Computational Challenges with Cliques, Quasi-cliques and Clique Partitions in Graphs*

Panos M. Pardalos and Steffen Rebennack

Center for Applied Optimization
Department of Industrial & Systems Engineering
University of Florida, USA
{pardalos,steffen}@ufl.edu
http://www.ise.ufl.edu/pardalos/

Abstract. During the last decade, many problems in social, biological, and financial networks require finding cliques, or quasi-cliques. Cliques or clique partitions have also been used as clustering or classification tools in data sets represented by networks. These networks can be very large and often massive and therefore external (or semi-external) memory algorithms are needed. We discuss four applications where we identify computational challenges which are both of practical and theoretical interest.

1 Introduction

An undirected graph G is a pair (V, E) consisting of a nonempty, finite node set V, $|V| < \infty$, and a finite (possibly empty) edge set $E \subseteq V \times V$ of unordered pairs of distinct elements of V. Graphs without loops and multiple edges are so-called simple graphs. In a multigraph, we allow a graph to have multiple edges but loops are not allowed. As we mainly consider undirected simple graphs in this article we shall call them from now on just graphs and mention it otherwise explicitly. Two nodes $u, v \in V$ of graph $\mathrm{G} = (V, E)$ are adjacent if (u, v) is an edge of G; *i.e.*, $(u, v) \in E$. A graph is said to be complete if there is an edge between any to nodes. The complement $\overline{\mathrm{G}} := (V, \overline{E})$ of a graph $\mathrm{G} = (V, E)$ is the graph with the same node set as G and the complement edge set $\overline{E}$, containing only the edges which are not in E; *i.e.*, $\overline{E} := \{(i, j) \,|\, i, j \in V,\, i \neq j \wedge (i, j) \notin E\}$. A graph is connected if there is a path between any two nodes of the graph, otherwise it is disconnected. The connected components of a graph G are the connected nonempty inclusion-maximal subgraphs of G. The length of the longest path among all shortest paths between any two nodes in the graph is called the diameter. The density of a graph is defined as the ration $\frac{2|E|}{|V|^2-|V|}$. Two graphs $\mathrm{G}_1 = (V_1, E_1)$ and $\mathrm{G}_2 = (V_2, E_2)$ are called isomorphic, if there is a bijection $\phi\colon V_1 \to V_2$ such that $(u, v) \in E_1 \Leftrightarrow (\phi(u), \phi(v)) \in E_2$. A common subgraph of two graphs G_1 and G_2 consists of subgraphs $\overline{\mathrm{G}}_1$ and $\overline{\mathrm{G}}_2$ of G_1 and G_2, respectively, such that $\overline{\mathrm{G}}_1$ is isomorphic to $\overline{\mathrm{G}}_2$. For a given node set $S \subseteq V$, $\mathrm{G}(S) := (V, E \cap S \times S)$ is

* This research is partially supported by DTRA and Air Force grants.

P. Festa (Ed.): SEA 2010, LNCS 6049, pp. 13–22, 2010.

the subgraph induced by S. This shall be enough on general graph theoretical definitions for this article; more details can be found, for instance, in [1,2].

A set of nodes S is called a clique if the subgraph $\mathrm{G}(S)$ is complete. We distinguish between a maxim*al* clique which is not a proper subset of any other clique in G and a maxim*um* clique which is a clique of maximum cardinality; *i.e.*, the largest clique in graph G. A set of nodes S in a graph G is a stable set if any two nodes in S are not adjacent. A stable set is sometimes also called independent set, vertex packing, co-clique or anticlique. The definition of cliques can be generalized by the concept of quasi-cliques. A quasi-clique, or γ-clique, C_γ of graph $\mathrm{G} = (V, E)$ is a subset of V such that the induced subgraph $\mathrm{G}(C_\gamma)$ is connected and has at least

$$\left\lfloor \gamma \frac{q(q-1)}{2} \right\rfloor \tag{1}$$

edges; where $q := |C_\gamma|$ and $\gamma \in [0, 1]$. In the extreme case of $\gamma = 0$, $\mathrm{G}(C_\gamma)$ may have no edges and if $\gamma = 1$, then C_γ is a clique in G. A coloring of G is a partition of V into disjoint stable sets, while a clique covering is a partition into disjoint cliques. In the following, we call a clique covering a clique partition.

The maximum clique problem is to find a maximum clique in a given graph G. We denote the cardinality of a maximum clique in graph G by $\omega(\mathrm{G})$ which is also called the clique number. Analogously, the maximum stable set problem asks to find a stable set of maximum cardinality. The cardinality of such a stable set is denoted by $\alpha(\mathrm{G})$ and is called the stability number or stable set number. The coloring number or chromatic number, which is denoted by $\chi(\mathrm{G})$, is the smallest number of stable sets needed for a coloring of G. Similarly, the smallest number of cliques for a clique partition of G is called clique covering number and is abbreviated with $\overline{\chi}(\mathrm{G})$. In this article, we are interested in finding a γ-clique of maximum size for fixed density γ; there are several other optimization problems for quasi-cliques, such as, maximize γq, or fix q and maximize γ.

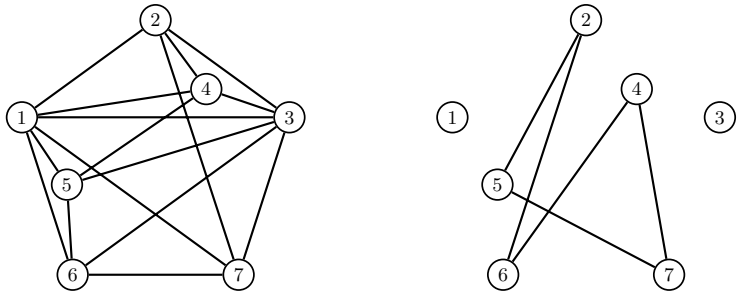
As a clique in G corresponds to a stable set in the complement graph $\overline{\mathrm{G}}$, we obtain the relation

$$\alpha(\mathrm{G}) = \omega(\overline{\mathrm{G}}). \tag{2}$$

Furthermore, the following relations hold true

$$\chi(\mathrm{G}) = \overline{\chi}(\overline{\mathrm{G}}), \tag{3}$$
$$\omega(\mathrm{G}) \leq \chi(\mathrm{G}), \tag{4}$$
$$\alpha(\mathrm{G}) \leq \overline{\chi}(\mathrm{G}). \tag{5}$$

Since the number of stable sets needed to cover a graph is equal to the number needed to cover the complement with cliques, equality (3) is true. Hence, to find the coloring number or the clique covering number are algorithmically equivalent problems and we may discuss either of them depending on the application. To partition the node set of a graph G into disjoint stable sets, one needs at least



(a) Graph G (b) Complement graph $\overline{\mathrm{G}}$

Fig. 1. A graph G and its complement $\overline{\mathrm{G}}$ with $\omega(\mathrm{G}) = \alpha(\overline{\mathrm{G}}) = 4$ and $\chi(\mathrm{G}) = \overline{\chi}(\overline{\mathrm{G}}) = 5$

Table 1. Maximum quasi-cliques for graph G corresponding to Figure 1 for different values of γ

γ	Maximum Cardinality	quasi-clique
1	4	{1,2,3,4}
[13/15, 1)	5	{1,2,3,4,5}
[17/21, 13/15)	6	{1,2,3,4,5,6}
[0, 17/21)	7	{1,2,3,4,5,6,7}

the size of a maximum clique in G. This is stated by inequality (4). Inequality (5) is the consequence of (2), (3), (4) and the observation that the complement of $\overline{\mathrm{G}}$ is again G.

Let us now have a closer look at the graph G and its complement $\overline{\mathrm{G}}$ in Figure 1. Node sets $\{1, 2, 3, 4\}$, $\{1, 2, 3, 7\}$, $\{1, 3, 4, 5\}$, $\{1, 3, 5, 6\}$ and $\{1, 3, 6, 7\}$ define all the maximum cliques in G (or maximum stable sets in $\overline{\mathrm{G}}$) leading to a clique number of 4 for G. The clique covering number for $\overline{\mathrm{G}}$ is 5 due to the nodes 2, 5, 7, 4 and 6 which form a so-called odd hole. This yields to the coloring number of 5 for the original graph G, showing that the coloring number and the clique number are not in general equal, confirming relation (4). A smaller example is given when the graph itself is an odd hole with 5 nodes. In this case, a maximum clique has size 2, but the coloring number is 3. Table 1 provides maximum quasi-cliques for different values of γ. For instance, if $\gamma < 17/21$, then the whole node set V defines a maximum quasi-clique C_γ. Increasing the γ value steadily reduces the maximum cardinality of a quasi-clique.

The maximum clique problem and the clique covering problem are one of Karp's original 21 problems shown to be NP-complete [3,4]; *i.e.*, unless P = NP, exact algorithms are guaranteed to return a solution only in a time which increases exponentially with the number of nodes in the graph. Furthermore, Arora and Safra [5] proved that for some $\varepsilon > 0$ the approximation of the clique

number within a factor of $|V|^{\varepsilon}$ is NP-hard. A similar result was proven by Lund and Yanakakis [6] for the chromatic number. Computing a maximum quasi-clique for an arbitrary γ is also NP-complete, as for $\gamma = 1$ the problem is equivalent to the maximum clique problem. There is a large variety of exact and heuristic algorithms available for the maximum clique problem [7,8,9] and the clique partition problem [10]. Some recent work on quasi-clique algorithms can be found in [11].

In this article, we focus on computational challenges related to cliques, quasi-cliques as well as clique partitions arising from four applications: Call graphs (Section 2), coding theory (Section 3), matching molecular structures (Section 4), and Keller's conjecture (Section 5).

2 Call Graphs

Phone companies are faced with enormous data sets; *e.g.*, resulting from long distance phone-calls. In 1999, AT&T had approximately 300 million phone calls per day leading to a yearly storage space of 20 terabytes [12]. However, the analysis of such data is of great importance for the companies to study customer patterns and to be able to optimize their operations.

Given the data for phone calls over a specific time period (*e.g.*, ranging from days to months), one can construct a so-called call graph as follows. Each mobile user represents one node of the graph and there is a directed edge for each phone call. Hence, the resulting graph is a directed multigraph as one user may call the same user multiple times. Of interest in these graphs are especially undirected quasi-cliques as they provide information about highly interconnected users [13,14].

Graphs having millions of nodes are often refereed to as massive graphs. Even the visualization of such graphs on a screen or basic statistical analysis are challenging tasks [15]. As the graphs are very large, they typically do not fit into the RAM of the computer or even into the main memory – hence, so-called external memory algorithms have been developed.

The call graphs tend to have specific properties. The most important ones are [16,17]

- the graphs are very large; *i.e.*, they have millions of nodes;
- the graphs have a very low density; *i.e.*, in the order of 0.0000001;
- the graphs are often disconnected, though connected components can be very large; *i.e.*, they may have millions of nodes;
- the undirected diameter of the graphs is low; *i.e.*, in the order of $\log(n)$;
- the node indegree d_{in} and the node outdegree d_{out} distributions follow a power-law; *i.e.*, $P(d_{in}) \sim d_{in}^{-\gamma_{in}}$ with $\gamma_{in} \in [2,3]$ and $P(d_{out}) \sim d_{out}^{-\gamma_{out}}$ with $\gamma_{out} < 2$ where $P(d)$ equals the number of nodes having degree d divided by the total number of nodes in the graph.

Abello et al. [16] studied a call graph corresponding to 1-day landline phone calls at AT&T and derived a call graph with 53 million nodes and 170 million edges.

To exploit the special structure mentioned above, the authors made extensive use of preprocessing. The algorithmic analysis of such graphs is practically very important. However, the known algorithms do not scale well on such graphs. This leads us to our first challenge.

Challenge 1 (Algorithm for Massive Graphs with Very Low Density) *Design an efficient algorithm together with a data base for the maximum γ-clique problem tailored to massive graphs characterized by very low density and by the node degree distribution following a power-law. Real world call graphs serve as an excellent test bed.*

Graphs with similar properties as the call graphs are the internet graphs, mobile graphs, graphs from social networks, SMS graphs, or www graphs [17,18,19,20]. Sometimes scale-free properties can be observed in these graphs due to self-organizing processes making them so-called small-world networks [21]. Biological networks lead to similar challenging problems with graphs [22].

3 Graphs in Coding Theory

A fundamental problem of interest is to send a message across a noisy channel with a maximum possible reliability. In coding theory, one wishes to find a binary code as large as possible that can correct a certain number of errors for a given size of the binary words.

For a binary vector $u \in \{0,1\}^n$, representing the words, denote by $F_e(u)$ the set of all vectors which can be obtained from u resulting from a certain error e, such as deletion or transposition of bits. Note that the elements in $F_e(u)$ do not necessarily have to have length n; *e.g.*, due to the deletion of digits. A subset $C \subseteq \{0,1\}^n$ is said to be an e-correcting code if $F_e(u) \cap F_e(v) = \emptyset$ for all $u, v \in C$ with $u \neq v$. The problem of interest is to find the largest correcting codes; *i.e.*, a set C of maximum size. For this to be meaningful, one has to have an idea about the nature of the error e. One may distinguish single deletion errors, two-deletion errors, single transposition including or excluding the end-around transposition errors, or one error on the Z-channel [23].

Consider a graph G having a node for every vector $u \in \{0,1\}^n$ and having an edge (u, v), if $F_e(u) \cap F_e(v) \neq \emptyset$. This way, an edge represents a conflict for an e-correcting code. Such graphs are called conflict graphs. Due to the construction of the graph, a correcting code corresponds to a stable set in G. Therefore, a largest e-correcting code can be found by solving the maximum independent set problem in the considered graph G.

Good lower bounds on the code size are especially interesting for asymmetric codes, such as codes correcting one error on the Z-channel (non zero components of any vector may change from 1 to 0). For that, several partition methods have been proposed in the literature, using minimum stable set partitions on conflict graphs. For the details of these methods, we refer the interested reader to [24,23]. The challenge for finding good lower bounds for code sizes is ongoing and tailored stable set partition algorithms are needed.

Challenge 2 (Algorithm for Conflict Graphs in Coding Theory). *Design an efficient algorithm for the minimum stable set partition problem tailored to conflict graphs resulting from applications in coding theory.*

Another example where minimum clique partitions are used as lower bounds are mandatory coverage problems, where a set of demand points has to be covered by a set of potential sites. Examples are ambulance location problems [25] or tiling problems [26], which have both to be solved in real-time, *i.e.*, within 2 minutes. For the latter problem, real-world instances resulting from cytology applications can have the size of the magnitude of tens of thousands of nodes.

4 Matching Molecular Structures

In the pharmaceutical and agrochemical industry, the problem of establishing structural relationships between pairs of three-dimensional molecular structures is an important problem. These three-dimensional molecular structures can be represented using graphs. For a protein, for instance, the nodes of the graph are given by the α-helix and β-strand secondary structure elements, whereas the edges are defined through inter-secondary structure element angles and distances [27]. In addition, both the nodes and the edges have labels, corresponding to the atomic types and the interatomic distances, respectively [28].

Consider a pair of node and edge labeled graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$. Then, one can construct the correspondence graph C as follows. Whenever two nodes $v_1 \in V_1$ and $v_2 \in V_2$ have the same label, there is a node in graph C. Hence, the nodes in C are pairs v_1v_2 and two nodes v_1v_2 and $v_1'v_2'$ are connected in C if the labels of the edges $(v_1, v_1') \in E_1$ and $(v_2, v_2') \in E_2$ are the same.

For a pair of three-dimensional chemical molecules, a maximum common subgraph in their corresponding graphs relates to the largest set of atoms that have matching distances between atoms. Hence, a maximum common subgraph is an obvious measure of structural similarity and gives important information about the two molecules.

Due to the construction of the correspondence graph C for two graphs corresponding to a pair of such molecules, it is almost obvious that the maximum common subgraphs in G_1 and G_2 correspond to cliques in their correspondence graph C. Therefore, one can find the maximum common subgraph of two arbitrary graphs by finding a maximum clique in their correspondence graph.

These correspondence graphs are characterized by their low edge densities – typically between 0.005 and 0.2 [29]. As suggested in the review by Raymond and Willett [30], the maximum clique algorithms currently available are computationally too expensive for these applications. This leads us to the following challenge.

Challenge 3 (Algorithm for Correspondence Graphs with Low Density). *Design an efficient algorithm for the maximum clique problem tailored to correspondence graphs resulting from matching of three-dimensional chemical molecules.*

Similar challenges occur when computing maximum cliques for the protein docking problem [31], where one wants to find out whether two proteins form a stable complex or not, or for comparing protein structures [32].

5 Keller's Conjecture

Keller's conjecture [33] goes back to Minkowski's conjecture [34] which stated that in a lattice tiling of $\mathbb{R}^n$ by translates of a unit hypercube, there exists two cubes that share $(n-1)$ dimensional face [35]. Minkowski's theorem was proven by Hajós [36] in 1950. Keller suggested that Minkowski's theorem can be generalized as the lattice assumption might not be necessary. Indeed, Perron [37] showed that this is true for $n \leq 6$. However, for $n \geq 8$ the lattice assumption is necessary, which was shown by Lagarias and Shor [38] and Mackey [39]. The Keller conjecture remains open for $n = 7$.

After 80 years, the rally towards the Keller's conjecture has not been ended. Almost as a by-product, a very interesting class of graphs for the maximum clique problem has been derived. For any given $n \in \mathbb{N}^+$, Corrádi and Szabó [40] constructed the so-called Keller graph Γ_n. The nodes of Γ_n are vectors of length n with values of $0, 1, 2$ or 3. Any two vectors are adjacent, if and only if in some of the n coordinates, they differ by precisely two (in absolute value). The Keller graph Γ_n is a dense graphs where the clique size is bounded by 2^n, [41]. Corrádi and Szabó [40] proved that there is an counterexample to Keller's conjecture, if and only if Γ_n has a clique of size 2^n.

With the help of these results, Lagarias and Shor [38] used a block substitution method to construct an appropriate clique, providing a counterexample for 10 dimensions. In a similar way, Mackey [39] constructed such a clique for dimension 8, proving that the Keller's conjecture does not hold true for $n \geq 8$. However, as the case of $n = 7$ remains open, we get the following challenge.

Challenge 4 (Open problem [40]). *For the Keller graph Γ_7, either find a maximum (cardinality) clique of size* 128 *or prove that none such clique exists.*

Hasselberg et al. [41] contributed several test case generators to the DIMACS challenge on cliques, among them are the Keller graphs. It turned out that the Keller graphs lead to challenging maximum clique problems and, to our best knowledge, no computational algorithms could solve the problem for $n \geq 6$. Especially as the maximum clique size for the Keller graphs are known (except for the cases of $n = 6, 7$), the Keller graphs are very useful graphs when benchmarking maximum clique algorithms with large graphs having high density. This leads us to the next challenge.

Challenge 5 (Algorithm for Dense Graphs). *Design an efficient algorithm for the maximum clique problem which is tailored to dense graphs. The Keller graphs should be used to benchmark its performance while the goal should be that the algorithm computes optimal solutions for the graphs with* $2 \leq n \leq 8$.

Table 2 summarizes the Keller graphs and their clique numbers. For the case of $n \geq 6$, we also know a lower bound on the clique number, given by $\omega(\Gamma_n) \geq \frac{57}{64}2^n$.

Table 2. Keller graphs and their clique numbers

n	# nodes	# edges	$\omega(\Gamma_n)$	reference
2	16	40	2	[40]
3	64	1,088	5	[40]
4	256	21,888	12	[40]
5	1,024	397,312	28	[40]
6	4,096	6,883,328	–	
7	16,384	116,244,480	–	
≥ 8	4^n	$\frac{1}{2}4^n(4^n - 3^n - n)$	2^n	[39,38,41]

– **open problem**

6 Conclusions

We have seen four applications leading to large graphs for clique, quasi-clique and clique partition problems. However, these graphs have different structural properties. Most significantly, the size of the graphs and their density vary greatly. Tailored algorithms to each of these problems are required to be able to solve these problems efficiently.

References

1. Diestel, R.: Graph Theory. Electronic Edition 2000. Springer, New York (2000)
2. West, D.B.: Introduction to Graph Theory, 2nd edn. Prentice-Hall, Englewood Cliffs (2000)
3. Karp, R.: Reducibility Among Combinatorial Problems. In: Miller, R.E., Thatcher, J. (eds.) Proceedings of a Symposium on the Complexity of Computer Computations, pp. 85–103. Plenum Press, New York (1972)
4. Garey, M.R., Johnson, D.S.: Computers and Intractability, A guide to the Theory of NP-Completeness. In: Klee, V. (ed.) A series of books in the mathematical sciences. W. H. Freeman and Company, New York (1979)
5. Arora, S., Safra, S.: Probabilistic Checking of Proofs; a new Characterization of NP. In: Proceedings 33rd IEEE Symposium on Foundations of Computer Science, pp. 2–13. IEEE Computer Society, Los Angeles (1992)
6. Lund, C., Yannakakis, M.: On the hardness of approximating minimization problmes. JACM 41, 960–981 (1994)
7. Bomze, I., Budinich, M., Pardalos, P., Pelillo, M.: The maximum clique problem. In: Du, D.Z., Pardalos, P. (eds.) Handbook of Combinatorial Optimization, pp. 1–74. Kluwer Academic Publishers, Dordrecht (1999)
8. Rebennack, S.: Stable Set Problem: Branch & Cut Algorithms. In: Floudas, C.A., Pardalos, P.M. (eds.) Encyclopedia of Optimization, 2nd edn., pp. 3676–3688. Springer, Heidelberg (2008)
9. Rebennack, S., Oswald, M., Theis, D., Seitz, H., Reinelt, G., Pardalos, P.: A Branch and Cut solver for the maximum stable set problem. Journal of Combinatorial Optimization, doi:10.1007/s10878-009-9264-3

10. Pardalos, P., Mavridou, T., Xue, J.: The graph coloring problem: a bibliographic survey. In: Du, D.Z., Pardalos, P. (eds.) Handbook of Combinatorial Optimization, vol. 2, pp. 331–395. Kluwer Academic Publishers, Dordrecht (1990)
11. Brunato, M., Hoos, H., Battiti, R.: On Effectively Finding Maximal Quasi-cliques in Graphs. In: Maniezzo, V., Battiti, R., Watson, J.-P. (eds.) LION 2007 II. LNCS, vol. 5313, pp. 41–55. Springer, Heidelberg (2008)
12. Hayes, B.: Graph Theory in Practice: Part I. American Scientist 88(1), 9 (2000)
13. Cipra, B.: Massive graphs pose big problems. Technical report, SIAM NEWS, April 22 (1999)
14. Abello, J., Resende, M., Sudarsky, S.: Massive Quasi-Clique Detection. In: Rajsbaum, S. (ed.) LATIN 2002. LNCS, vol. 2286, p. 598. Springer, Heidelberg (2002)
15. Ye, Q., Wu, B., Suo, L., Zhu, T., Han, C., Wang, B.: TeleComVis: Exploring Temporal Communities in Telecom Networks. In: Buntine, W., Grobelnik, M., Mladenić, D., Shawe-Taylor, J. (eds.) ECML PKDD 2009. LNCS, vol. 5782, pp. 755–758. Springer, Heidelberg (2009)
16. Abello, J., Pardalos, P., Resende, M.: On Maximum Clique Problems in Very Lagre Graphs. In: External Memory Algorithms. DIMACS Series, pp. 119–130. American Mathematical Society, Providence (1999)
17. Nanavati, A., Singh, R., Chakraborty, D., Dasgupta, K., Mukherjea, S., Das, G., Gurumurthy, S., Joshi, A.: Analyzing the Structure and Evolution of Massive Telecom Graphs. IEEE Transactions on Knowledge and Data Engineering 20(5), 703–718 (2008)
18. Narasimhamurthy, A., Greene, D., Hurley, N., Cunningham, P.: Community Finding in Large Social Networks Through Problem Decomposition. Technical report, UCD School of Computer Science and Informatics (2008)
19. Faloutsos, M., Faloutsos, P., Faloutsos, C.: On Power-law Relationships of the Internet Topology. In: Proceedings of the ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, pp. 251–262 (1999)
20. Hayes, B.: Connecting the Dots: Can the tools of graph theory and social-network studies unravel the next big plot? American Scientist 94(5), 400 (2006)
21. Schintler, L., Gorman, S., Reggiani, A., Patuelli, R., Nijkamp, P.: Small-World Phenomena in Communications Networks: A Cross-Atlantic Comparison. Advances in Spatial Science. In: Methods and Models in Transport and Telecommunications, pp. 201–220. Springer, Heidelberg (2005)
22. Butenko, S., Chaovalitwongse, W., Pardalos, P. (eds.): Clustering Challenges in Biological Networks. World Scientific, Singapore (2009)
23. Butenko, S., Pardalos, P., Sergieko, I., Shylo, V., Stetsyuk, P.: Estimating the size of correcting codes using extremal graph problems. In: Optimization: Structure and Applications. Springer Optimization and Its Applications, vol. 32, pp. 227–243. Springer, Heidelberg (2009)
24. van Pul, C., Etzion, T.: New lower bounds for constatn weight codes. IEEE Trans. Inform. Theory 35, 1324–1329 (1989)
25. Gendreau, M., Laporte, G., Semet, F.: Solving an ambulance location model by tabu search. Location Science 5, 75–88 (1997)
26. Brotcorne, L., Laporte, G., Semet, F.: Fast heuristics for large scale covering ocation problems. Computers and Operations Research 29, 651–665 (2002)
27. Mitchell, E., Artymiuk, P., Rice, D., Willett, P.: Use of techniques derived from graph theory to compare secondary structure motifs in proteins. J. Mol. Biol. 212, 151 (1990)

28. Brint, A., Willett, P.: Algorithms for the Identification of Three-Dimensional Maximal Common Substructures. J. Chem. ZnJ Comput. Sci. 27, 152–158 (1987)
29. Gardiner, E., Artymiuk, P., Willett, P.: Clique-detection algorithms for matching three-dimensional molecular structures. Journal of Molecular Graphics and Modelling 15, 245–253 (1997)
30. Raymond, J., Willett, P.: Maximum common subgraph isomorphism algorithms for the matching of chemical structures. Journal of Computer-Aided Molecular Design 16, 521–533 (2002)
31. Gardiner, E., Willett, P., Artymiuk, P.: Graph-theoretic techniques for macromolecular docking. J. Chem. Inf. Comput. 40, 273–279 (2000)
32. Butenko, S., Wilhelm, W.: Clique-detection models in computational biochemistry and genomics. Euorpean Journal of Operational Research 173, 1–17 (2006)
33. Keller, O.: Über die lückenlose Einfüllung des Raumes mit Würfeln. J. Reine Angew. Math. 163, 231–248 (1930)
34. Minkowski, H.: Diophantische Approximationen. Teubner, Leipzig
35. Stein, S., Szabó, S.: Algebra and Tiling: Homomorphisms in the Service of Geometry. The Carus Mathematical Monographs, vol. 25. The Mathematical Associtaion of America (1994)
36. Hajós, G.: Sur la factorisation des abeliens. Casopis 50, 189–196
37. Perron, O.: Über lückenlose Ausfüllung des n-dimensioanlen Raumes durch kongruente Würfel. Math. Z. 46, 161–180 (1940)
38. Lagarias, J., Shor, P.: Keller's Cube-Tiling Conjecture is False in High Dimensions. Bulletin AMS 27, 279–283 (1992)
39. Mackey, J.: A Cube Tiling of Dimension Eight with No Facesharing. Discrete Comput. Geom. 28, 275–279 (2002)
40. Corrádi, K., Szabó, S.: A Combinatorial Approach for Keller's Conjecture. Periodica Math. Hung. 21(2), 95–100 (1990)
41. Hasselberg, J., Pardalos, P., Vairaktarakis, G.: Test Case Generators and Computational Results for the Maximum Clique Problem. Journal of Global Optimization 3, 463–482 (1993)