut, the information about the order of resulting fragments is lost. To recover this information, mapping procedures are used. After mapping, the pieces are cut again into parts consisting of several tens of thousands of nucleotides. Another mapping procedure is required to recover the order of these parts. Yet again, these parts are randomly broken into sequences of length no greater than 1000 nucleotides. The resulting DNA fragments have length acceptable for sequencing. After sequencing, the short sequences must be ordered to form the original fragment from which they were obtained. This is done using an assembling process.

For a continuing explanation of the structure and functions of DNA the reader is referred to Frank-Kamenetskii (1997).

2.2 Proteins

It is known that genes encode instructions for the production of *proteins*. Proteins are large, complex molecules, which play a vital role in the structure and function of the organism. The human genome consists of 3 billion bases,

but only 30,000-40,000 genes code for proteins. In 1948, George Gamow, a theoretical physicist, conjectured that the universe was created by the big bang. In 1954, he conjectured that proteins are the cell's primary working molecules. It was not long until scientists proved that proteins are, in fact, responsible for all chemical transformations in the cell. Thus, the proteins in a cell determine the work that a cell can do.

Proteins are built up from 20 different amino acids, each of which comprises a central (*alpha*) carbon atom bonded to an amino group (NH_2), a carboxyl group ($COOH$) and a side chain, one of 20 residues, that is unique for each amino acid and determines its characteristics such as size, shape, and polarity (Chandru et al. (2003)). To form a protein, amino acids are linked by *peptide bonds*, creating a *backbone* of alpha carbon atoms with tight constraints on bond angles and *side chains* comprising residues. The sequence of amino acids forms the *primary structure* and combinations of simple *motifs* (e.g., α -helices and β -sheets) form *secondary structures*. Proteins *fold* into specific, 3D *native structures* (*tertiary structures*), which are thought to be minimum-energy configurations (*conformations*) that determine their properties and functions. Helper molecules called *chaperones* may facilitate the folding of some proteins.

The same set of amino acids is used in all life forms on earth! It is the sequence of amino acid residues in its proteins that distinguishes one living thing from another. The structure of amino acids is $H_2N - CHR - COOH$, in which R represents a radical. The structure of an amino acid residue, which is the remaining form of an amino acid after it is incorporated in a protein chain, is $HN - CHR - CO$.

The “central dogma” of molecular biology states that the sequence of nucleotides in one of the two complementary DNA strands determines the amino acid sequences of all cell proteins. Over its length, the DNA molecule is a continuous sequence of nucleotides (A, C, G, T). A *gene* is the sequence of nucleotides in a section of the DNA strand that encodes the amino acid sequence of one protein. Certain nucleotides act as “punctuation marks”, setting off each gene. A gene gives rise to a protein by a two-stage process in which *ribonucleic acid* (RNA) plays a crucial role. RNA is a single-stranded polymer chain comprising nucleotides (bases) of four types: A, C, G and U (uridine).

During the first stage, *transcription*, a particular enzyme (called *RNA polymerase*) recognizes the sequence of nucleotides between genes (the *promoter* section of the DNA strand) and moves along the strand, making a copy of it in the form of an RNA molecule. The entire length of the strand must be accessible to RNA polymerase so that it can transcribe a gene to create a *messenger RNA* (mRNA). The second stage, *translation*, is an intricate process that involves a host of protagonists, including the cell's *ribosome*, a complex “machine” comprising some 50 different proteins and a molecule of RNA, *ribo-*

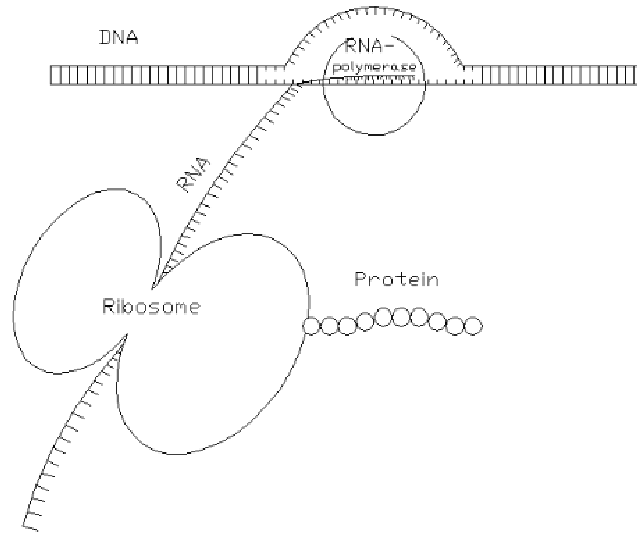


Fig. 3. Protein making: RNA polymerase moves along DNA and synthesizes *mRNA*. The information from *mRNA* is copied by the ribosome which synthesizes protein (Frank-Kamenetskii (1997)).

somal RNA (rRNA). The ribosome translates the mRNA code into the amino acid sequence of the corresponding protein. The mRNA strand comprises a series of three-nucleotide sequences, each of which is called a *codon* and encodes one amino acid residue of the protein. Combinations of the four bases (A, C, G, U) give rise to $4^3 = 64$ different codons. Some, *terminal codons*, serve as signals that end the protein chain. If two codons have the same nucleotides in the first two positions and the third nucleotide is from the same class (pyrimidine or purine) they encode the same amino acid (*pyrimidines* comprise (U and C); and *purines*, (A and G)). Thus, the 64 codons encode the 20 different amino acids.

The structure of proteins is determined using X-ray crystallography and nuclear magnetic resonance (NMR) spectroscopy. Even though X-ray crystallography and NMR-spectroscopy are too slow to analyze a large number of proteins, modern technologies allow the sequences of proteins to be identified quickly and inexpensively (Heun (2003)). Thus, predicting how a protein will fold knowing only the sequence of amino acids that comprise it is an important area of research.

3 Graph-theoretic background

We will use the following definitions and notation. Denote by $G = (V, E)$ a simple undirected graph with the set of n vertices V and the set of edges $E \subseteq V \times V$. For a vertex $u \in V$, let $N(u) = \{v : (u, v) \in E\}$ denote the set of its neighbors.

Given a subset $S \subseteq V$, by $G(S) = (S, (S \times S) \cap E)$ we denote the *subgraph induced by S*. A subset $C \subseteq V$ is a *clique* if $G(C)$ is a complete graph; *i.e.*, it has all possible edges. The maximum clique problem (MCP) is to find the largest clique in a graph. The cardinality of a maximum clique in G is called the clique number of G and is denoted by $\omega(G)$. The maximum clique problem is closely related to another discrete optimization problem, the *maximum independent set* problem (MISP), which is defined as follows. An *independent set* (*stable set*, *vertex packing*) is a subset of V whose elements are pairwise nonadjacent. The maximum independent set (MIS) problem is to prescribe an independent set of maximum cardinality. The size of a maximum independent set is the *stability number* of G (denoted by $\alpha(G)$). Let us denote by $\bar{G} = (V, \bar{E})$ the *complement* of graph G , where $\bar{E} = \{(i, j) \in V^2 : (i, j) \notin E\}$. Then it is easy to see that a set $S \subseteq V$ is a clique in G if and only if S is an independent set in \bar{G} . Therefore, the maximum clique problem is equivalent to the maximum independent set problem in the complement graph.

The maximum clique problem is NP-hard (Garey and Johnson (1979)), so it is unlikely that a polynomial-time algorithm for computing the maximum clique of an arbitrary graph can be devised. Moreover, this problem is associated with a series of recent results about hardness of approximations. Arora and Safra (1992) proved that, for some positive ϵ , the approximation of the maximum clique within a factor of n^ϵ is NP-hard. Håstad (1999) has shown that, in fact for any $\delta > 0$, the maximum clique is hard to approximate in polynomial time within a factor $n^{1-\delta}$. All of the above facts, together with practical evidence (Johnson and Trick (1996)) suggest that, in the worst case, the maximum clique problem is hard to solve even in graphs of moderate sizes.

A clique C in G is called *maximal* if there is no larger clique U such that $C \subset U$. We will refer to the problem of finding all maximal cliques as the *all maximal cliques detection problem* (AMCDP). AMCDP is obviously no easier than the maximum clique problem. In fact, the number of maximal cliques of a graph may be exponential with respect to the number of vertices (Moon and Moser (1965)).

Many diverse application areas involve MCP and AMCDP problems. Among the numerous application areas of these problems are telecommunications (Abello et al. (1999)), coding theory (Butenko et al. (2004)), economics (Avondo-Bodeno (1962)), finance (Boginski et al. (2003)), register allocation, frequency assignment, and fault tolerance (Bomze et al. (1999)). This review concentrates on discussion of large-scale clique-detection problems arising in computational biochemistry and genomic analysis. Next, we describe some of such problems.

4 Integration of genome mapping data

Methods for mapping genome data, as introduced in Section 2, can be classified into two categories: overlap-based and probe-based. Due to differences in the methods used for analyzing overlap and probe data, the integration of these data becomes an important problem. It appears that overlap data can be effectively converted to probe-like data elements by finding maximal sets of mutually overlapping clones, which can be viewed as cliques in the overlap graph (Harley et al. (1999)).

Recall that in the process of mapping, the DNA is cut into small fragments called clones. The clones are replicated and checked for common base subsequences to physically map how the fragments are organized in the DNA sequence. In overlap-based methods, the clone overlaps can be modelled using *interval graphs*. Each vertex in an interval graph represents an interval, an edge represents a pair of intervals that have a nonempty intersection, and the mapping data indicate overlapping pairs of clones.

In cosmid contig mapping of the human genome, some of the clones are labeled radioactively, creating *probes*. This gives two types of elements to be mapped in probe-based methods – clones and probes. The mapping data are given by an incidence matrix, indicating which clones contain which probes. Again, these data can be modelled using a graph. The vertices of the probe interval graph can be partitioned into two sets: V_1 , representing probes and V_2 , representing clones not used as probes. Each edge connects a pair of nodes that represent overlapping intervals with the provision that at least one node must represent a probe. McMorris et al. (1998) showed that probe interval graphs are perfect (*i.e.*, the chromatic number and the clique number are equal for each induced sub-graph) and characterized probe interval graphs by a pan-consecutive ordering of their intrinsic cliques. A clique C of G is an intrinsic clique if C is a maximal clique of $G(V_1)$ or a maximal clique of $G(V_1 \cap N(u))$ where u is a nonprobe.

Harley et al. (2001) devised a method to analyze probe- and overlap-based methods using the *virtual probe*, the region of the genome common to a maximal set of mutually overlapping clones. Their method, which is based on the assumption that data contain no errors (*i.e.*, all overlaps are known with no false-positive pairs), identifies each virtual probe as a maximal clique in a graph with vertices that represent clones and edges that represent pairs of overlapping clones. Figure 4 illustrates this approach. Part (a) of Figure 4 depicts a chromosome; part (b) – cloned fragments that are known (*i.e.*, by experiment) to overlap. Part (c) of Figure 4 shows an *overlap graph* in which each edge indicates an overlap between the associated pair of clones.

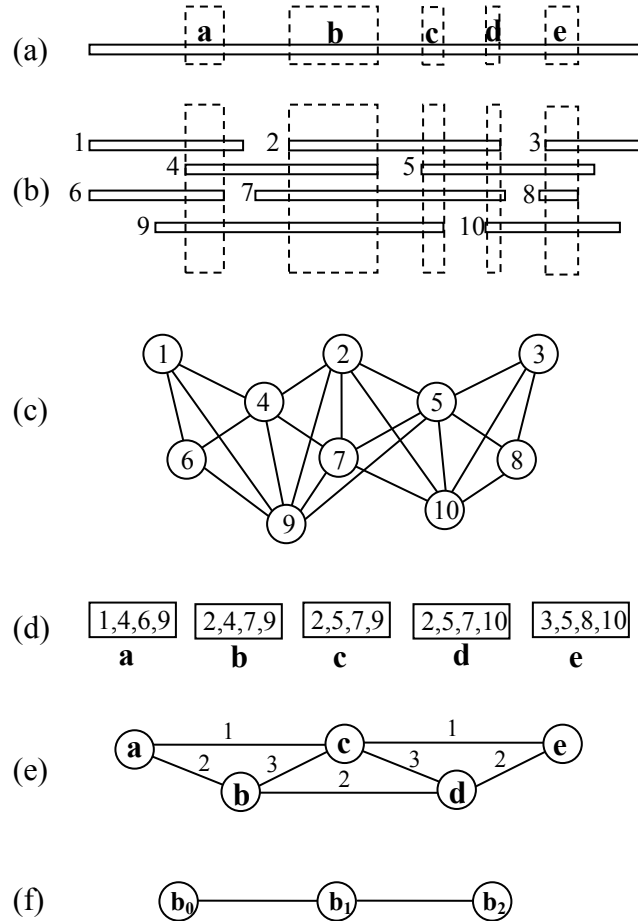


Fig. 4. Illustration to the model of Harley et al. (2001): (a) chromosome, (b) cloned fragments, (c) overlap graph, (d) maximal cliques, (e) clique intersection graph, and (f) structure graph.

Overlap graphs are typically large and sparse. Harley et al. (2001) devised a modification of the Bron-Kerbosch algorithm in which maximal cliques are found only among each vertex i and its neighbors j with $j > i$ to find all maximal cliques in the overlap graph. As an example, Figure 4 (d) lists the set of maximal cliques in the overlap graph. Each clique represents a set of clones that overlap in regions called virtual probes, which mark sites in the genome. Reflecting the fact that the genomic regions of two virtual probes cannot overlap, one maximal clique cannot be contained in another. Figure 4 (e) constructs a *clique-intersection graph* in which each vertex represents a maximal clique and each edge connects two cliques that have a subset of clones in common. Each edge is given a weight that indicates the number of clones in common. Harley et al. (2001) eliminate edges with weights of 1 to reduce the number of edges that arise from false positives as well as the effects of chimerism (Arratia et al. (1991)); this results in a *double-linkage* intersection graph. They also ignore cliques of just two clones to filter out false positives. Finally, they form a *structure graph* (see also Harley et al. (1999)) by a two-step procedure that starts at an arbitrary vertex in the clique-intersection

graph; identifies the vertex that is farthest from it; and then partitions the clique-intersection graph, forming layers of vertices according to their distance from the identified vertex. Vertices that form a connected component within a layer are called a *blob* and a vertex represents each blob in the structure graph. Each edge in the structure graph represents a pair of blobs that have elements that are connected by an edge in the intersection graph. When data are perfect, the structure graph will be a simple path. The structure graph for the considered example is shown in Figure 4 (f). To construct it, we start at vertex **d** of the intersection graph and detect the furthest vertex **a**, which forms a blob represented by **b₀** in the structure graph. Vertices **b₁** and **b₂** correspond to the blobs **{b, c}** and **{d,e}**, respectively.

Harley et al. (2001) invoked the rule of assigning a virtual probe to a particular chromosome if the number of clones associated with that chromosome is at least two and greater than the number associated with any other chromosome. They described tests of their approach using Alu-PCR data; fingerprint data; and the integration of STS, Alu-PCR, and fingerprint data. In the first of these tests, they identified 34,643 maximal cliques, including 20,285 maximal cliques of size two, which they discarded. The clique intersection graph, which comprised 14,358 vertices, allowed identification of vertices that are equally associated with several chromosomes as well as blobs that do not have two or more clones associated with a single chromosome.

5 Nonoverlapping local alignments

Evaluation of the local and global similarity between two genetic sequences is a fundamental problem arising in genome analysis. To determine to what extent two given sequences are alike, a certain “distance metric” is usually used. Knowledge of the “distance” between two DNA or amino acid sequences may help to understand the nature of evolutionary relationship between the two sequences.

The problem of optimal sequence alignment can be stated as follows. Given two strings of letters, insert dashes into the strings so that the similarity between the sequences is maximized. The similarity is evaluated using a given scoring function. For example, the two sequences X=CTGCAT and Y=TGTGCCAGT can be aligned as follows:

```
-CTGC-A-T
TGTGCCAGT
```

To score this alignment, consider a scoring scheme that compares the corresponding positions and assigns a score of 3 if the two characters are identical,

1 if the characters are different, and 0 if a character corresponds to a dash. Using this scheme would give the score of $3(5) + 1(1) + 0(3) = 16$ for the above alignment. The problem of computing optimal sequence alignment is typically approached using dynamic programming (Abbas and Holmes (2004); Clote and Backofen (2000); Needleman and Wunsch (1970); Smith and Waterman (1981)).

The studies on *genome rearrangements* deal with more large-scale changes in the sequences, such as a transposition transferring a block of a genomic sequence to another position. Such rearrangements have been observed between several species in their evolutionary history (Pevzner (2003)). One of the problems arising in genome rearrangement research is the following. Given two sequences and a set of disjoint fragments (genes) occurring in both strings, including their relative order and orientation, determine a (minimal) set of operations transforming one sequence to another. If the number of fragments that appear (possibly with some changes) in both sequences is sufficiently large, solving the formulated genome rearrangement problem allows one to hypothesize the evolutionary process of mutations that has led to the deviation between the analyzed sequences. To select the fragments for comparison, one of the standard local alignment methods can be used. Since rearrangements act on disjoint pieces of the sequence, overlapping fragments cannot be separated, therefore it is important that the compared fragments do not overlap in any of the two sequences. However, the substrings selected using local alignment methods may overlap so that they would not be suitable for testing on rearrangements. Therefore, one needs to select a subset of fragments such that no two of them overlap in any of the two analyzed sequences. The problem of selection of such a subset can be stated in terms of the maximum (weighted) independent set problem in the graph, which has vertices corresponding to the fragments and edges corresponding to pairs of overlapping fragments. The vertices may be weighted with the similarity scores characterizing the strength of the local similarity.

It is rather convenient to represent this problem geometrically by drawing the fragments of local similarity as axis-parallel rectangles in the plane where the two compared sequences serve as the axis. Indeed, a pair of subsequences of high local similarity can be thought of as the rectangle formed by the intersection of the vertical and horizontal slabs corresponding to these subsequences (Fig. 5). The weights of rectangles are defined similarly to the weights of vertices above. Then the problem of interest is to find a maximum-weight set of rectangles whose projections in both axes are pairwise disjoint. Bafna et al. (1996) refer to the recognition version of this problem as the *independent subset of rectangles* (IR). With this representation, the problem can be equivalently viewed as the maximum weight independent set problem in a conflict graph of nonoverlapping axis-parallel rectangles. Bafna et al. (1996) proved that IR is NP-complete even in the uniform case (all weights are the same). Moreover,

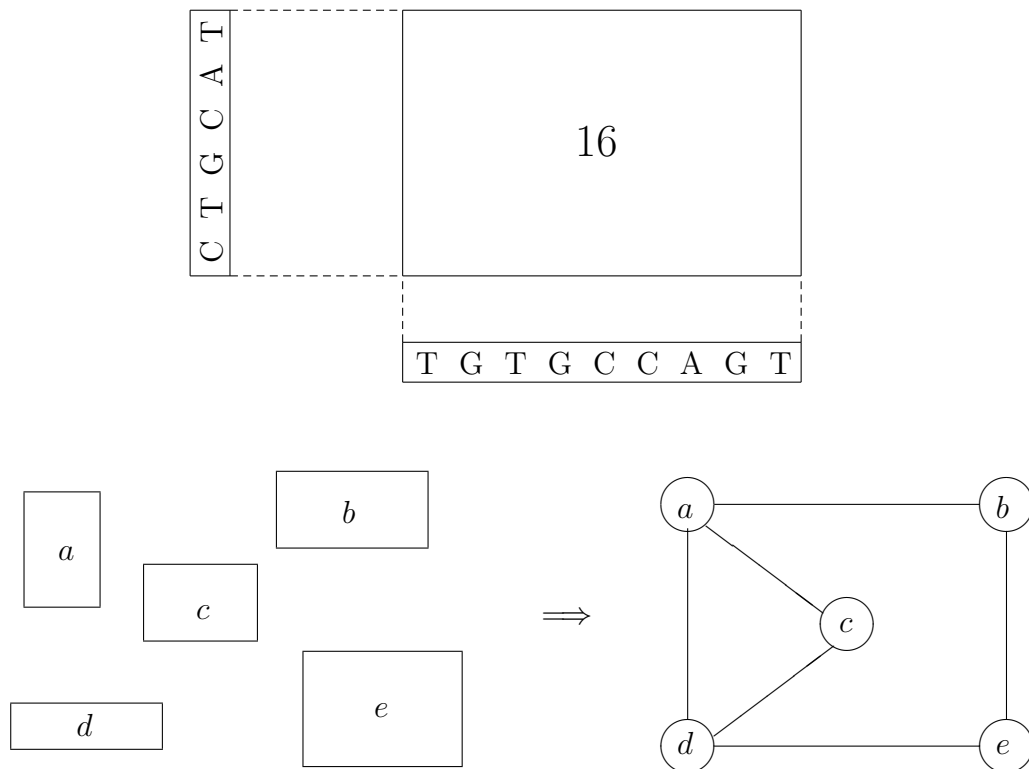


Fig. 5. An illustration to IR problem and its graph representation

they observe that a conflict graph of nonoverlapping axis-parallel rectangles is 5-claw-free¹ and use this fact to derive a local-improvement algorithm with performance ratio of 3.25 (in general, their algorithm provides a tight performance ratio of $d - 1 + 1/d$ for the maximum-weight independent set problem in $(d + 1)$ -claw-free graphs). The performance ratio of the algorithm is $d/2$ in the uniform case (see also Berman (2000)).

6 Matching three-dimensional molecular structures

Two graphs, G_1 and G_2 , are called isomorphic if there exists a one-to-one correspondence between their vertices, such that adjacent pairs of vertices in G_1 are mapped to adjacent pairs of vertices in G_2 . A common subgraph of two graphs G_1 and G_2 comprises subgraphs G'_1 and G'_2 of G_1 and G_2 , respectively, such that G'_1 is isomorphic to G'_2 . The largest such common subgraph is the maximum common subgraph (MCS). For a pair of three-dimensional (3D) chemical molecules the MCS is defined as the largest set of atoms that have matching distances between atoms (given user-defined tolerance values). It

¹ A d -claw is a the graph $K_{1,d}$, *i.e.*, a star with d leaves.

can be shown that the problem of finding the MCS can be solved effectively using clique-detection algorithms applied to a correspondence graph (Gardiner et al. (1997)). For a pair of graphs, $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$, their correspondence graph C has all possible pairs (v_1, v_2) , where $v_i \in V_i, i = 1, 2$, as its set of vertices; two vertices (v_1, v_2) and (v'_1, v'_2) are connected in C if the values of the edges from v_1 to v'_1 in G_1 and from v_2 to v'_2 in G_2 are the same.

Gardiner et al. (1997) evaluated the efficacies of five clique-detection algorithms in identifying the maximum common sub-graph (MCS) in graphs representing two 3D molecules. They noted the importance of this clique-detection approach, especially in pharmaceutical and agricultural applications. They constructed a correspondence graph with vertices representing structural equivalences between the new sequence (target) and a related sequence with a known structure (parent or template). For example, in 3D protein structures, a vertex represents a common α -helix or β -sheet SSE; and an edge represents inter-SSE angle and distance.

Brint and Willett (1987) evaluated five clique-detection algorithms and found the Bron-Kerbosch (1973) algorithm to be best. Gardiner et al. (1997) compared that algorithm with four others (Babel (1991); Balas and Yu (1986); Carraghan and Pardalos (1990); Gendreau et al. (1988), and Shindo and Tomita (1990)) in prescribing all cliques, maximal cliques, and maximum clique(s). Test instances involved both small molecules and pairs of proteins. The Carraghan-Pardalos algorithm proved best in finding the maximum clique(s) but the Bron-Kerbosch algorithm was best in enumerating all maximal cliques. This prompted Gardiner et al. to propose a hybrid method that utilized the Carraghan-Pardalos algorithm as a screen to determine if it is worthwhile to apply the more time-consuming Bron-Kerbosch algorithm. Their hybrid method doubled search speed in their tests. Gardiner et al. noted that chemical and biological correspondence graphs tend to have low edge density, a characteristic that may favor certain algorithms.

6.1 Comparative modeling of protein structure

“The comparison of proteins, either in sequence or structure, is the most fundamental technique in computational biology” (Koike et al. (2004)). It is important to be able to predict the three-dimensional structure of a protein-binding region so that the function of a protein can be annotated and the binding mode between a protein and its ligand can be predicted. This knowledge is key, for example, in accurately designing drugs that bind in a desired manner (de Weese-Scott and Moulton (2004)).

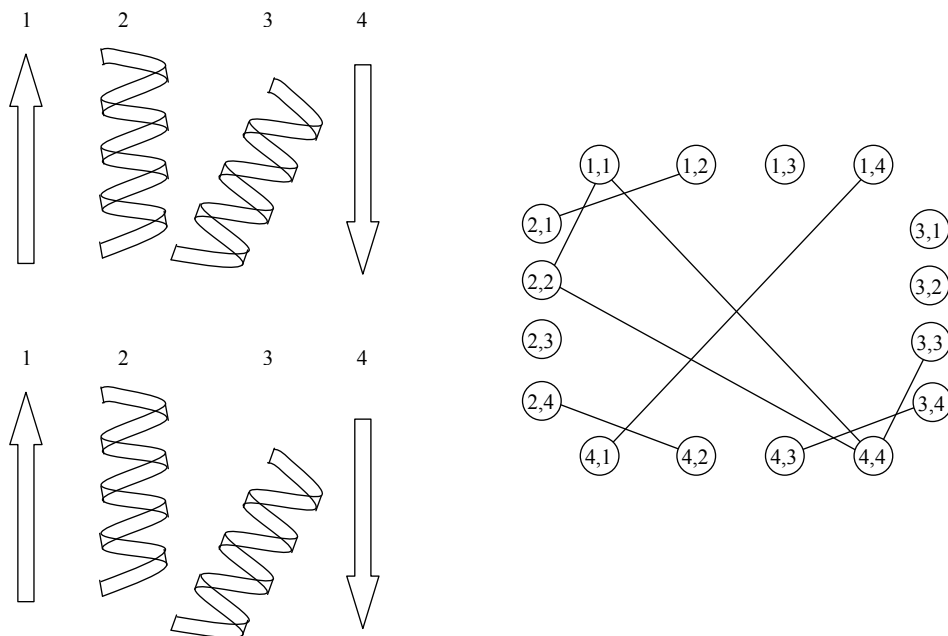


Fig. 6. Illustration of a correspondence graph, in which the maximum clique corresponds to the maximal common subgraph between the two schematic protein structures. Here coils represent helices and arrows represent strands. The SSEs in each protein structure are numbered and the correspondence graph is constructed based on the equivalence (within a specified tolerance) of inter-SSE distances and angles. The maximum clique $\{(1, 1), (2, 2), (4, 4)\}$ in the graph corresponds to the largest group of SSEs that are spatially similar in both protein structures ($\{1, 2, 4\}$ and $\{1, 2, 4\}$, respectively).

Grindley et al. (1993) proposed the PROtein Topographic Exploration Program (PROTEP), which rapidly compares 3D protein structures to identify common patterns of secondary structure elements (SSEs), including angular and spatial relationships. The ultimate goals of PROTEP are to identify common functional relationships, improve understanding of protein folding, and gain insights into the evolution of related structures. PROTEP works with descriptions of proteins that have been obtained by X-ray crystallography or nuclear magnetic resonance (NMR) and then stored in the Protein Data Bank (PDB). It extends a program developed previously by the authors, Protein Online Sub-structure Searching-Ullmann Method (POSSUM), which searches for super-secondary-structure motifs using a (related) sub-graph isomorphism algorithm.

PROTEP implements a four-step procedure. First, it constructs a “connection table,” which lists all non-hydrogen atoms and bond information, describing exactly how individual atoms are linked together in the structure of a protein. Hydrogen atoms within the molecule can be deduced from bond orders and types of non-hydrogen atoms. Second, it represents each table as a graph in which vertices represent objects; and edges, relative position of individual structural elements (not relationships). This approach represents SSEs

as linear axes and compares them relative to distances and interline angles. Third, it forms a correspondence graph. Starting with the graphs representing the structures of the two proteins, define a vertex for each pair of secondary structures of the same type (*i.e.*, both may be α -helices or β - sheets), then incorporate an edge to connect each pair of vertices that represent a pair of secondary structures for which the interline angle and separation distance are below pre-determined thresholds (e.g., $\pm 30^\circ$ or $\pm 5\text{\AA}$) and the directions of the two edges in the protein graphs are the same (*i.e.*, the sequence ordering along the polypeptide chain is important to the match. Fig. 6 illustrates a correspondence graph. Fourth, identify the maximum common sub-graph in the two protein graphs by prescribing a maximum clique in the correspondence graph. PROTEP did not represent the first use of clique-detection algorithms (e.g., see Ghose and Crippen (1985); Golender and Rozenblit (1983); Kuhl et al. (1984), and Takahashi et al. (1987)), but it is effective.

PROTEP employs a set of five parameters: ANGTOL and DISTOL give the angular and distance thresholds, respectively; SEQORD indicates if structural order of SSEs is important; and DISTYP specifies the distance measure (closest approach distance (CAD), midpoint distances (MD), or both types (BD)). Clique-detection algorithms may prescribe small cliques that may not indicate strong structural relationships and are, therefore, not of interest. The parameter MINCLQ indicates the size of the smallest clique of interest so that the algorithm will not report smaller cliques. Low ANGTOL and DISTOL thresholds allow closely related patterns to be identified. Precision decreases (*i.e.*, cliques identify matches that actually exhibit little structural resemblance) rapidly with increases in thresholds.

Tests compared pairs of whole proteins as well as substructures representing important structural or functional motifs. PROTEP searched databases of 371 and 510 proteins within minutes, identifying structural matches independent of the sequence homology or number of insertions and deletions incorporated in each protein.

Samudrala and Moult (1997) devised a clique-detecting approach to identify interconnected structural changes in the comparative modeling of protein structures, focusing on the fact that residues have different main chain and side chain conformations. They formed a graph to model each protein, using a vertex to represent each possible conformation of each residue and incorporating an edge between a pair of vertices if there were no clashes between atoms of the two residues and the interaction between them was covalently acceptable, and if the two residues were within one main-chain region and they were connected by a single covalently linked main-chain conformation. A clash occurs if two non-hydrogen atoms, one in each of two different residues, have a contact of less than 2.0\AA . Samudrala and Moult did not evaluate contacts between pairs of atoms in the main chain for clashes and did not incorporate

an edge between two vertices representing a residue in a side chain and one in the main chain if the interaction weight was extremely positive. Finally, they did not incorporate an edge between any pair of vertices representing two different conformations of the same residue.

They assigned a weight to each vertex based on the strength of the interaction in pairs of atoms between the residue side chain and the local main chain and a weight to each edge based on the strength of interaction between pairs of atoms in the two residues. They calculated vertex and edge weights using an all-atom distance-dependent conditional probability-based discriminatory function, which assigned scores related to the probability of observing a native conformation, given a set of distances between specific atom types.

Cliques of the size of the target structure represent self-consistent arrangements for individual amino acid conformations in this graph. After searching for maximal cliques in this graph, they selected the clique with the highest weight as the correct conformation.

In a related paper, Samudrala and Moult (1988) extended their approach for comparative modeling of protein structures. Their objective was to determine if the sequence of a new protein is related to one or more known proteins. By the mid 1990s, databases allowed some 30% of new proteins (Schneider and Sander (1996)) and 10% of genome sequences (Scharf et al. (1994)) to be matched with known structures. The goal of such modeling is to find the best set of interactions in a protein structure given a variety of side chain and main chain conformational choices for each position in the structure.

Again, they formed a graph using a vertex to represent each possible conformation of a residue in an amino acid sequence with weight determined by the degree of interaction between its side chain atoms and the local main chain atoms. They incorporated an edge between each pair of vertices that represented conformations that were consistent with each other (*i.e.*, those that were clash free and satisfied certain constraints on geometry) and weighted each edge according to the interactions between the atoms of the two conformations. They did not incorporate edges between pairs of possible side chain conformations of the same residue. Their approach maintained packing consistency by incorporating edges between vertices whose atoms do not clash with each other and main chain consistency by partitioning the complete protein main chain conformation into segments, each of which may have one or more conformations. If two vertices represent the same main chain segment, both must be part of the same segment conformation.

Cliques with best weights identify the optimal combinations of the various main chain and side chain possibilities. Samudrala and Moult claimed that their approach offered three primary advantages: (1) a simple framework

for analysis, (2) control over the sub-conformations included, and (3) pre-calculation of the fitness of each vertex to reduce run time.

They applied the Bron and Kerbosch (1973) algorithm to identify maximal cliques in graphs with up to 30,000 vertices within a 24-hour run time using an SGI Challenge R10000 workstation. In each case, they retained the 100 best cliques and then selected the clique with the best scoring conformation as the correct structure.

Martin et al. (1993) devised software called DIStance COmparison (DISCO) that used a clique-detection algorithm to find the maximal common sub-structure (MCS) that is common to all input molecules. Their goal was to identify the largest pattern of pharmacophore features. Holliday and Willett (1997) pointed out that this goal reflects the assumption that a single pharmacophore is responsible for the shared activity being studied and that the approach may result in making a large number of “matches” in querying a database. They adopted an alternative strategy, describing the Mapping Pharmacophores In Ligands (MPHIL) program, which searches for a pattern of K point-like features and associated inter-feature distances for which a given set of N ($5 \leq N \leq 20$) molecules share at least m features in common, where m is given and K should be as small as possible. If $K = m$, their approach reduces to the traditional method of pharmacophore matching such as that implemented by DISCO. Holliday and Willett devised a two-phased approach, using one genetic algorithm (GA) to identify sets of m points common to many of the input molecules and a second GA that assembles the m -point subsets using a clique-detection procedure to yield the K -point site that is as compact as possible. They described their GAs in detail and related their computational tests.

More recently, Rhodes et al. (2003) proposed the Candidate Ligand Identification Program (CLIP) for 3D similarity searching. They searched for similarity between two sets of pharmacophore points by using the Bron-Kerbosch (1973) algorithm to identify the maximum common sub-structure (MCS) in 3D, considering only inter-atomic distances that satisfied a tolerance defining acceptable matches. In addition, they devised two (“corrected”) similarity coefficients to “normalize” the number of points in each MCS, providing the decisive measure of comparison. They compared their approach with three others and demonstrated its efficacy in an HIV assay.

The maximum independent set problem on a graph, which is equivalent to the maximum clique problem on the complement of the graph, has also been used in protein structure alignment. First, a contact map, a graph that represents the 3D fold of a protein, must be constructed. In such a graph, each vertex represents a residue; and each edge, a contact (a Euclidean distance less than some pre-determined threshold) between the two residues (vertices)

when the protein is folded. In the map of the first protein, the number of contacts with end points aligned with residues that share contacts in the map of a second protein gives a measure of the alignment of residues in the two proteins. Alignment must preserve the order of residues so vertices must be numbered in sorted order. The contact map overlap problem, which was introduced by Godzik et al. (1992) and shown to be NP-hard by Goldman et al. (1999), may be reduced to a large-scale maximum independent set problem.

Carr et al. (2000) applied a branch-and-cut approach to the independent set problem, devising a separation algorithm that identifies the most violate clique inequality in polynomial time, even though there are an exponential number of clique inequalities. They applied their approach to the maximum contact map overlap problem to give a measure of protein structure alignments. In these graphs, clique inequalities imply the more general rank inequalities, giving a theoretical explanation of the efficacy of their approach, which they successfully tested on pairs of proteins comprising 50-100 residues/contacts.

6.2 Protein docking

Approaches similar to the ones for matching 3D molecular structures can be used to solve other important problems concerning proteins. Gardiner et al. (2000) extended their previous work described above to the *protein docking* problem. The protein docking problem is to find whether two given proteins interact to form a stable complex; and, if they do, to determine how. Gardiner et al. (2000) state that this problem is fundamental to all aspects of biological function and that development of reliable theoretical protein-docking techniques is an important goal. The approach they used involved representing each of two proteins as a set of potential hydrogen bond donors and acceptors. Then, clique-detection algorithms were used to find maximally complementary sets of donor/acceptor pairs.

Gardiner et al. (2000) observed that the methods they used to find similarities in molecular structures can be employed in matching complementary structures as well. If we consider surfaces of two proteins then complementary surfaces have similar shape (here we do not distinguish between interior and exterior). For example, Figure 7 shows two surfaces. A part of one surface is characterized by white dots and the complementary part of another surface is represented by black dots having a similar pattern.

Hence, in order to find the maximum complementarity between two proteins, one can use a clique-detection approach similar to the one for matching 3D molecular structures. However, such an approach would detect surfaces of similar shape along with complementary surfaces. These areas of simple simi-

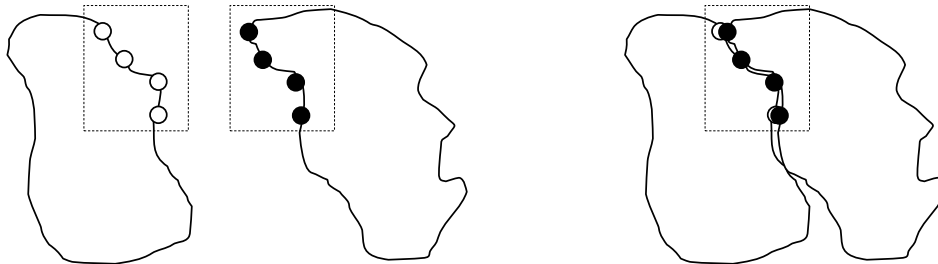


Fig. 7. An illustration to the protein docking problem: Complementary surfaces.

larity should be eliminated since they would correspond to infeasible dockings. Also, it is important that a protein docking model addresses chemical complementarity in addition to geometric shape complementarity. To chemical complementarity, Gardiner et al. (2000) used the hydrogen-bonding nature of solvent-accessible atoms. In proteins, the potential hydrogen bond donors (HBD) are nitrogen atoms and the oxygen atoms of a hydroxyl group, whereas the potential hydrogen bond acceptors (HBA) are given by all oxygen atoms and some nitrogen atoms. To form a hydrogen bond, a HBA needs to be positioned approximately 2 \AA from the hydrogen atom of a HBD so that donor, hydrogen atom and acceptor are approximately collinear. All these considerations are taken into account when a docking graph is constructed.

Next, we briefly describe the docking graph - building procedure due to Gardiner et al. (2000). First, they identified the solvent accessible atoms in both proteins to be docked and “attach” a pseudodonor atom 2 \AA away from each hydrogen of a potential HBD so that it is approximately collinear to the hydrogen atom and the HBD. Then, they recorded the 3D coordinates of all pseudodonor and actual acceptor atoms for each protein. After that, the docking graph was formed with vertices represented by ordered pairs (D, a) and (A, d) , where $A(a)$ and $D(d)$ are labels used for acceptor and donor atoms of the first (second) protein, respectively. An edge in this graph corresponds to a pair of vertices for which the distances between corresponding atoms in the first protein and in the second protein are the same to within some predefined tolerance, say 1.2 \AA . A maximal clique in the constructed graph corresponds to a maximal set of possible hydrogen bonds. However, the vertices in a clique may correspond to atoms sparsely distributed over the surfaces of proteins, while in reality the area of interaction between two proteins is usually not very large. Therefore, Gardiner et al. (2000) imposed another constraint, which requires that the maximum distance between any two atoms of the same protein appearing in the vertices of a clique is less than a given diameter. In their experiments, the authors found that a diameter of 18 \AA was sufficiently reasonable. The number of vertices in docking graphs that the authors used to demonstrate feasibility of their approach ranged between 11,942 and 40,017.

7 Concluding remarks

The aim of this paper is to provide an easy entry for researchers in the field of operations research to join in an interdisciplinary effort to make further advances in the realm of computational biochemistry and genomics. The proposed review does not attempt to exhaust the emerging field of applications of combinatorial optimization techniques to computational biochemistry and genomic analysis, it only demonstrates some examples of the applicability of clique-detection models to a variety of important problems in computational biochemistry and genomics. Moreover, the presented material covers only a fraction of known models in computational biochemistry and genomics that involve detecting cliques or independent sets in graphs. Below we briefly mention some of the related approaches that we are aware of.

An application of the maximum independent set problem appears in a study of human genetic variations that are responsible for some genetic diseases. The most common variation involves only one nucleotide and is called *single nucleotide polymorphism* (SNP). Constructing a complete map of all SNPs occurring in the human genome is currently considered one of the most important goals of genome studies. To construct such a map, one needs to determine the DNA sequences forming all chromosomes. A human chromosome consists of two copies called haplotypes (one is inherited from the father and the other from the mother). The problems related to determining the SNPs haplotypes form an emerging area of computational biology referred to as *haplotyping*. Bonizzoni et al. (2003) recently published a review on such problems. Lancia et al. (2001) considered several versions of the haplotyping problem in terms of their computational complexity. In particular, they showed that one of the versions, the minimum SNPs removal problem, reduces to finding a maximum independent set in a weakly triangulated graph. A graph G is weakly triangulated if neither G nor \bar{G} have a chordless cycle of length > 4 . Weakly triangulated graphs are known to be perfect. Hence, the minimum SNPs removal problem can be solved in polynomial time.

Clique-detection techniques have been applied in several other data-mining problems arising in genome analysis. Due to the computational intractability of the combinatorial problems of interest and massive size of the datasets involved in some of the applications, parallel and grid computing seem to offer one of the most viable options in future computational efforts along these lines. For example, Abu-Khzam et al. (2003) applied a parallel clique-finding algorithm in order to identify putatively co-regulated genes in regulatory networks based on massive microarray datasets. In other related applications, clique-detection algorithms have been used to analyze differential gene expression data (Langston et al. (2004)), to attack the problems of spot matching for two-dimensional gel electrophoresis images, protein structure alignment and

protein side-chain packing Bahadur et al. (2002), and to discover motifs (Baldwin et al. (2004); Pevzner and Sze (2000); Kato and Takahashi (1997)).

To conclude, computational biochemistry and genomics offer endless opportunities for fruitful collaborations between biologists, mathematicians, computer scientists, operations researchers, and representatives of other disciplines. Apart from intractability of combinatorial problems, one of the major challenges that operations researchers face when dealing with problems arising in biology, likewise in other applications, is providing a reasonable model to represent the problem of interest. We hope that the proposed survey will help the reader to make a step in this direction.

References

- Abbas, A., Holmes, S., 2004. Bioinformatics and management science: Some common tools and techniques. *Operations Research* 52, 165–190.
- Abello, J., Pardalos, P. M., Resende, M. G. C., 1999. On maximum clique problems in very large graphs. In: Abello, J., Vitter, J. S. (Eds.), *External Memory Algorithms*. Vol. 50 of DIMACS Series in Discrete Mathematics and Theoretical Computer Science. American Mathematical Society, Providence, RI, pp. 119–130.
- Abu-Khzam, F., Langston, M., Shanbhag, P., November 2003. Scalable parallel algorithms for difficult combinatorial problems: a case study in optimization. In: *International Conference on Parallel and Distributed Computing and Systems*. Los Angeles, CA, pp. 563–568.
- Arora, S., Safra, S., 1992. Approximating clique is NP-complete. In: *Proceedings of the 33rd IEEE Symposium on Foundations of Computer Science*. Piscataway, NJ, pp. 2–13.
- Arratia, R., Lander, E., Tavaré, S., Waterman, M., 1991. Genomic mapping by anchoring random clones: A mathematical analysis. *Genomics* 11, 806–827.
- Avondo-Bodeno, G., 1962. *Economic Applications of the Theory of Graphs*. Gordon and Breach Science Publishers, New York.
- Babel, L., 1991. Finding maximum cliques in arbitrary and special graphs. *Computing* 46, 321.
- Bafna, V., Narayanan, B., Ravi, R., 1996. Nonoverlapping local alignments (weighted independent sets of axis-parallel rectangles). *Discrete Applied Mathematics* 71, 41–53.
- Bahadur, D., Akutsu, T., Tomita, E., Seki, T., Fujiyama, A., 2002. Point matching under non-uniform distortions and protein side chain packing based on an efficient maximum clique algorithm. *Genome Informatics* 13, 143–152.
- Balas, E., Yu, C., 1986. Finding a maximum clique in an arbitrary graph. *SIAM Journal of Computing* 15, 1054–1068.

- Baldwin, N. E., Collins, R. L., Langston, M. A., Leuze, M. R., Symons, C. T., Voy, B. H., April 2004. High performance computational tools for motif discovery. In: IEEE International Workshop on High Performance Computational Biology.
- Berman, P., 2000. A $d/2$ -approximation for maximum weight independent set in d -claw free graphs. In: Proceedings of the 7-th Scandinavian Workshop on Algorithmic Theory. Vol. 1851 of Lecture Notes in Computer Science. Springer-Verlag, pp. 214–219.
- Blazewicz, J., Formanowicz, P., Kasprzak, M., 2004. Selected combinatorial problems of computational biology. European Journal of Operational ResearchForthcoming.
- Boginski, V., Butenko, S., Pardalos, P. M., 2003. On structural properties of the market graph. In: Nagurney, A. (Ed.), Innovation in Financial and Economic Networks. Edward Elgar Publishers, London, pp. 29–45.
- Bomze, I. M., Budinich, M., Pardalos, P. M., Pelillo, M., 1999. The maximum clique problem. In: Du, D.-Z., Pardalos, P. M. (Eds.), Handbook of Combinatorial Optimization. Kluwer Academic Publishers, Dordrecht, The Netherlands, pp. 1–74.
- Bonizzoni, P., Vedova, G. D., Dondi, R., Li, J., 2003. The haplotyping problem: An overview of computational models and solutions. Journal of Computer Science and Technology 18, 675–688.
- Brint, A., Willett, P., 1987. Algorithms for the identification of three-dimensional maximal common substructures. J. Chem. Inf. Comput. Sci. 27, 152–158.
- Bron, C., Kerbosch, J., 1973. Algorithm 457: Finding all cliques on an undirected graph. Communications of ACM 16, 575–577.
- Butenko, S., Pardalos, P. M., Sergienko, I. V., Shylo, V., Stetsyuk, P., 2004. Estimating the size of correcting codes using extremal graph problems. In: Pearce, C. (Ed.), Optimization: Structure and Applications. Kluwer Academic Publishers, Dordrecht, The Netherlands, in press.
- Carr, R., Lancia, G., Istrail, S., 2000. Branch-and-cut algorithms for independent set problems: Integrality gap and an application to protein structure alignment. Tech. Rep. SAND2000-2171, Sandia National Laboratories, Albuquerque, NM.
- Carraghan, R., Pardalos, P., 1990. An exact algorithm for the maximum clique problem. Operations Research Letters 9, 375–382.
- Chandru, V., Sharma, A. D., Kumar, V. A., 2003. The algorithms of folding proteins on lattices. Discrete Applied Mathematics 127, 145–161.
- Clote, P., Backofen, R., 2000. Computational Molecular Biology. Wiley.
- de Weese-Scott, C., Moulton, J., 2004. Molecular modeling of protein function regions. Proteins: Structure, Function, and Bioinformatics 55, 942–961.
- Frank-Kamenetskii, M., 1997. Unraveling DNA the Most Important Molecule of Life. Addison-Wesley, Reading, MA.
- Gardiner, E., Artymiuk, P., Willett, P., 1997. Clique-detection algorithms for matching three-dimensional molecular structures. Journal of Molecular

- Graphics and Modelling 15, 245–253.
- Gardiner, E., Willett, P., Artymiuk, P., 2000. Graph-theoretic techniques for macromolecular docking. *J. Chem. Inf. Comput. Sci.* 40, 273–279.
- Garey, M., Johnson, D., 1979. *Computers and Intractability: A Guide to the Theory of NP-completeness*. W.H. Freeman and Company, New York.
- Gendreau, M., Picard, J., Zubieta, L., 1988. An efficient implicit enumeration algorithm for the maximum clique problem. *Lecture Notes Econ. Mathematics Systems* 304, 70.
- Ghose, A. K., Crippen, G. M., 1985. Geometrically feasible binding modes of a flexible ligand molecule at the receptor-site. *Journal of Comput. Chem.* 6, 350–359.
- Godzik, A., Sklonick, J., Kolinski, A., 1992. A topology fingerprint approach to inverse protein folding problem. *Journal of Molecular Biology* 227, 227–238.
- Goldman, D., Istrail, S., Papadimitriou, C., 1999. Algorithmic aspects of protein structure similarity. In: *Proceedings of the 40th IEEE Symposium on Foundations of Computer Science*. pp. 512–522.
- Golender, V., Rozenblit, A., 1983. *Logical and Combinatorial Algorithms for Drug Design*. Research Studies Press, Letchworth.
- Greenberg, H. J., Hart, W. E., Lancia, G., 2004. Opportunities for combinatorial optimization in computational biology. *INFORMS Journal on Computing* 16, 211–231.
- Grindley, H., Artymiuk, P., Rice, D., Willett, P., 1993. Identification of tertiary structure resemblance in proteins using a maximal common sub-graph isomorphism algorithm. *Journal Mol. Biol.* 229, 707–721.
- Harley, E., Bonner, A., Goodman, N., 1999. Revealing hidden interval graph structure in STS-content data. *Bioinformatics* 15, 278–285.
- Harley, E., Bonner, A., Goodman, N., 2001. Uniform integration of genome mapping data using intersection graphs. *Bioinformatics* 17, 487–494.
- Håstad, J., 1999. Clique is hard to approximate within $n^{1-\epsilon}$. *Acta Mathematica* 182, 105–142.
- Heun, V., 2003. Approximate protein folding in the HP side chain model on extended cubic lattices. *Discrete Applied Mathematics* 127, 163–177.
- Holliday, J., Willett, P., 1997. Using a genetic algorithm to identify common structural features in sets of ligands. *Journal of Molecular Graphics and Modelling* 15, 221–232.
- Johnson, D. S., Trick, M. A. (Eds.), 1996. *Cliques, Coloring, and Satisfiability: Second DIMACS Implementation Challenge*. Vol. 26 of DIMACS Series. American Mathematical Society, Providence, RI.
- Kato, H., Takahashi, Y., 1997. SS3D-P2: a three dimensional substructure search program for protein motifs based on secondary structure elements. *Computer Applications in the Biosciences* 13, 593–600.
- Koike, R., Kinoshita, K., Kidera, A., 2004. Probabilistic description of protein alignments for sequences and structures. *Proteins: Structure, Function, and Bioinformatics* 56, 157–166.
- Kuhl, F., Crippen, G., Friesen, D., 1984. A combinatorial algorithm for cal-

- culating ligand binding. *Journal Comput. Chem.* 5, 24–34.
- Lancia, G., Bafna, V., Istrail, S., Lippert, R., Schwartz, R., 2001. Snps problems, complexity, and algorithms. In: *ESA 2001*. pp. 182–193.
- Langston, M., Lin, L., Peng, X., Baldwin, N., Symons, C., Zhang, B., Snoddy, J., 2004. A combinatorial approach to the analysis of differential gene expression data. Tech. Rep. 04-514, Department of Computer Science, University of Tennessee, Knoxville, TN.
- Martin, Y., Bures, M., Danaher, E., DeLazzer, J., Lico, I., P.A.Pavlik, 1993. A fast new approach to pharmacophore mapping and its application to dopaminergic and benzodiazepine agonists. *Journal of Comput.-Aided Mol. Design* 7, 83.
- McMorris, F., Wang, C., Zhang, P., 1998. On probe interval graphs. *Discrete Applied Mathematics* 88, 315–324.
- Moon, J. W., Moser, L., 1965. On cliques in graphs. *Israel Journal of Mathematics* 3, 23–28.
- Needleman, S. B., Wunsch, C. D., 1970. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology* 48, 443–453.
- Pevzner, Sze, S.-H., 2000. Combinatorial approaches to finding subtle signals in dna sequences. In: *Proc. 8th International Conf. on Intelligent Systems for Molecular Biology*. pp. 269–278.
- Pevzner, P. A., 2003. *Computational Molecular Biology: An Algorithmic Approach (Computational Molecular Biology)*. Bradford Book.
- Rhodes, N., Willett, P., Calvert, A., Dunbar, J., Humblet, C., 2003. CLIP: Similarity searching of 3D databases using clique detection. *Journal of Chem. Info. Computer Science* 43, 443–338.
- Samudrala, R., Moult, J., 1988. A graph-theoretic algorithm for comparative modeling of protein structure. *J. Mol. Biol.* 209, 287–302.
- Samudrala, R., Moult, J., 1997. Handling context-sensitivity in protein structures using graph theory: Bona fide prediction. *Proteins: Structure, Function, and Genetics Suppl.* 1, 43–49.
- Scharf, M., Schneider, R., Casari, G., Bork, P., Valencia, A., Ouzounis, C., Sander, C., 1994. GeneQuiz: A workbench for sequence analysis. In: *Altmann, R., Brutlag, D., Karp, P., Lathrop, R., Searls, D. (Eds.), Proceedings of the Second International Conference on Intelligent Systems in Molecular Biology*. AAAIPress, Menlo Park, CA, pp. 348–353.
- Schneider, R., Sander, C., 1996. The HSSP database of protein structure-sequence alignments. *Nucl. Acids Res.* 24, 201–205.
- Shindo, M., Tomita, E., 1990. Simple algorithm for finding a maximum clique and its worst-case time complexity. *Systems Comput.* 21, 1.
- Smith, T. F., Waterman, M. S., 1981. Identification of common molecular subsequences. *Journal of Molecular Biology* 147, 195–197.
- Takahashi, Y., Maeda, S., Sasaki, S.-I., 1987. Automated recognition of common geometrical patterns among a variety of three-dimensional molecular structures. *Anal. Chim. Acta.* 200, 363–377.

Watson, J., Crick, F., 1953. Genetic implications of the structure of deoxyribonucleic acid. *Nature* 171, 964–967.