Technical Report

Department of Computer Science
and Engineering
University of Minnesota
4-192 EECS Building
200 Union Street SE
Minneapolis, MN 55455-0159 USA

TR 02-003

Polynomial-Time Approximation Scheme for Minimum Connected Dominating Set in Ad Hoc Wireless Networks

Xiuzhen Cheng, Xiao Huang, Deying Li, and Ding-zhu Du

January 22, 2002

Polynomial-Time Approximation Scheme for Minimum Connected Dominating Set in Ad Hoc Wireless Networks

Xiuzhen Cheng [*] Xiao Huang [†] Deying Li [‡] Ding-Zhu Du [*]

Abstract

A connected dominating set in a graph is a subset of vertices such that every vertex is either in the subset or adjacent to a vertex in the subset and the subgraph induced by the subset is connected. The minimum connected dominating set is such a vertex subset with minimum cardinality. An application in ad hoc wireless networks requires the study of the minimum connected dominating set in unit-disk graphs. In this paper, we design $(1+1/s)$-approximation for the minimum connected dominating set in unit-disk graphs, running in time $n^{O((s \log s)^2)}$.

[*]Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN 55455, USA. Email: {cheng,dzd}@cs.umn.edu.

[†]3M Center, Building 0235-03-F-08, St. Paul, MN 55144, USA. Email: xchuang@mmm.com.

[‡]Department of Computer Science, City University of Hong Kong, Hong Kong, China. Email: dyli@cs.cityu.edu.hk.

1 Introduction

Ad hoc wireless networking attracts more and more attentions in these years[2, 6, 8, 12]. It will revolutionize information gathering and processing in both urban environments and inhospitable terrain. Ad hoc wireless network is an autonomous system consisting of mobile hosts (or routers) connected by wireless links. It can be quickly and widely deployed. Example applications of ad hoc wireless network include emergency search-and-rescue operations, decision making in the battlefield, data acquisition operations in inhospitable terrain, etc.

Two important features of ad hoc wireless network are *dynamic topology* and *resource limitation*. In an ad hoc wireless network, every host can move to any direction at any time and any speed. There is no fixed infrastructure and central administration. A temporary infrastructure can be formed in any way. Due to multipath fading, multiple access, background noise and interference from other transmissions, an active link between two hosts may become invalid abruptly. Thus communication link is unreliable and retransmission is quite often for reliable services. The resource constraints for an ad hoc wireless network include battery capacity, bandwidth, CPU, etc. These two features make routing decisions very challenging.

Existing routing protocols rely on *flooding* for the dissemination of topology update packets (*proactive routing protocols* [5]) or route request packets (*reactive routing protocols* [9][13]). Network wide flooding (global flooding) may cause the following two problems:

- *Broadcast storm problem* [12]. Network-wide flooding may result in excessive *redundancy*, *contention*, and *collision*. This causes high protocol overhead and interference to other ongoing communication traffic.
- *Flooding is unreliable* [8]. "in moderately sparse graphs the expected number of nodes in the network that will receive a broadcast message was shown to be as low as 80% [15]."

To overcome or at least alleviate these problems, *virtual backbone-based routing* strategy is introduced [2][6][16]). Note that virtual backbone mimics the wired backbone. The most important benefit of virtual backbone-based routing is the dramatic reduction of protocol overhead, thus greatly improve the network throughput. This is achieved by propagating control packets inside the virtual backbone, not the whole network. Other benefits include the support of broadcast/multicast traffic and the propagation of "link quality" information for QoS routing [14].

Based on these applications, we can summarize the essential requirements for a virtual backbone as follows: (i) the number of hosts in the backbone is minimized; (ii) all hosts in the backbone are connected; (iii) each of the hosts not in the backbone has at least one neighbor in the backbone. This is clearly the idea of a minimum connected dominating set. A *connected dominating set* on a graph is a subset of vertices such that (a) every vertex is either in the subset or adjacent to a vertex in the subset and (b) the subgraph induced by the subset is connected. The problem of minimum connected dominating set (MCDS) is to compute a connected dominating set of minimum cardinality.

On the other hand, we assume an ad hoc wireless network contains only homogeneous mobile hosts. Each host is supplied with an equal-power omni-directional antenna. Similar assumptions are taken by most researchers in the field of mobile ad hoc wireless networking. Thus the footprint of an ad hoc wireless network is a unit-disk graph. Indeed, in a *unit-disk graph*, the vertex set consists of finite number of points on the Euclidean plane and an edge exists between two vertices (points) if and only if the distance between them is at most one.

According to the above analysis, we formulate the problem of constructing a virtual backbone to the problem of *minimum connected dominating set (MCDS) in unit-disk graphs*.

MCDS in general graphs has been studied by [7], which proposes a reduction from set-cover problem. This implies that for any fixed $0 < \varepsilon < 1$, no polynomial time algorithm with performance ratio $(1-\varepsilon)H(\Delta)$ exists unless $NP \subset DTIME[n^{O(log\ log\ n)}]$ [10], where Δ is the maximum degree and H is the harmonic function. MCDS in unit-disk graphs is still NP-hard [4]. The best know performance ratio of previous polynomial-time approximations is a contant > 7 [1, 3, 11]. In this paper, we will propose a Polynomial Time Approximation Scheme (PTAS) for MCDS in unit-disk graphs.

An algorithm A is a Polynomial Time Approximation Scheme (PTAS) for a minimization problem with optimal cost OPT if the following is true: Given an instance I of the problem and a small positive error parameter ε, (i) the algorithm outputs a solution which is at most $(1+\varepsilon)\,OPT$; (ii) when ε is fixed, the running time is bounded by a polynomial in the size of the instance I. If there exists a PTAS for an optimization problem, the problem's instance can be approximated to any required degree.

2 Preliminary

A *dominating set* in a graph is a subset of vertices such that every vertex is either in the subset or adjacent to at least one vertex in the subset. If in addition, the subgraph induced by a dominating set is connected, then the dominating set is called a connected dominating set. The following is a well-known fact about the dominating set and the connected dominating set.

Lemma 2.1 *For any dominating set D in a connected graph, we can find at most $2(|D|-1)$ vertices to connect D. Moreover, if D_1^* and D_2^* are respectively the minimum dominating set and the minimum connected dominating set, then $|D_2^*| \leq 2(|D_1^*|-1)$.*

We are interested in the minimum connected dominating set in unit-disk graphs. The unit-disk graph has the following property.

Lemma 2.2 *Suppose a unit-disk graph G lies in an $m \times m$ square such that every vertex is away from the boundary with distance at least $1/2$. Then G has at most $\lfloor 4m^2/\pi \rfloor$ connected components.*

Proof. Let x denote the number of connected components of such a unit-disk graph. From each connected component, we choose a vertex and identify it with the center of a unit-disk. (A unit-disk has diameter one.) Such unit-disks are disjoint and all lie in the cell. Therefore, we have

$$x \cdot \pi(1/2)^2 < m^2.$$

Hence, $x < 4m^2/\pi$. □

It has been know that the minimum connected dominating set has some polynomial-time approximation with constant performance ratio [1, 3, 11]. Here, we quote a result from [3].

Lemma 2.3 *There exists a polynomial-time approximation for the minimum connected dominating set in unit-disk graphs, with performance ratio eight.*

In design of a PTAS for the minimum connected dominating set, we sometimes consider the following extension of the concept of dominating set and connected dominating set.

Consider a graph $G = (V, E)$. Suppose H is a subgraph of G. A subset D of vertices in G is said to be a *connected dominating set* in G for H if every vertex in H is either in D or adjacent to a vertex in D, and in addition, the subgraph of G induced by D is connected.

3 Main Results

For input connected unit-disk graph $G = (V, E)$, we initially find a minimal square Q to contain all vertices in V. Without loss of generality, assume $Q = \{(x, y) \mid 0 \leq x \leq q, 0 \leq y \leq q\}$. Let m be a large integer that we will determine later. Let $p = \lfloor q/m \rfloor + 1$. Consider the square $\bar{Q} = \{(x, y) \mid -m \leq x \leq mp, -m \leq y \leq mp\}$. Partition $\bar{Q}$ into $(p+1) \times (p+1)$ gride so that each cell is a $m \times m$ square. This partition of $\bar{Q}$ is denoted by $P(0, 0)$ (Fig. 1).. In general, the partition $P(a, b)$ is obtained from $P(0, 0)$ by shifting the left-bottom corner of $\bar{Q}$ from $(-m, -m)$ to $(-m + a, -m + b)$.

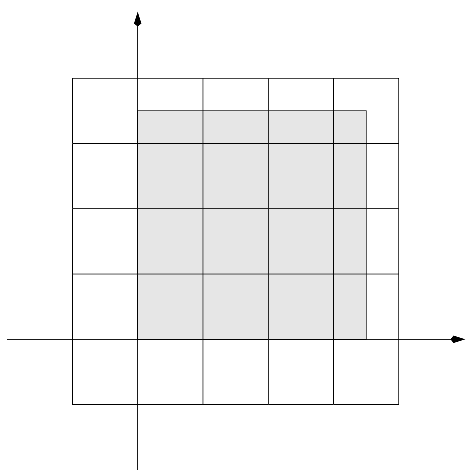

Figure 1: Squares Q and $\bar{Q}$.

For each cell e as an $m \times m$ square, we denote by $C_e(d)$ the set of points in e away from the boundary by distance at least d, e.g., $C_e(0)$ is the cell e itself. Fix a positive integer h whose value will be determined later. We will call $C_e(h)$ the *central area* of e and $C_e(0) - C_e(h+1)$ the *boundary area* of e (Fig.2). For simplicity of notation, we denote $B_e(d) = C_e(0) - C_e(d)$. Note that for each cell its boundary area and central area are overlapping with width one. For each partition $P(a,a)$, denote by $C^a(d)$ ($B^a(d)$) the union of $C_e(d)$ ($B_e(d)$) for e over all cells in $P(a,a)$. $C^a(h)$ and $B^a(h+1)$ are called the *central area* and the *boundary area* of $P(a,a)$.

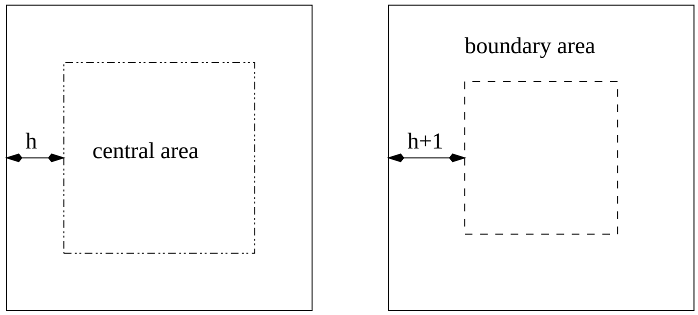

Figure 2: Cental area and boundary area.

For a graph G, denote by $G_e(d)$ ($\tilde{G}_e(d)$) the subgraph of G induced by all vertices lying in $C_e(d)$ ($B_e(d)$) and by $G^a(d)$ ($\tilde{G}^a(d)$) the subgraph of G induced by all vertices lying in $C^a(d)$ ($B^a(d)$).

Let $G=(V,E)$ be an input connected unit-disk graph. Consider a subgraph $G_e(h)$. This subgraph may consist of several connected components. For each connected component H, we compute the minimum connected dominating set D_H in $G_e(0)$ for H by brutal search. The union of D_H for H over all connected components of $G_e(h)$ is denoted by K_e. Thus, K_e has the property that for every connected component H of $G_e(h)$, K_e has a connected component dominating H. Now, we denote by K^a the union of K_e for e over all cells in partition $P(a,a)$.

By Lemma 2.3, we can compute in polynomial time a connected dominating set F for input connected graph G within a factor of 8 from optimal. Set $A^a = K^a \cup \tilde{F}^a(h+1)$. (note: We consider F as a graph without edge. According to above definition, $\tilde{F}^a(h+1) = F \cap B^a(h+1)$.)

Lemma 3.1 *For $0 \le a \le m-1$, A^a is a connected dominating set for input graph G. Moreover, A^a can be computed in time $n^{O(m^2)}$.*

Proof. A_a is clearly a dominating set for input graph G. We next show its connectivity. Note that for any connected component H of the subgraph $G_e(h)$ for some cell e in partition $P(a,a)$, if a connected component E of $\tilde{F}^a(h+1)$ has a vertex in H, then E must connect to the connected dominating set D_H for H. Therefore, the connectivity of A^a follows from the connectivity of F.

To establish the time for computing A^a, we note the fact that for a square with edge length $\sqrt{2}/2$, all vertices lying inside the square induces a complete subgraph in which any vertex must dominate all other vertices. It follows from this fact that the minimum dominating set for V_e has size $\leq (\lceil\sqrt{2}m\rceil)^2$. Hence, the minimum connected dominating set for V_e is at most $3 \leq (\lceil\sqrt{2}m\rceil)^2$. Therefore, $|K_e| \leq 3(\lceil\sqrt{2}m\rceil)^2$. Suppose cell e contains n_e vertices of input unit-disk graph. Then the number of candidates for each dominater in K_e is at most

$$\sum_{k=0}^{3(\lceil\sqrt{2}m\rceil)^2} \binom{n_e}{k} = n_e^{O(m^2)}.$$

Hence, computing A^a can be done in time

$$\sum_e n_e^{O(m^2)} \leq (\sum_e n_e)^{O(m^2)} = n^{O(m^2)}.$$

$\square$

By Lemma 3.1, we may take A^a to approximate the minimum connected dominating set. The next lemma will help us estimate the approximation performance of A^a.

Lemma 3.2 *Suppose $h = 7 + 3\lfloor\log_2(4m^2/\pi)\rfloor$. Let D^* be the minimum connected dominating set for input graph G. Then $|K^a| \leq |D^*|$ for $0 \leq a \leq m-1$.*

Proof. Recall that $G^a(h)$ is the subgraph of input graph $G = (V,E)$ induced by its vertices lying in the central area $C^a(h)$ of the partition $P(a,a)$. Let D be a minimum connected dominating set in G for $G^a(h)$. Then, we must have $|D| \leq |D^*|$.

Now, let $G[D]$ be the subgraph of G induced by D. We first claim that $G[D]$ has a spanning tree T without crossing edges in the plane. In fact, suppose T is a spanning tree of $G[D]$ with the minimum number of crosspoints. If T contains two edges (u,v) and (x,y) crossing at a point w in the plane. Without loss of generality, assume (u,w) is the longest one among four segments (u,w), (v,w), (x,w) and (y,w) (Fig. 3). Removal (x,y) from T would break T into two connected components containing vertices x and y respectively. One of them contains edge (u,v). Note that $d(x,v) \leq d(x,w)+d(v,w) \leq d(u,w)+d(w,v) \leq 1$ and $d(y,v) \leq d(y,w)+d(v,w) \leq d(u,w)+d(w,v) \leq 1$.

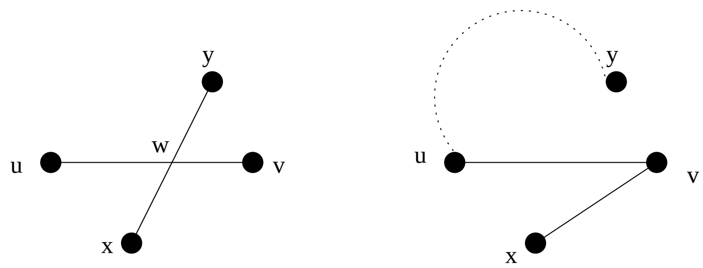

Figure 3: Two edges (u,v) and (x,y) have crosspoint w.

Therefore, we can add either (x,v) or (y,v) to connect the two connected components of $T-(x,y)$ into one. This operation removed a crosspoint, contradicting the assumption that T has the minimum number of crosspoints.

Assume T is a spanning tree of $G[D]$ without crosspoint. Let T_b be the sub-forest of T induced by those vertices not dominating any vertex in $V^a(h)$. We next modify T to a forest with two operations.

Operation 1: If after deleting a vertex u of T_b, T still keeps the following property (B1), then delete u.

(B1) For any connected component H of $G^a(h)$, T connects every two vertices in $H \cap T$, i.e., T has a connected component dominating H.

Through Operation 1, T becomes a forest with property that deleting any vertex would destroy property (B1). Now, we apply the second operation to T.

Operation 2: If T_b has two adjacent vertices u and v both with degree two, then delete them and restore the property (B1) as follows: Note that deleting u and v breaks a connected component of T into two parts, say C_1 and C_2. Since T already passed Operation 1, there must exist a connected component H of G_c such that $T \cap H$ exists in both C_1 and C_2. Since $T \cap C_1$ and $T \cap C_2$ dominates H, there must exist either one vertex x in H such that x is dominated by both $T \cap C_1$ and $T \cap C_2$, or two adjacent vertices x and y in H such that x is dominated by $T \cap C_1$ and y is dominated by $T \cap C_2$. Therefore, adding either x or x and y to T would restore the property (B1).

After Operation 2 is employed once, it may be possible to apply Operation 1 again. At any time, if Operation 1 can be applied, then we use Operation 1; if Operation 1 cannot be applied but

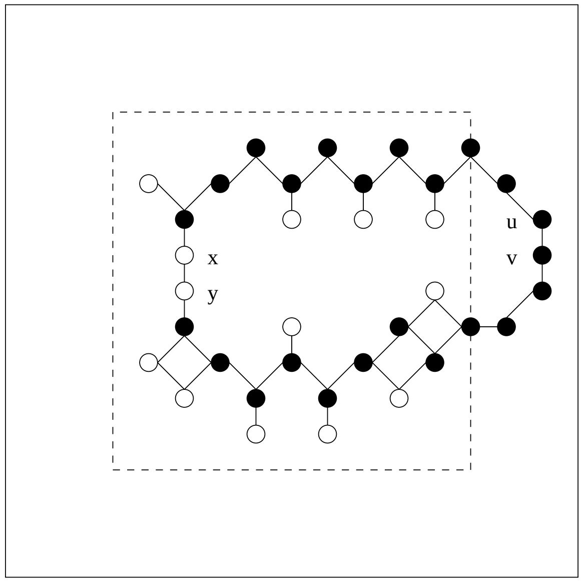

Figure 4: Operation 2

Operation 2 can be, then we employ Operation 2. Since both Operations 1 and 2 reduce the number of vertices in T_b, this process has to end in finitely many steps. At the end, forest T would still have property (B1) and in addition have the following properties:

(B2) T_b has no adjacent two vertices both with degree two.

(B3) T has at most $|D^*|$ vertices.

Since any vertex dominating some vertex in the central area of e must lie in $C_e(h-1)$, every vertex of T lying in $B_e(h-1)$ must belong to T_b. By Lemma 2.2, $T_e(h-1)$ has at most $\lfloor 4m^2/\pi \rfloor$ connected components.

Now, consider a maximal subtree T' of T such that

(C) T' has all leaves in $C_e(h-1)$ and all other vertices not in $C_e(h-1)$ (Fig. 5).

We claim that T' lies in cell e. To show our claim, suppose T' has k leaves. Since no two leaves can lie in the same connected component of $T_e(h-1)$, we have $k \leq \lfloor 4m^2/\pi \rfloor$. Note that $T'_e(h-1)$ consists of k leaves of T' and hence has k connected components. The *outer path* p of T' is a path between two leaves such that T' lies in the area between the path p and the boundary of $C_e(h)$. Since T has no crosspoint and T' is a maximal subtree satisfying (C), only vertices in path p is possible to

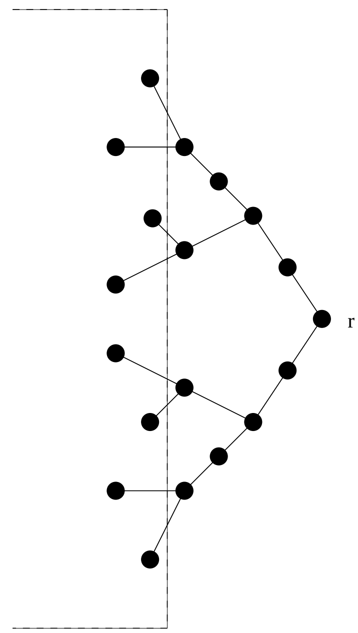

Figure 5: Tree T'

meet an edge in T but not in T'.

For contradiction, suppose T' has a vertex r lying outside of cell e. Without loss of generality, we may assume that r is on the path p. We consider r as a root for T' and study the k paths from leaves to r. The path p is broken at r into two such paths. Note that any path passes through area $C_e(h-4) - C_e(h-1)$ must meet an edge not on the path. (Otherwise, the path would contain two vertices in T_b both with degree two.) It follows that except the two paths obtained from path p, every path has to be merged into another one in area $C_e(h-4) - C_e(h-1)$. This means that these k paths become at most $2 + k/2$ paths when they go out from $C_e(h-4)$. Namely, $T_e'(h-4)$ contains at most $2 + (k-2)/2$ connected components. Similarly, $T_e'(h-1-3(\lfloor \log_2(k-2) \rfloor))(\subseteq T_e'(6))$ contains at most three connected components and $T_e'(3)$ contains at most two connected components. That is, all k paths in $C_e(3)$ have merged into two paths. Note that these two paths will merge into one at r lying outside of cell e. Therefore, each of them has a vertex u in area $C_e(0) - C_e(3)$ incident to an edge in $T - T'$, so that there exists a path in T from u to $C_{e'}(h-1)$ for another cell $e' \neq e$. This path at area $C_{e'}(0) - C_{e'}(3)$ can meet another similar path. In this way, we can find a path of infinite long in T, which is impossible since T contains no cycle. This contraction completes the proof of our claim that T' lies in cell e.

Note that every path with only two endpoints in $C_e(h-1)$ must be contained in such a maximal subtree T'. It follows immediately from our claim that for every connected component H of $G_e(h)$, $T_e(0)$ has a connected component dominating H since T has. This means that the number of vertices in $T_e(h)$ is at least $|K_e|$. Thus,

$$|K^a| = \sum_e |K_e| \leq \sum_e |T_e(h)| \leq |T| \leq |D^*|,$$

where $|T|$ denotes the number of vertices in T. □

We are ready to present the following main theorem.

Theorem 3.3 *Suppose* $h = 7 + 3\lfloor \log_2(4m^2/\pi) \rfloor$ *and* $\lfloor m/(h+1) \rfloor \geq 32s$*. Then there is at least a half number of* $i = 0, 1, ..., \lfloor m/(h+1) \rfloor - 1$ *such that* $\tilde{A}_{i(h+1)}$ *is* $(1+1/s)$*-approximation for the minimum connected dominating set.*

Proof. By Lemma 3.2, for every $i = 0, 1, ..., \lfloor m/(h+1) \rfloor - 1$, $|K^{i(h+1)}| \leq |D^*|$ where D^* is a minimum connected dominating set for G. Moreover, let F_H^a (F_V^a) denote the subset of vertices in $\tilde{F}^a(h+1)$ each with distance $< h+1$ from horizontal (vertical) boundary of some cell in $P(a,a)$. Then $\tilde{F}^a(h+1) = F_H^a \cup F_V^a$. Moreover, all $F_H^{i(h+1)}$ for $i = 0, 1, ..., \lfloor m/(h+1) \rfloor - 1$ are disjoint. Hence,

$$\sum_{i=0}^{\lfloor m/(h+1) \rfloor - 1} |F_H^{i(h+1)}| \leq |F| \leq 8|D^*|.$$

Similarly, all $F_V^{i(h+1)}$ for $i = 0, 1, ..., \lfloor m/(h+1) \rfloor - 1$ are disjoint and

$$\sum_{i=0}^{\lfloor m/(h+1) \rfloor - 1} |F_V^{i(h+1)}| \leq |F| \leq 8|D^*|.$$

Thus

$$\sum_{i=0}^{\lfloor m/(h+1) \rfloor - 1} |\tilde{F}^{i(h+1)}(h+1)| \leq \sum_{i=0}^{\lfloor m/(h+1) \rfloor - 1} (|F_H^{i(h+1)}| + |F_V^{i(h+1)}|) \leq 16|D^*|.$$

Therefore,

$$\begin{aligned}
& \sum_{i=0}^{\lfloor m/(h+1)\rfloor-1} |A^{i(h+1)}| \\
\leq & \sum_{i=0}^{\lfloor m/(h+1)\rfloor-1} (|K^{i(h+1)}| + |\tilde{F}^{i(h+1)}(h+1)|) \\
\leq & (\lfloor m/(h+1)\rfloor + 16)|D^*|.
\end{aligned}$$

That is,

$$\frac{1}{\lfloor m/(h+1)\rfloor} \sum_{i=0}^{\lfloor m/(h+1)\rfloor-1} |A^{i(h+1)}| \leq (1+1/(2s))|D^*|.$$

This means that there are at least a half number of $\tilde{A}_{i(h+1)}$ for $i = 0, 1, \lfloor m/(h+1)\rfloor - 1$ satisfying

$$|A^{i(h+1)}| \leq (1+1/s)|D^*|.$$

□

The following corollary follows immediately from the theorem.

Corollary 3.4 *There is a $(1+1/s)$-approximation for minimum connected dominating set in connected unit-disk graphs, running in time $n^{O((s\log s)^2)}$.*

Proof. Note that computing each A^a needs time $n^{O(m^2)}$. By Theorem 3.3, a $(1+1/s)$-approximation can be obtained by computing all $\lfloor m/(h+1)\rfloor$ A^a's and choosing the best one. Thus, the total running time is $(mn^{O(m^2)} = n^{O(m^2)}$. Choose m to be the least integer satisfying $\lfloor m/(h+1)\rfloor \geq 32s$, where $h = 7+3\lfloor \log_2(4m^2/\pi)\rfloor$. Then $m = O(s\log s)$. This completes the proof. □

4 Conclusion

We designed a PTAS for the minimum connected dominating set in unit-disk graphs. This is an evidence to show that currently existing implemented approximations have a large space for improvement.

References

[1] K.M. Alzoubi, P.-J. Wan and O. Frieder, New distributed algorithm for connected dominating set in wireless ad hoc networks, to appear in *Proc. HICSS 2002*.

[2] A.D. Amis and R. Prakash, Load-balancing clusters in wireless ad hoc networks Application-Specific Systems and Software Engineering Technology, *Proc. 3rd IEEE Symposium on*, 2000, pp. 25-32.

[3] X. Cheng and D.-Z. Du, Virtual backbone-based routing in ad hoc wireless networks, submitted to *IEEE Transactions on Parallel and Distributed Systems*.

[4] B. N. Clark, C. J. Colbourn and D. S. Johnson, Unit disk graphs, *Discrete Mathematics*, Vol. 86, 1990, pp. 165-177.

[5] T. Clausen, P. Jacquet, A. Laouiti, P. Minet, P. Muhlethaler, and L. Viennot, Optimized link state routing protocol, IETF Internet Draft, draft-ietf-manet-olsr-05.txt, October 2001.

[6] B. Das and V. Bharghavan, Routing in ad hoc networks using minimum connected dominating sets, *ICC '97*, Montreal, Canada, June 1997.

[7] S. Guha and S. Khuller, Approximation algorithms for connected dominating sets, Algorithmica, Vol. 20(4), April 1998, pp. 374-387.

[8] P. Johansson, T. Larsson, N. Hedman, B. Mielczarek and M. Degermark, Scenario-based performance analysis of routing protocols for mobile ad hoc networks, *Proc. IEEE MOBICOM*, Seattle, Aug. 1999, pp. 195-206.

[9] D.B. Johnson, D. A. Maltz, Y.-C. Hu and J. G. Jetcheva, The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks, Internet Draft draft-ietf-manet-dsr-05.txt, March 2001.

[10] C. Lund and M. Yannakakis, On the hardness of approximating minimization problems, *Journal of the ACM*, Vol. 41(5), pp. 960-981, 1994.

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

[12] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen and J.-P. Sheu, The broadcast storm problem in a mobile ad hoc network, *Proc. MOBICOM*, Seattle, Aug. 1999, pp. 151-162.

[13] C. E. Perkins, E. M. Royer and S. R. Das, Ad hoc On-Demand Distance Vector (AODV) Routing, Internet Draft draft-ietf-manet-aodv-08.txt, March 2001.

[14] R. Sivakumar, P. Sinha and V. Bharghavan, CEDAR: a core-extraction distributed ad hoc routing algorithm, *Selected Areas in Communications, IEEE Journal on*, Vol. 17(8), Aug. 1999, pp. 1454 -1465.

[15] P. Sinha, R. Sivakumar and V. Bharghavan, Enhancing ad hoc routing with dynamic virtual infrastructures, *INFOCOM 2001*, Vol. 3, pp. 1763-1772.

[16] J. Wu and H. Li, On calculating connected dominating set for efficient routing in ad hoc wireless networks, *Proc. of the 3rd International Workshop on Discrete Algothrithms and Methods for MOBILE Computing and Communications*, 1999, Seattle, WA USA, pp.7-14.