OPTIMIZATION PROBLEMS IN UNIT-DISK GRAPHS

B. Balasundaram*and S. Butenko†

Department of Industrial & Systems Engineering
Texas A&M University
College Station, Texas 77843, USA.

1 Introduction

Unit-Disk Graphs (UDGs) are intersection graphs of equal diameter (or unit diameter w.l.o.g.) circles in the Euclidean plane. In the *geometric (or disk) representation*, each circle is specified by the coordinates of its center. Three *equivalent* graph models can be defined with vertices representing the circles [18]. In the *intersection graph model*, two vertices are adjacent if the corresponding circles intersect (tangent circles are also said to intersect). In the *containment graph model*, two vertices are adjacent when one circle contains the center of the other. In the *proximity graph model*, an edge exists between two vertices if the Euclidean distance between the centers of corresponding circles is within a specified bound. Recognizing UDGs is NP-hard [10] and hence no polynomial time algorithm is known for deriving the geometric representation from the graph model. From an algorithmic perspective this places an emphasis on whether or not the geometric representation is needed as input. UDGs are not necessarily perfect or planar [18] as several other geometric intersection graph classes are and thus motivate the need for dedicated theoretical study.

The remainder of this article is organized as follows. We introduce the necessary definitions and notations for the various optimization problems on graphs considered in this article in Section 2. A brief survey of applications modeled using UDGs and the role of optimization problems discussed in this chapter are presented in Section 3. A survey of algorithms and their key ideas for cliques, independent sets, vertex covers, domination, graph coloring and clique paritioning are presented in Section 4. We conclude with a summary in Section 5.

*baski@tamu.edu
†butenko@tamu.edu

2 Definitions

We consider simple, undirected graphs on n vertices and m edges denoted by $G = (V, E)$. For a vertex $v \in V$, $N(v)$ is its neighborhood and $N[v] = N(v) \cup \{v\}$ is its closed neighborhood. We denote the complementary graph by $\bar{G} = (V, \bar{E})$ and the subgraph induced by $S \subseteq V$ by $G[S]$. We denote by $\delta(G)$ and $\Delta(G)$ the minimum and maximum vertex degrees in G respectively. Denote by $d(u, v)$ the shortest distance between $u, v \in V$, then the k-*neighborhood* of v is defined as $N_k(v) = \{u \in V : d(u, v) \leq k\}$. We also use the notation $G-v$ and $G - I$ to refer to the graph obtained from G by deleting vertex v (and incident edges) and by deleting a subset of vertices I (and incident edges), respectively. That is $G - v = G[V \setminus \{v\}]$ and $G - I = G[V \setminus I]$.

A *clique* is a subset of pairwise adjacent vertices in G. The *maximum clique problem* is to find a largest cardinality clique in G and the *clique number* $\omega(G)$ is the size of a maximum clique. An *independent set* (or *stable set*) is a subset of mutually non-adjacent vertices and the maximum independent set problem is to find an independent set of maximum cardinality. The *independence number* (or *stability number*) of a graph G is denoted by $\alpha(G)$ and it is the size of a maximum independent set. A *maximal* clique (independent set) is one that is not a proper subset of another clique (independent set). Cliques and independent sets are complementary to each other in the sense that $C \subseteq V$ is a clique in G if and only if C is an independent set in $\bar{G}$. For arbitrary graphs, the maximum clique and independent set problems are equivalent and possess similar complexity and approximation results. Algorithms and heuristics for one can be adapted via complement for the other. However, this equivalence is not naturally extended to geometric graphs that do not preserve upon complementing, their geometric property. For instance planar independent set is NP-complete [30] while clique is trivial. Results on UDGs will be discussed in Sections 4.1, 4.2. A *vertex cover* S is a subset of vertices such that every edge in G has at least one end point in S. We denote by $\beta(G)$, the size of a minimum vertex cover. Clearly if $I \subseteq V$ is independent, then $V \setminus I$ is a vertex cover of G.

A *proper coloring* of a graph is one in which every vertex is colored (assigned a natural number) such that no two vertices of the same color are adjacent. A graph is said to be *k-colorable* if it admits a proper coloring with k colors. Vertices of the same color are referred to as a *color class* and they induce an independent set. The *chromatic number* of the graph, denoted by $\chi(G)$ is the minimum number of colors required to properly color G. Note that for any graph G, $\omega(G) \leq \chi(G)$, as different colors are required to color the vertices of a clique. The famous theorem by Brooks on graph coloring [11] states that $\chi(G) \leq \Delta(G)$ if G is neither a complete graph nor an odd cycle. A related problem is the *minimum clique partitioning* problem which is to partition the given graph G into a minimum number of cliques, $\bar{\chi}(G)$. Note that this is exactly the graph coloring problem on $\bar{G}$ and $\bar{\chi}(G) = \chi(\bar{G})$.

A *dominating set* D is a subset of vertices such that every vertex in the graph is either in this set or has a neighbor in this set. A *minimal* dominating set contains no proper subset which is also dominating. The minimum cardinality of a dominating set is called the *domination number*, denoted by $\gamma(G)$. Note that every *maximal* independent set is also a *minimal* dominating set. If a dominating set D is independent, it is called an *independent dominating set*. A a dominating set D is called a *connected dominating set* if $G[D]$ is connected. The independent and connected domination numbers (obviously defined) are denoted by $\gamma_i(G)$ and $\gamma_c(G)$. Naturally, G is assumed to be connected when we consider connected domination.

An approximation algorithm with approximation ratio $\rho > 1$ for an optimization problem Π, outputs for every instance x of Π with an optimal value $opt(x)$, a solution of value $sol(x)$ in time polynomial in size of x, such that $sol(x) \leq \rho \times opt(x)$ if Π is a minimization problem or $sol(x) \geq opt(x)/\rho$ if Π is a maximization problem.

An optimization problem Π admits a *fully polynomial time approximation scheme* (FPTAS) if there is an approximation algorithm with approximation ratio $1+\epsilon$ for any $\epsilon > 0$ that runs in time polynomial in size of the input *and* $1/\epsilon$. Π is said to admit a *polynomial time approximation scheme* (PTAS) if it has a polynomial time approximation algorithm with approximation ratio $1+\epsilon$ for each *fixed* $\epsilon > 0$. A problem that is NP-hard in the strong sense [30], does not admit a FPTAS unless P=NP.

3 Applications

A major application area for UDG models is in wireless communication. Here the underlying *connectivity graph* of the wireless nodes with equal and omnidirectional transmission-reception range can be modeled as a UDG [33, 5]. Various optimization problems studied on UDGs are solved to facilitate effective operation of such networks.

For instance, a maximum independent set corresponds to a largest set of wireless nodes that can broadcast simultaneously without interference [56]. Alternately in location logistics, $\alpha(G)$ is also the maximum number of facilities that can be located in n potential locations if proximity between any two facilities is undesirable [61, 45]. Clique and clique partitioning are popular approaches for clustering wireless networks [40]. Maximal cliques are also used to model and avoid link interference in ad-hoc networks [32]. A dominating set in UDGs modeling wireless network function as a small set of nodes that can send an emergency communication to the entire graph [18]. Domination and connected domination are also used to cluster wireless networks. The vertices in a dominating set D are designated as *cluster-heads* and $N[v]$ for each $v \in D$ forms a cluster. Inside a cluster formed in this fashion, 2-hop communication is possible between any pair of nodes via the cluster-head. In mobile wireless networks, the nodes are weighted appropriately to find a weighted dominating that can yield cluster-heads that are less mobile [8, 16]. Alternately, if a virtual backbone is desirable among the cluster-heads, a connected dominating set is used in clustering [19, 20]. Clustering is an important problem in wireless networks as it helps routing and improves efficiency and throughput [57]. Graph coloring problems are used to solve channel assignment problems in wireless networks such as frequency assignment, code or time slot assignment depending on the protocols used [33]. The idea is that the chromatic number of the connectivity graph is the smallest set of frequency bands (time-slots or codes) required to communicate without interference. The UDG recognition problem also has applications in determining molecular conformations [34].

4 Models

4.1 Cliques

An $O(n^{4.5})$ time algorithm for finding a maximum clique in a UDG $G = (V, E)$ given the disk representation is presented in [18]. We briefly describe the ideas presented in [18]. Consider the set of disks $V = \{1, \ldots, n\}$ with centers at $c_i,\ \forall\, i \in V$. For a pair $i, j \in V$, $(i, j) \in E$ if and only if the Euclidean distance $L(c_i, c_j) \leq 1$. Denote by R_{ij} the region of intersection of two disks of *radius* $L(c_i, c_j)$ centered at c_i and c_j (see Figure 1). Let $H_{ij} \subseteq V$ denote the disks with centers in the region R_{ij}. Consider a maximum clique C and let $i, j \in C$ be the farthest pair (in terms of Euclidean distance) of vertices in C, then $C \subseteq H_{ij}$. If such a farthest pair i, j in some maximum clique C is known, then we only need to find a maximum clique in $G[H_{ij}]$ to find a maximum clique in G. Since such an i, j pair is unknown, we can enumerate over all $(i, j) \in E$ to derive a polynomial time algorithm, if we can solve the maximum clique problem in polynomial time on every $G[H_{ij}]$. This is facilitated by the following observation made in [18]. Consider the region R_{ij} with $L(c_i, c_j) \leq 1$. The line joining c_i and c_j bisects R_{ij} into R^1_{ij} and R^2_{ij}. The disk centers located in each half form a clique and hence the complement $\overline{G[H_{ij}]}$ is a bipartite graph. Since maximum independent set problem in bipartite graphs can be solved in $O(n^{2.5})$, we can find the maximum clique in $G[H_{ij}]$ in the same time which results in the claimed polynomial runtime. After the polynomial solvability was established in [18], the running time has been improved to $O(n^{3.5} \log n)$ in [9].

A relevant notion of *robust algorithms* for

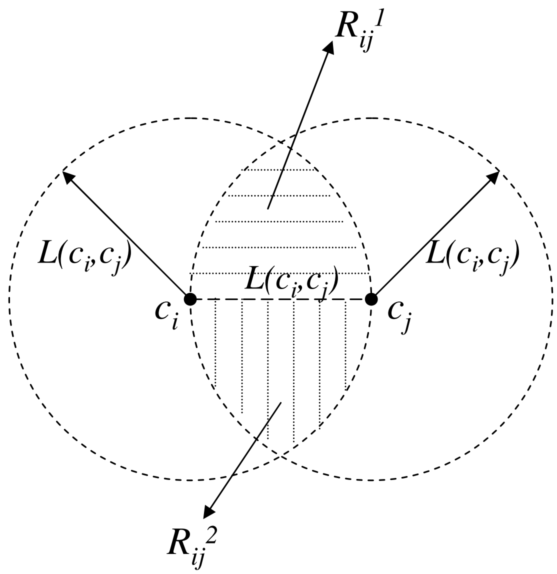

Figure 1: The region R_{ij} is shaded.

restricted graph classes such as UDGs was introduced recently in [55]. A robust algorithm for solving a problem on UDGs would accept only the graph G in standard format (adjacency list or matrix) as an input and solve the problem if it is indeed a UDG, or report that G is not a UDG. A polynomial time robust algorithm is presented in [55] for finding a maximum clique in UDGs (without the geometric representation) which returns a maximum clique or reports that G is not a UDG. The existence of a *polynomial* time *robust* algorithm for the maximum clique problem on UDGs is a surprising result given the NP-hardness of UDG recognition. A key idea is an ordering $L = e_1, e_2, \ldots, e_m$ of edges of G (input in standard format) referred to as a *co-bipartite neighborhood edge elimination ordering* (CNEEO). Denote by $G_L(i)$ the subgraph of G with edge set $\{e_i, e_{i+1}, \ldots, e_m\}$. Define for each edge $e_i = (u, v)$ the set $N_L(i)$ to be the set of vertices adjacent to both u and v in $G_L(i)$. The authors define an edge ordering L to be CNEEO if for each e_i, $N_L(i)$ induces a cobipartite (complement of bipartite) graph in G. The authors then prove that given G and a CNEEO L, a maximum clique can be found in polynomial time and describe a greedy algorithm for determining a CNEEO L if it exists or certifying that G has none in polynomial time. Finally, the authors show that every UDG admits a CNEEO there by completing the robust polynomial time algorithm for maximum clique problem on UDGs (in fact for the larger class of graphs that admit a CNEEO).

4.2 Independent Sets

Contrary to maximum clique, the maximum independent set (MIS) problem on UDGs is known to be NP-hard, even when the disk representation is given [18]. However, simple constant factor approximation algorithms and PTASs have been developed for this problem. Note that the strong NP-hardness of the MIS problem precludes the possibility of a FPTAS unless P=NP [30].

Given a graph G that does not contain a $(p + 1)$-*claw* as an induced subgraph, an $O(n \log n + m)$ algorithm is presented in [36] to find an independent set of size at least $\alpha(G)/p$. A p-claw is a graph on $p + 1$ vertices $V_p = \{u_0, u_1, \ldots, u_p\}$ such that u_0 is adjacent to all other vertices and $V_p \setminus \{u_0\}$ is an independent set. The algorithm proceeds by adding a vertex $v \in V$ to I followed by the removal of v and its neighbors *i.e.,* $N[v]$ from the graph. This step is repeated until V is empty, and the resulting independent set I is maximal. Let I^* denote a MIS in G. Suppose for the sake of argument that we sequentially removed vertices of $N[v]$ from I^* for each v removed from I. In any step, if v removed from I is also in I^*, the number of vertices in I^* deleted in that step is exactly one. If $v \in I \setminus I^*$, the number of vertices removed from I^* is at most p since a MIS in $N(v)$ has at most p vertices. Since I is maximal, I^* will be empty when I is empty and $\alpha(G) = |I^*| \leq |I \cap I^*| + p \times |I \setminus I^*| \leq p|I|$. By geometry, UDGs do not contain a 6-claw [45] and the above algorithm is a 5-approximation for the MIS problem on UDGs.

A simple 3-approximation algorithm is presented in [45] that, given a UDG G constructs an independent set of size at least $\alpha(G)/3$. This algorithm is based on the observation that every UDG has some vertex v such that $\alpha(G[N(v)]) \leq 3$. In particular, this is true for the vertex corresponding to the disk with minimum x-coordinate. Since every induced subgraph of a UDG is also a UDG, we can apply the same algorithm stated before from [36] and the observation will continue to hold in each step. But the vertex v added to I in each step is one with a MIS of size at most 3 in $N(v)$ yielding the desired approximation ratio. Given the disk representation, such a vertex v can be found easily in each step and without the disk representation such a vertex can certainly be found in polynomial time ($O(n^5)$).

The shifting strategy for geometric graphs introduced in [37], analogous to techniques for planar graphs introduced in [4] is the key ingredient in the PTASs developed for the MIS problem in [38, 47]. The approaches are similar and we follow the presentation in [38]. Let $G = (V, E)$ be the UDG and the center of each disk in V is specified. If we seek an independent set $IS[G]$ such that $|IS[G]| \geq (1 - \epsilon)\alpha(G)$, then choose parameter k to be the smallest integer for which $(k/(k+1))^2 \geq 1 - \epsilon$. Grid the region containing G with unit squares by dividing into horizontal and vertical strips of unit width. Assume that the intervals on the axes corresponding to each strip are left closed and right open. This is necessary to deal with disks with centers on the boundary of two strips. For some $0 \leq i \leq k$, delete all the disks with centers in every horizontal strip congruent to $i \mod (k+1)$ and denote the resulting UDG by $G(i)$. This leaves r disjoint horizontal "super"-strips of width k containing disjoint UDGs $G(i)_1, G(i)_2, \ldots, G(i)_r$ such that

$$G(i) = \bigcup_{1 \leq j \leq r} G(i)_j.$$

See Figure 2 and Figure 3. Varying i varies

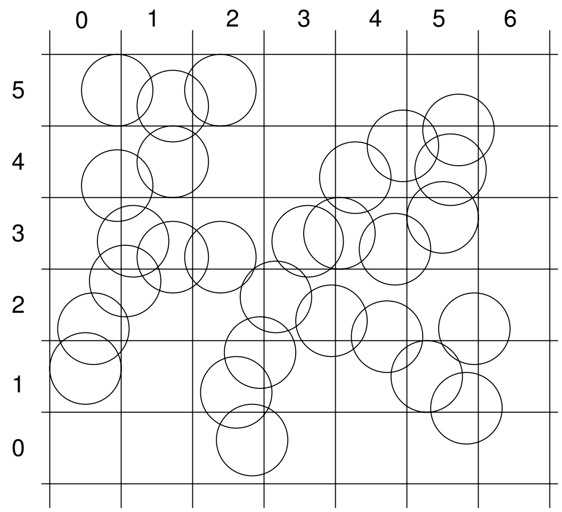

Figure 2: An example UDG G with a unit grid applied.

the deleted horizontal strips and hence permits us to *shift* the horizontal super-strips vertically over the graph G.

Now for some $1 \leq j \leq r$, select the j^{th} horizontal super-strip. For some $0 \leq l \leq k$, delete from the super-strip $G(i)_j$, all disks with centers in every vertical strip congruent to $l \mod (k+1)$ and denote the resulting UDG by $G(i,l)_j$. Parameter l can similarly be seen as the horizontal shift parameter for the vertical super-strips. This partitions $G(i,l)_j$ into s_j UDGs $G(i,l)_{j,1}, \ldots, G(i,l)_{j,s_j}$ each contained in a square block of side k. See Figure 4. Thus any MIS of $G(i,l)_{j,t}$ with $1 \leq j \leq r$ and $1 \leq t \leq s_j$ is of size at most $O(k^2)$ and can be found in time $n^{O(k^2)}$ by enumeration. The independent set returned is the union of independent sets from disjoint UDGs, but the one corresponding to the best block partition which depends on the shift parameters i, l. This can be expressed as follows. For a fixed vertical shift i and horizontal super-strip j, and for some choice of l,

$$IS[G(i,l)_j] := \bigcup_{1 \leq t \leq s_j} MIS[G(i,l)_{j,t}].$$

The best independent set in super-strip j is then obtained by horizontally shifting the

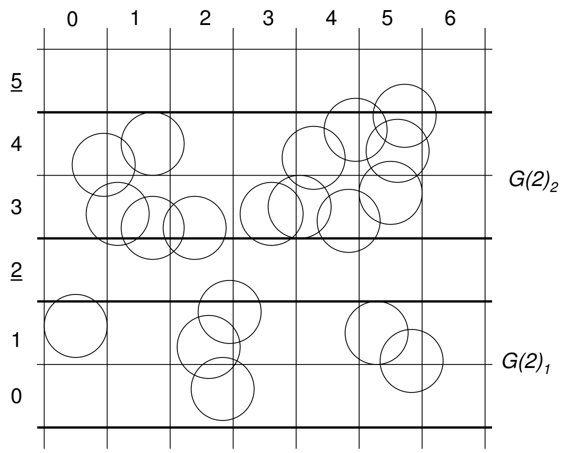

Figure 3: Graph $G(i)$ with $i = k = 2$. Disks centered in the underlined horizontal strips (2,5) have been deleted leaving 2 horizontal super-strips corresponding to disjoint graphs $G(2)_1$ and $G(2)_2$.

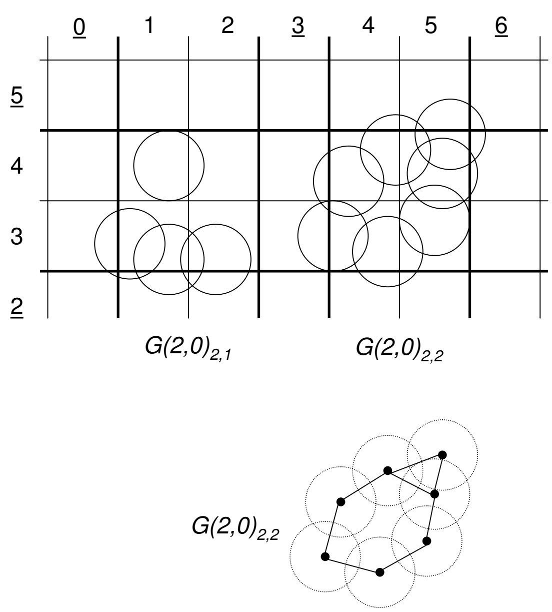

Figure 4: Graph $G(2, 0)_2$ with $l = 0$ (above). Disks centered in the underlined vertical strips (0,3,6) have been deleted leaving two 2×2 square blocks. Graph $G(2, 0)_{2,2}$ inside a 2×2 square block (below).

grid, *i.e.,* varying l, thus

$$IS[G(i)_j] := \max_{0 \le l \le k} IS[G(i, l)_j].$$

For each i, having found the best independent set in each super-strip,

$$IS[G(i)] := \bigcup_{1 \le j \le r} IS[G(i)_j].$$

By varying the horizontal shift parameter i, we can select the best independent set for the graph as

$$IS[G] := \max_{0 \le i \le k} IS[G(i)].$$

It is shown in [38] that

$$|IS[G]| \ge (k/(k+1))^2 \alpha(G)$$

and the algorithm has a running time of $n^{O(k^2)}$. By the choice of k, this implies a PTAS for the MIS problem on UDGs. Suggestions for improving running time and solution quality are presented in [38, 47]. After a vertical shift i, this involves solving the MIS problem optimally on each horizontal super-strip j using dynamic programming (DP) instead of approximating by horizontal shifting. This results in $|IS[G]| \ge (k/(k+1))\alpha(G)$ and a total running time of $n^{O(k)}$.

In either version, the shifting strategy helps us to divide-and-conquer by breaking down the graph into pieces on which optimal resolution is possible (by enumeration or DP) in polynomial time, but at the same time bound the error created by the division process as division is flexible. We refer to [38] for details on the performance guarantees mentioned.

Clearly, the disk representation is required for the above PTAS to work. The open problem of whether a *robust* PTAS exists for the problem in the sense described in Section 4.1 was settled positively in [51]. Given a graph $G = (V, E)$ (in standard format) and a desired error $\epsilon > 0$ we seek an independent set of size at least $\alpha(G)/\rho$ where $\rho = 1 + \epsilon$.

The algorithm starts with some arbitrary vertex v and finds a MIS I_k in $G[N_k(v)]$ for $k = 0, 1, 2, \ldots$ sequentially until the condition $|I_{k+1}| > \rho|I_k|$ is violated. Let r denote the smallest $k \geq 0$ for which $|I_{r+1}| \leq \rho|I_r|$. The authors show that there exists a constant (dependent on ρ) $c(\rho)$ such that $r \leq c(\rho)$ and each MIS I_k can be found in polynomial time. By the choice of r, we know that $\alpha(G[N_{r+1}(v)]) \leq \rho|I_r|$, *i.e.,* I_r is a ρ-approximate MIS for $G[N_{r+1}(v)]$. Suppose we have a ρ-approximate MIS I' for $G' = G[V \setminus N_{r+1}(v)]$, then clearly $I = I' \cup I_r$ is independent since $I' \subset V \setminus N_{r+1}(v)$ and $I_r \subset N_r(v)$. Furthermore, $\alpha(G) \leq \alpha(G[N_{r+1}(v)]) + \alpha(G') \leq \rho|I|$, *i.e.,* I is a ρ-approximate MIS of G. This fact combined with the fact that every vertex induced subgraph of a UDG is also a UDG, we have an inductive argument leading to the required PTAS. Robustness of the above algorithm is due to the following observations. The performance guarantee does not require G to be a UDG, and the algorithm always returns a $(1 + \epsilon)$-approximate solution. However, geometry of UDGs is required to establish polynomially bounded running times for finding MIS in k-neighborhoods and, the existence of a constant $c(\rho)$. The proof of these claims also shows that if there exists an independent set $I_r > (2r+1)^2$ (bound assumes unit-radius disk representation, which is equivalent to other representations discussed before) then G is not a UDG. Since this certificate can be obtained in polynomial time, this PTAS is robust.

4.3 Vertex Cover

The minimum vertex cover problem on UDGs is also NP-hard as shown in [18]. Given a UDG $G = (V, E)$, a polynomial time heuristic that does not require a disk representation to find a vertex cover of size at most $1.5\beta(G)$ is presented in [45]. This algorithm requires results from [50, 36]. The first result is the well-known Nemhauser-Trotter (NT) decomposition [50] which states given an arbitrary graph $G = (V, E)$ there exist disjoint vertex subsets P and Q such that (1) there exists a minimum vertex cover containing P; (2) if D is a vertex cover for $G[Q]$ then $D \cup P$ is a vertex cover for G; (3) any minimum vertex cover of $G[Q]$ contains at least $|Q|/2$ vertices. The second result from [36] states that following a NT decomposition, if $G[Q]$ can be colored using k colors, then $P \cup (Q \setminus S)$ is a vertex cover of size at most $2(1 - 1/k)\beta(G)$, where S is the largest color class in $G[Q]$.

The authors of [45] show that triangle-free UDGs can be colored using 4 colors. Given a UDG G, the heuristic first deletes vertices that form a triangle in G (call it V'). With $G := G - V'$, NT decomposition is then applied to the resulting triangle-free UDG G to identify sets P and Q. $G[Q]$ is then colored using 4 colors and the set $S \subseteq Q$ corresponding to the largest color class is identified. The heuristic then returns $V' \cup P \cup Q \setminus S$ as the approximate vertex cover. The approximation ratio follows by applying the *local-ratio principle* [45, 6, 7] as follows. In V' we pick 3 vertices for each triangle, and we have to pick at least 2. And for the triangle-free UDG, which is 4-colorable, the result from [36] applies. The running time of the heuristic is dominated by the time to obtain NT decomposition which can be accomplished in polynomial time.

A PTAS has been developed in [38] for minimum vertex cover that uses an approach similar to the PTAS for the MIS problem described in Section 4.2 from the same article. For $0 \leq i < k$, instead of deleting a horizontal strip congruent to $i \mod (k + 1)$, this approach uses super-strips of width $k + 1$ overlapping at horizontal strips congruent to $i \mod k$. Then solving the MIS problem exactly using DP on each super-strip $G(i)_j$ also yields a minimum vertex cover. For a fixed i, the union over $0 \leq j \leq r$ of minimum vertex covers of each $G(i)_j$ is a valid vertex cover for G and the smallest vertex cover found over all i has size at most $((k+1)/k)\beta(G)$. Details are available in [38].

4.4 Domination

Minimum dominating set (MDS) problem, minimum independent dominating set (MIDS) problem and minimum connected dominating set (MCDS) problem are known to be NP-hard for UDGs [18]. In fact, they are NP-hard even when restricted to a subclass of UDGs called *grid graphs* on which MIS is polynomial time solvable [18]. The observation that a maximal independent set is also a minimal dominating set is used frequently in approximating dominating sets. It has been proven in [45] that any maximal independent set in a UDG G is no larger than five times its domination number, *i.e.,* $\alpha(G) \leq 5\gamma(G) \leq 5\gamma_i(G)$. This follows from the observation that if D is a maximal independent set in a UDG G, then any vertex in a MDS (or a MIDS) can dominate at most 5 vertices in D. Any maximal independent set is hence a 5-approximate solution for the minimum dominating set (MDS) problem and the minimum independent dominating set (MIDS) problem. A 10-approximate algorithm for MCDS problem is also presented in [45]. This bound has been improved to 8 in several papers [2, 14, 12, 60] which present distributed implementations that are applicable in a practical setting in wireless networks. These heuristics construct a maximal independent set (which is dominating) and then connect it using a tree approach to obtain a CDS. This approach is based on the result from [2] that for a UDG G,

$$\alpha(G) \leq 4\gamma_c(G) + 1. \tag{1}$$

The maximal independent set I that is constructed (in polynomial time [60]) also has the property that for any $I' \subset I$, I' and $I \setminus I'$ are exactly distance two away from each other *i.e.,* there exist $u_1 \in I'$ and $u_2 \in I \setminus I'$ with $d(u_1, u_2) = 2$. The maximal independent set is connected using a spanning tree approach in [2, 12, 60] and using a *Steiner tree* in [14]. The bound (1) was improved recently in [62] to

$$\alpha(G) \leq 3.8\gamma_c(G) + 1.2. \tag{2}$$

This tighter bound shows that the 8-approximate algorithms such as the ones from [60, 14] are in fact 7.8-approximate. It is also observed in [62] that if

$$\alpha(G) \leq a\gamma_c(G) + b,$$

then $2.5 \leq a \leq 3.8$, which suggested that further improvement of their result was possible. Recently in [28], it has been shown that

$$\alpha(G) \leq 3.453\gamma_c(G) + 8.291 \tag{3}$$

for UDGs and a distributed algorithm is presented that finds a CDS of size at most $6.91\gamma_c(G) + 16.58$.

Using the bound (2), a 6.8-approximate algorithm for the MCDS problem has also been proposed recently in [48] that connects a maximal independent set using a Steiner tree approach. Given a set of vertices designated as *terminals*, a tree connecting the terminals such that every leaf is a terminal is called a *Steiner tree*. The non-terminal nodes are called *Steiner nodes*. In principle, we could find a Steiner tree with minimum number of Steiner nodes (ST-MSN) with a maximal independent set I as terminals and the Steiner nodes S_I^* union I yields a CDS for the UDG G. Instead of solving ST-MSN optimally, the authors approximate this problem. This is sufficient if we make the following observation.

From a MCDS D, we can obtain a solution for ST-MSN problem with terminals I as follows. We can find a spanning tree in $G[D]$; add an edge between each vertex in $I \setminus D$ and some vertex in D; and remove any leaf which is not a terminal in the resulting tree. The Steiner nodes in this solution are contained in D. Hence in the optimal solution to the ST-MSN problem, the number of Steiner nodes is at most $\gamma_c(G)$.

In the approach taken in [48], first a maximal independent set I is found such that every subset of I and its complement are exactly distance two apart. The authors develop and use a 3-approximate algorithm for the ST-MSN problem on UDGs given terminals I (see [48] for details). Denote the Steiner nodes in the 3-approximate solution for the ST-MSN problem by S_I, then its size is at most $3\gamma_c(G)$. Thus, the CDS $S_I \cup I$ has size at most $6.8\gamma_c(G) + 1.2$. This appears to be the approach with best performance guarantee available presently.

The weighted version of the MDS and MCDS problems, where the vertices of the UDG G are weighted and the objective is to find a dominating or a connected dominating set of minimum weight (sum of the weights of the selected vertices) have only been studied recently and the first constant factor approximation algorithms have been developed in [3]. A factor 72 approximation for MWDS and a factor 89 approximation for MWCDS are available and these problems appear to be more complicated than their unweighted counterparts.

A PTAS for the MDS problem given the disk representation was developed in [38], along similar lines as the schemes proposed for maximum independent set and minimum vertex cover problems. MCDS problem also has a PTAS developed in [17] for UDGs when the UDG is presented in its disk representation. In this work, an approximation algorithm running in time $n^{O((s \log s)^2)}$ is presented that constructs a CDS of size no larger than $(1 + 1/s)\gamma_c(G)$. The algorithm uses a grid based divide-and-conquer approach in combination with the shifting strategy. A robust PTAS for the MDS problem on UDGs was proposed recently in [52].

We briefly describe the robust PTAS for MDS on UDGs from [52]. Given a graph $G = (V, E)$, the authors define a *2-separated* collection of subsets $\mathcal{S}$, as $\mathcal{S} = \{S_1, \ldots, S_k\}$ with $S_i \subset V, i = 1, \ldots, k$ satisfying

$$\forall i \neq j, d(s,t) > 2,\ \forall s \in S_i,\ \forall t \in S_j.$$

If $D(S)$ denotes a MDS of $G[S]$, the authors show that for a 2-separated collection $\mathcal{S}$ in G,

$$\gamma(G) = |D(V)| \geq \sum_{i=1}^{k} |D(S_i)|.$$

Furthermore, if we have subsets T_i such that $S_i \subset T_i, i = 1, \ldots, k$ and a bound $\rho \geq 1$ such that

$$|D(T_i)| \leq \rho |D(S_i)|, \forall i = 1, \ldots, k, \quad (4)$$

and

$$D' = \bigcup_{i=1,\ldots,k} D(T_i) \text{ dominates } G, \quad (5)$$

then D' is a ρ-approximate MDS of G. This is true since,

$$|D'| \leq \sum_{i=1}^{k} |D(T_i)| \leq \rho \sum_{i=1}^{k} |D(S_i)| \leq \rho |D(V)| = \rho\gamma(G).$$

Given a UDG $G = (V, E)$ and an $\epsilon > 0$, the algorithm in [52] constructs in polynomial time (for fixed ϵ), a 2-separation S_i and the supersets T_i with $\rho = 1 + \epsilon$ satisfying the required properties (4), (5). This is accomplished as follows. First, we start with an arbitrary vertex $v_1 \in V_1 := V$ and compute a MDS $D(N_k(v_1))$ of $N_k(v_1)$ for $k = 0, 1, 2, \ldots$ until the condition

$$|D(N_{k+2}(v_1))| > \rho |D(N_k(v_1))|$$

is violated. Denote by r_1 the smallest k for which the above condition is violated, *i.e.,* $|D(N_{r_1+2}(v_1))| \leq \rho |D(N_{r_1}(v_1))|$. Then we iterate this procedure for the graph induced by $V_{i+1} := V_i \setminus N_{r_i+2}(v_i)$ until $V_{i+1} = \emptyset$. Note that in the subsequent iterations, the k-neighborhood is defined with respect to the current graph $G[V_{i+1}]$. Suppose this procedure terminates after K iterations, let $T_i = N_{r_i+2}(v_i)$ and let $S_i = N_{r_i}(v_i)$ for $i = 1, \ldots, K$. The authors then show that $S_1, \ldots, S_K$ is a 2-separated collection and $\bigcup_{i=1}^{K} D(T_i)$ dominates G. The termination condition for each iteration, $|D(T_i))| \leq$

$\rho|D(S_i)|$, guarantees the required approximation ratio.

It is noted in [52] that G needs not be a UDG to derive the approximation ratio, however it is necessary to show polynomial time solvability. The running time guarantee is provided by the following results from [52]. Firstly, the number of iterations $K \leq n$ and in each iteration i, the MDS in each k-neighborhood $N_k(v_i)$ can be found in polynomial time since its size is shown to be bounded by a polynomial function of k, and finally the number of k-neighborhoods considered is also bounded since $r_i \leq c(\rho)$ where $c(\rho)$ is a constant that depends only on the desired approximation factor ρ. Finally, this PTAS can be made robust by utilizing the same certification approach to show the graph is not a UDG used for the MIS problem developed by the same authors, described in Section 4.2.

4.5 Coloring and Clique Partitioning

The graph coloring problem on UDGs is known to be NP-hard. In [18], 3-colorability of UDGs is shown to be NP-complete and hence it follows that no approximation algorithm can achieve a ratio within 4/3, unless P=NP. In fact k-colorability of UDGs is NP-complete for any fixed $k \geq 3$ [31]. A simple 3-approximation algorithm for the problem was presented in [45] based on results from [36, 58]. Let

$$p(G) = \max_{H \subseteq G} \delta(H),$$

the largest p such that G contains a subgraph H of minimum degree p. Then

$$\chi(G) \leq p(G) + 1 \text{ [58]}$$

and $p(G)$ can be found in $O(m + n)$ steps [58, 36] as follows. Let $p := 0$ and let v be a vertex of minimum degree in G. Repeating the steps, $p := \max\{p, \delta(G)\}$ followed by $G := G - v$ until no vertices remain in G, finds $p(G)$. If we denote by v_i, the vertex removed in step i, each v_i then has at most $p(G)$ neighbors in $v_{i+1}, \ldots, v_n$. Processing the vertices in the order $v_n, \ldots, v_1$, and coloring each vertex with the smallest color not yet assigned to any of its neighbors already colored, guarantees a coloring of G with at most $p(G) + 1$ colors. If G is a UDG, then it is proven in [45] that

$$p(G)/3 + 1 \leq \chi(G).$$

Using similar approaches, it has also been shown in [54] that a UDG G can be colored using no more than $3\omega(G) - 2$ colors. A 3-approximate algorithm for coloring UDGs using network flow and matching techniques is also available from [31].

Clique partitioning is NP-complete even when restricted to *coin graphs* (UDGs where all overlaps are tangential) [15]. A polynomial time 3-approximate algorithm for this problem that uses the disk representation is available from [15]. The algorithm proceeds by first partitioning the plane into horizontal strips of width $\sqrt{3}$. A disk belongs to strip i if its center lies on the strip. A disk with its center on the boundary is assigned to the strip on top. Let G_i denote the UDG induced by disks in strip i and $V(G_i)$ are vertex disjoint. Solve the minimum clique partitioning problem exactly on each G_i and let Z_i denote the collection of cliques. The authors observe that this can be accomplished in polynomial time by coloring the complement since each G_i is a *cocomparability graph* [15, 9]. The clique partition returned by the algorithm is $Z := \bigcup_i Z_i$ and it can be shown that $|Z| \leq 3\bar{\chi}(G)$ as follows. Let Z^* denote a minimum clique partition of UDG G and let Z_i^* be the restriction of Z^* to G_i obtained by excluding the vertices not in $V(G_i)$ from the cliques in Z^*. Z_i^* is a valid clique partition of G_i and hence $|Z_i^*| \geq |Z_i|$. If C is a clique in Z^*, the authors observe that based on geometric arguments, the centers of disks in C must lie inside three consecutive strips. Hence, each C in Z^* is a union of at most 3 disjoint cliques from Z^*_{j-1}, Z^*_j, and Z^*_{j+1} for some j. Hence we have,

$$|Z| = \sum_i |Z_i| \leq \sum_i |Z^*_i| \leq 3|Z^*|.$$

The running time of the approximation algorithm is dominated by the exact solution step on each strip resulting in $O(n+\bar{m})$ where $\bar{m}$ denotes the number of edges in $\bar{G}$.

4.6 Related Results

A survey of on-line and off-line approximation algorithms for independent set and coloring problems on UDGs and general *disk graphs* (intersection graphs of disks of arbitrary radii) can be found in [23]. A short survey of results for cliques, independent sets and coloring of disk graphs is also available in [27]. A survey of complexity results on recognizing several variants of UDGs can be found in [35]. PTAS for *maximum weighted independent set* and *minimum weighted vertex cover* problems on intersection models of disks are available in [24].

A notion of *thickness* of UDGs is introduced and *fixed parameter tractability* of maximum independent set, minimum vertex cover and minimum (connected) dominating set problems (with thickness as parameter) is established in [59]. A parameterized algorithm running in $n^{O(\sqrt{k})}$ for finding an independent set of size k on *bounded ratio disk graphs* (the ratio of maximum diameter to minimum diameter is bounded by a constant) are presented in [1].

Several variants of the classical vertex coloring problem have been considered on UDGs, primarily motivated by different frequency assignment problems that arise in wireless networks. Apart from natural generalizations of UDGs such as general disk graphs, and bounded ratio disk graphs mentioned before, other generalizations of UDGs such as *Quasi* UDGs [42], *bisectored* UDGs [53] and *double disk graphs* [44] have also been developed motivated by wireless applications. Coloring problems have been studied in the context of these generalizations.

Algorithms for *distance constrained labeling*, which is a generalization of the well-known vertex coloring problem, for disk graphs are presented in [26]. Another variant of coloring called the *multicoloring* problem on UDGs is considered in [49]. The notion of *conflict-free coloring* is introduced and studied in the context of disk graphs in [25]. A k-improper coloring of a graph is one in which each color class induces a subgraph of maximum degree k. Note that 0-improper coloring is a proper coloring by the standard definition. For fixed k, the k-improper coloring problem has been shown to be NP-complete in [39]. Coloring and other problems on *bisectored unit disk graphs*, which generalize UDGs to allow for the phenomenon of *cell sectorization* in wireless communication are studied in [53]. Another generalization of UDGs motivated by frequency assignment problems in wireless networks are *double disk graphs*. Here, two concentric disks of arbitrary radii are associated with each vertex, and two vertices are adjacent if the inner disk of one intersects the outer disk of the other. For instance, one could think of the inner disk as the receiver range and the outer disk as the transmission range. Coloring problems on these graphs are studied and constant factor approximation algorithms are developed in [44, 22]. Hierarchical models of UDGs formed by a sequence of labeled UDGs is considered in [46]. PTAS for the maximum independent set, minimum dominating set, minimum clique cover, and minimum vertex coloring problems for UDGs specified hierarchically are developed in [46].

The notion of *well-separated pair decomposition* [13] with applications in geometric proximity problems is studied in the context of UDGs and algorithms for the same are developed in [29]. In [41], the hardness of approximately embedding UDGs is considered. Given a UDG $G = (V, E)$, let $L(c_u, c_v)$ denote the Euclidean distance between centers c_u, c_v of discs $u, v \in V$ in an embedding $emb(G)$. The authors define the quality of an embedding $emb(G)$ as

$$q(emb(G)) = \frac{\max\limits_{(u,v)\in E} L(c_u, c_v)}{\min\limits_{(u,v)\notin E} L(c_u, c_v)}.$$

Note that for any proper unit disk embedding $emb(G)$, the numerator of $q(emb(G))$ is at most 1 and the denominator is more than 1. The authors of [41] then show that finding an embedding $emb(G)$ for a UDG G such that $q(emb(G)) \leq \sqrt{3/2} - \epsilon$ where $\epsilon \to 0$ as $n \to \infty$ is NP-hard.

A data structure referred to as *extended doubly connected edge list* is developed in [43] for representing UDGs which facilitate faster implementation of routing algorithms in mobile wireless networks.

Max-cut and *max-bisection* problems in UDGs are shown to be NP-hard in [21].

5 Conclusions

In this chapter, we have surveyed results from literature on classical combinatorial optimization problems such as the maximum clique, maximum independent set, minimum vertex cover, minimum (connected) domination, graph coloring and minimum clique partitioning on unit-disk graphs. Brief descriptions of the approaches taken to solve these problems and the key ideas involved have been explained. Several recent results from literature have also been presented. A summary of important results surveyed can be found in Table 1.

References

[1] J. Alber and J. Fiala. Geometric separation and exact solutions for the parameterized independent set problem on disk graphs. *Journal of Algorithms*, 52(2):134–151, 2004.

[2] K. M. Alzoubi, P.-J. Wan, and O. Frieder. Distributed heuristics for connected dominating sets in wireless ad hoc networks. *Journal of Communications and Networks*, 4:22–29, 2002.

[3] C. Ambühl, T. Erlebach, M. Mihal'ák, and M. Nunkesser. Constant-factor approximation for minimum-weight (connected) dominating sets in unit disk graphs. Technical Report CS-06-008, Department of Computer Science, University of Leicester, Leicester, United Kingdom, 2006.

[4] B. S. Baker. Approximation algorithms for NP-complete problems on planar graphs. *Journal of the ACM*, 41(1):153–180, 1994.

[5] B. Balasundaram and S. Butenko. Graph domination, coloring and cliques in telecommunications. In M. G. C. Resende and P. M. Pardalos, editors, *Handbook of Optimization in Telecommunications*, pages 865–890. Spinger Science + Business Media, New York, 2006.

[6] R. Bar-Yehuda and S. Even. On approximating a vertex cover for planar graphs. In *STOC '82: Proceeding of the Fourteenth Annual ACM Symposium on Theory of Computing*, pages 303–309, 1982.

[7] R. Bar-Yehuda and S. Even. A local-ratio theorem for approximating the weighted vertex cover problem. *Annals of Discrete Mathematics*, 25:27–46, 1985.

[8] S. Basagni. Distributed clustering for ad hoc networks. In *Proceedings of the 1999 International Symposium on Parallel Architectures, Algorithms and Networks (ISPAN '99)*, page 310, 1999.

Table 1: A summary of results surveyed in this chapter.

Problem	Complexity	Constant factor	PTAS	Robust algo.
Clique	In P [18]	N/A	N/A	Poly-time [55]
Independent Set	NPC [18]	3 [45]‡	[38]†	PTAS [51]
Vertex cover	NPC [18]	1.5 [45]‡	[38]†	
Domination	NPC [18]	5 [45]‡	[38]†	PTAS [52]
Connected Domination	NPC [18]	6.8 [48]‡	[17]†	
Coloring	NPC [18, 31]	3 [45]‡		
Clique Partitioning	NPC [15]	3 [15]‡		

† Algorithm requires disk representation.
‡ Algorithm does not use a disk representation, but graph must be a UDG to ensure running time and/or performance guarantees.

[9] H. Breu. *Algorithmic Aspects of Constrained Unit Disk Graphs*. PhD thesis, University of British Columbia, 1996.

[10] H. Breu and D. G. Kirkpatrick. Unit disk graph recognition is NP-hard. *Computational Geometry: Theory and Applications*, 9(1-2):3–24, 1998.

[11] R. L. Brooks. On coloring the nodes of a network. *Proc. Cambridge Philos. Soc.*, 37:194–197, 1941.

[12] S. Butenko, X. Cheng, D.-Z. Du, and P. Pardalos. On the construction of virtual backbone for ad hoc wireless networks. In S. Butenko, R. Murphey, and P.M Pardalos, editors, *Cooperative Control: Models, Applications and Algorithms*, pages 43–54. Kluwer Academic Publisher, 2003.

[13] P. B. Callahan and S. R. Kosaraju. Algorithms for dynamic closest pair and n-body potential fields. In *SODA '95: Proceedings of the sixth annual ACM-SIAM symposium on discrete algorithms*, pages 263–272, Philadelphia, PA, USA, 1995. Society for Industrial and Applied Mathematics.

[14] M. Cardei, X. Cheng, X. Cheng, and D. Z. Du. Connected domination in multihop ad hoc wireless networks. In H. J. Caulfield, S. H. Chen, H. D. Cheng, R. J. Duro, V. Honavar, E. E. Kerre, M. Lu, M. G. Romay, T. K. Shih, D. V., P. P. Wang, and Y. Yang, editors, *Proceedings of the 6th Joint Conference on Information Science*, pages 251–255. JCIS / Association for Intelligent Machinery, Inc., 2002.

[15] M. R. Cerioli, L. Faria, T. O. Ferreira, and F. Protti. On minimum clique partition and maximum independent set on unit disk graphs and penny graphs: complexity and approximation. In *Latin-American Conference on Combinatorics, Graphs and Applications*, volume 18 of *Electron. Notes Discrete Math.*, pages 73–79 (electronic). Elsevier, Amsterdam, 2004.

[16] M. Chatterjee, S. Das, and D. Turgut. WCA: A weighted clustering algorithm for mobile ad hoc networks. *Journal of Cluster Computing, Special Issue on Mobile Ad hoc Networks*, 5:193–204, 2002.

[17] X. Cheng, X. Huang, D. Li, W. Wu, and D. Z. Du. A polynomial-time approximation scheme for the minimum-connected dominating set in ad hoc wireless networks. *Networks*, 42(4):202–208, 2003.

[18] B. N. Clark, C. J. Colbourn, and D. S. Johnson. Unit disk graphs. *Discrete Mathematics*, 86:165–177, 1990.

[19] B. Das and V. Bharghavan. Routing in ad-hoc networks using minimum connected dominating sets. In *Proceedings of IEEE International Conference on Communications*, pages 376–380, 1997.

[20] B. Das, R. Sivakumar, and V. Bharghavan. Routing in ad-hoc networks using a virtual backbone. In *Proceedings of the International Conference on Computers and Communication Networks (IC3N)*, pages 1–20, 1997.

[21] J. Diaz and M. Kaminski. Max-cut and max-bisection are NP-hard on unit disk graphs. Technical Report RRR-2006-23, Rutgers Center for Operations Research, Rutgers University, Piscataway, New Jersey, 2006.

[22] H. Du, X. Jia, D. Li, and W. Wu. Coloring of double disk graphs. *Journal of Global Optimization*, 28(1):115–119, 2004.

[23] T. Erlebach and J. Fiala. Independence and coloring problems on intersection graphs of disks. In *Approximation Algorithms in Combinatorial Optimization*, number 3484 in Lecture Notes in Computer Science, pages 135–155. Springer Verlag, 2006.

[24] T. Erlebach, K. Jansen, and E. Seidel. Polynomial-time approximation schemes for geometric graphs. In *SODA '01: Proceedings of the 12th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 671–679, Philadelphia, PA, USA, 2001. Society for Industrial and Applied Mathematics.

[25] G. Even, Z. Lotker, D. Ron, and S. Smorodinsky. Conflict-free colorings of simple geometric regions with applications to frequency assignment in cellular networks. *SIAM Journal on Computing*, 33(1), 2004.

[26] J. Fiala, A. V. Fishkin, and F. Fomin. On distance constrained labeling of disk graphs. *Theoretical Computer Science*, 326(1-3):261–292, 2004.

[27] A. V. Fishkin. Disk graphs: A short survey. In *Approximation and Online Algorithms*, volume 2909 of *Lecture Notes in Computer Science*, pages 260–264. Springer, Berlin, 2004.

[28] S. Funke, A. Kesselman, U. Meyer, and M. Segal. A simple improved distributed algorithm for minimum CDS in unit disk graphs. *ACM Transactions on Sensor Networks*, 2(3):444–453, 2006.

[29] J. Gao and L. Zhang. Well-separated pair decomposition for the unit-disk graph metric and its applications. *SIAM Journal on Computing*, 35(1):151–169, 2005.

[30] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-completeness*. W.H. Freeman and Company, New York, 1979.

[31] A. Gräf, M. Stumpf, and G. Weißenfels. On coloring unit disk graphs. *Algorithmica*, 20(3):277–293, 1998.

[32] R. Gupta and J. Walrand. Approximating maximal cliques in ad-hoc networks. In *Proceedings of the PIMRC 2004*, September 2004.

[33] W. K. Hale. Frequency assignment: theory and applications. *Proceedings of the IEEE*, 68(12):1497–1514, 1980.

[34] T. F. Havel. *The Combinatorial Distance Geometry Approach to the Calculation of Molecular Conformation*. PhD thesis, University of California, Berkeley, California, 1982.

[35] P. Hliněný and J. Kratochvíl. Representing graphs by disks and balls (a survey of recognition-complexity results). *Discrete Mathematics*, 229(1-3):101–124, 2001. Combinatorics, graph theory, algorithms and applications.

[36] D. S. Hochbaum. Efficient bounds for the stable set, vertex cover and set packing problems. *Discrete Applied Mathematics*, 6(3):243–254, 1983.

[37] D. S. Hochbaum and W. Maass. Approximation schemes for covering and packing problems in image processing and vlsi. *Journal of the ACM*, 32(1):130–136, 1985.

[38] H. B. Hunt III, M. V. Marathe, V. Radhakrishnan, S. S. Ravi, D. J. Rosenkrantz, and R. E. Stearns. Nc-approximation schemes for np- and pspace-hard problems for geometric graphs. *Journal of Algorithms*, 26(2):238–274, 1998.

[39] R. J. Kang, F. Havet, and J.-S. Sereni. Improper colouring of unit disk graphs. In *Proceedings of the 7th International Conference on Graph Theory (ICGT'05), Electronic Notes in Discrete Mathematics*, volume 22.

[40] P. Krishna, M. Chatterjee, N. H. Vaidya, and D. K. Pradhan. A cluster-based approach for routing in ad-hoc networks. In *Proceedings of the USENIX Symposium on Location Independent and Mobile Computing*, pages 1–8, 1995.

[41] F. Kuhn, T. Moscibroda, and R. Wattenhofer. Unit disk graph approximation. In *DIALM-POMC '04: Proceedings of the 2004 joint workshop on foundations of mobile computing*, pages 17–23, New York, NY, USA, 2004. ACM Press.

[42] F. Kuhn and A. Zollinger. Ad-hoc networks beyond unit disk graphs. In *DIALM-POMC '03: Proceedings of the 2003 joint workshop on Foundations of mobile computing*, pages 69–78, New York, NY, USA, 2003. ACM Press.

[43] J. Li, X. Liang, H. Selveraj, V. Muthukumar, and L. P. Gewali. A novel data structure for unit disk graphs. *Journal of Combinatorial Mathematics and Combinatorial Computing*, 54:145–156, 2005.

[44] E. Malesińska, S. Piskorz, and G. Weißenfels. On the chromatic number of disk graphs. *Networks*, 32(1):13–22, 1998.

[45] M. V. Marathe, H. Breu, H. B. Hunt III, S. S. Ravi, and D. J. Rosenkrantz. Simple heuristics for unit disk graphs. *Networks*, 25:59–68, 1995.

[46] M. V. Marathe, V. Radhakrishnan, H. B. Hunt III, and S. S. Ravi. Hierarchically specified unit disk graphs. *Theoretical Computer Science*, 174(1–2):23–65, 1997.

[47] T. Matsui. Approximation algorithms for maximum independent set problems and fractional coloring problems on unit disk graphs. In *Discrete and Computational Geometry*, number 1763 in Lecture Notes in Computer Science, pages 194–200. Springer Verlag, 2000.

[48] M. Min, H. Du, X. Jia, C. X. Huang, S. C.-H. Huang, and W. Wu. Improving construction for connected dominating set with steiner tree in wireless sensor networks. *Journal of Global Optimization*, 35(1):444–453, 2006.

[49] Y. Miyamoto and T. Matsui. Multicoloring unit disk graphs on triangular lattice points. In *SODA '05: Proceedings of the sixteenth annual ACM-*

SIAM symposium on Discrete algorithms, pages 895–896, Philadelphia, PA, USA, 2005. Society for Industrial and Applied Mathematics.

[50] G. L. Nemhauser and L. E. Trotter. Vertex packing: structural properties and algorithms. *Mathematical Programming*, 8:232–248, 1975.

[51] T. Nieberg, J. Hurink, and W. Kern. A robust PTAS for maximum weight independent sets in unit disk graphs. In *Graph-Theoretic Concepts in Computer Science*, number 3353 in Lecture Notes in Computer Science, pages 214–221. Springer, 2004.

[52] T. Nieberg and J. L. Hurink. A PTAS for the minimum dominating set problem in unit disk graphs. Technical Report 1732, Department of Applied Mathematics, University of Twente, Enschede, The Netherlands, 2004.

[53] J. Nolan. Bisectored unit disk graphs. *Networks*, 43(3):141–152, 2004.

[54] R. Peeters. On coloring j-unit sphere graphs. Technical report, Department of Economics, Tilburg University,, 1991. Cited by [31].

[55] V. Raghavan and J. Spinrad. Robust algorithms for restricted domains. *Journal of Algorithms*, 48(1):160–172, 2003.

[56] R. Ramaswami and K. K. Parhi. Distributed scheduling of broadcasts in a radio network. In *Proceedings of the INFOCOM*, 1989.

[57] I. Stojmenovic, editor. *Handbook of Wireless Networks and Mobile Computing*. Wiley InterScience, 2002.

[58] G. Szekeres and H. S. Wilf. An inequality for the chromatic number of a graph. *Journal of Combinatorial Theory*, 4:1–3, 1968.

[59] E. J. van Leeuwen. Approximation algorithms for unit disk graphs. In *Graph-theoretic concepts in computer science*, volume 3787 of *Lecture Notes in Computer Science*, pages 351–361. Springer, Berlin, 2005.

[60] P.-J. Wan, K. M. Alzoubi, and O. Frieder. Distributed construction of connected dominating set in wireless ad hoc networks. *Mobile Networks and Applications*, 9(2):141–149, 2004.

[61] D. W. Wang and Y.-S. Kuo. A study on two geometric location problems. *Information Processing Letters*, 28(6):281–286, 1988.

[62] W. Wu, H. Du, X. Jia, Y. Li, and S. C.-H. Huang. Minimum connected dominating sets and maximal independent sets in unit disk graphs. *Theoretical Computer Science*, 352(1):1–7, 2006.

MSC2000: 05C85, 05C69, 05C15, 05C62, 90C27, 90C59
Key words and phrases: unit-disk graphs, cliques, independent sets, dominating sets, coloring, approximation algorithms, PTAS