J Glob Optim (2013) 56:449–458
DOI 10.1007/s10898-011-9831-x

PTAS for the minimum k-path connected vertex cover problem in unit disk graphs

Xianliang Liu · Hongliang Lu · Wei Wang · Weili Wu

Received: 14 October 2011 / Accepted: 16 December 2011 / Published online: 29 December 2011

Abstract In the *Minimum k-Path Connected Vertex Cover Problem* (MkPCVCP), we are given a connected graph G and an integer $k \geq 2$, and are required to find a subset C of vertices with minimum cardinality such that each path with length $k-1$ has a vertex in C, and moreover, the induced subgraph $G[C]$ is connected. MkPCVCP is a generalization of the minimum connected vertex cover problem and has applications in many areas such as security communications in wireless sensor networks. MkPCVCP is proved to be NP-complete. In this paper, we give the first polynomial time approximation scheme (PTAS) for MkPCVCP in unit disk graphs, for every fixed $k \geq 2$.

Keywords PTAS · k-Path connected vertex cover · Unit disk graph

1 Introduction

Wireless Sensor Networks (WSN) has been a recently merged advanced technology with numerous applications in both military and civilian areas (e.g., surveillance in battlefield, disaster rescuing, environment monitoring, home automation, traffic control, electronics and wireless technologies and so on).

In many applications of WSN, it is usually important to ensure the security properties of WSN including confidentiality, authenticity, data integrity and so on. Traditional security techniques cannot be applied directly to WSN, because sensor devices usually have limited capabilities of computation, energy and communications. Moreover, they are often deployed

This work was supported in part by the Fundamental Research Funds for the Central Universities, the National Natural Science Foundation of China No. 11071191 and 11101329.

X. Liu · H. Lu · W. Wang (✉)
School of Science, Xi'an Jiaotong University, Xi'an 710049, People's Republic of China
e-mail: wang_weiw@163.com

W. Wu
Department of Computer Science, University of Texas at Dallas, Richardson, TX 75083, USA

in accessible areas, where they can be rather easily captured by attackers. In general, a standard sensor device is not considered as tamper-resistant. It is also undesirable to make all devices of a sensor network tamper-proof because of the increasing costs. Therefore, the design of WSN security protocols has become a challenge in security research field.

One of such protocols, known as the Canvas protocols, was designed in [4,6–8,10–12], which provides data integrity or data origin authentication [6] in a sensor network. The k-generalized Canvas scheme [8] guarantees data integrity if there at least one vertex is not captured on each path of the length $k-1$ in the communication graph. Thus, during the deployment and initialization of a sensor network, it should be ensured that, at least one protected node exists on each path of the length $k-1$ in the communication graph, and the problem of minimizing the cost of the network by minimizing the number of protected vertices is naturally arisen in [8], which can be formally described as follows:

Given a graph $G=(V,E)$ and an integer $k \geq 2$. A subset C of V is a *k-path vertex cover* (k-PVC) if each path of length $k-1$ contains a vertex in C. The minimum k-path vertex cover problem (MkPVCP) asks to find a k-PVC with minimum cardinality (denoted by $\psi_k(G)$).

Boš̌tjan et al. [1] proved that MkPVCP is NP-complete for each fixed $k \geq 2$, while for trees the problem can be solved in linear time. They also gave some upper bounds for $\psi_k(G)$ and showed in particular $\psi_k(G) \leq (2n+m)/6$ for every finite graph G with n vertices and m edges. Tu and Zhou [9] gave a 2-approximation for MkPVCP when $k=3$.

In this paper, we are mainly concerned with the minimum k-path vertex cover problem in unit disk graph (MkPCVCP-UDG) with connectivity requirement.

WSN is usually modelled by a unit disk graph (UDG), where the sensor nodes are corresponding to the vertices located on the Euclidean plane, and there is an edge between two vertices if and only if the Euclidean distance between them is at most one. When talking about a unit disk graph in this paper, we assume that the geometric location of each sensor is given, since it has been proved in [5] that determining whether a graph is a UDG is NP-complete.

Obviously, MkPCVCP-UDG is NP-complete, since for $k=1$, the problem is reduced to the minimum connected vertex cover (MCVC), which was shown to be NP-complete in [3] for UDGs. Zhang et al. [14] gave a PTAS for MCVC in UDGs. In this paper, we present the first PTAS for MkPCVC on UDGs, by using partition technique and shifting strategy. Such an approach was used for Steiner trees in the plane [13]. A more complicated approach was used for minimum connected dominating set [2].

Whereas our basic idea follows that of [2,14], we mention that there are a few parts of the design and analysis of our algorithm that involves some different ideas. For example, to ensure the connectivity of the k-path vertex cover, we have to add some additional vertices the number of which is minor compared with the optimal solution. Moreover, the time complexity analysis of the algorithm seems non-trivial.

The rest of the paper is organized as follows. In Sect. 2, some preliminaries are given which will be needed in the sequel. In Sect. 3 we present the algorithm and the proof of its correctness. Time complexity and performance analysis are given in Sects. 4 and 5, respectively.

2 Preliminaries

In this section, we introduce some notions and notations to help the later discussions.

Definition 2.1 P is called a k-path if P is a path which contains k vertices.

Definition 2.2 For two subgraphs G_1 and G_2 of G, the distance of G_1 and G_2 is the number edges of a shortest path of G connecting G_1 and G_2, denoted by $dist(G_1, G_2)$.

We use the notation $P = (v_1, v_2, \ldots, v_t)$ denote a path P which contains t vertices and $\{v_1, v_2, \ldots, v_t\}$ denote a set which contains t vertices. Moreover $G - \{v_1, v_2, \ldots, v_t\}$ denote the subgraph $G[V \setminus \{v_1, v_2, \ldots, v_t\}]$ of G induced by $V \setminus \{v_1, v_2, \ldots, v_t\}$.

Definition 2.3 (*k-PCVC cf. [1]*) A subset C of vertices of a graph G is called a k-path vertex cover if each path of order k in G contains at least one vertex from C. Moreover, if the subgraph $G[C]$ induced by C also connected in graph G, then C is said to be a k-path connected vertex cover.

Definition 2.4 Minimum k-Path Connected Vertex Cover Problem (MkPCVCP): Given a connected graph $G = (V, E)$ and an integer $k \geq 2$, find a k-path connected vertex cover set with minimum cardinality.

For any connected unit disk graph $G = (V, E)$, where $|V| = n$, we can obtain a PTAS for MkPCVCP, by using partition technique and shifting strategy.

First, we suppose all the disks associated with vertices of graph G are located in a square $Q = \{(x, y)|0 \leq x \leq q, 0 \leq y \leq q\}$. Set $p = \lfloor \frac{q}{m} + 1 \rfloor$ and $m = \lceil \frac{40(k-1)k^3}{\varepsilon} \rceil$, where ε is an arbitrary positive number. Let $\overline{Q} = \{(x, y)| -m \leq x \leq mp, -m \leq y \leq mp\}$. Partition $\overline{Q}$ into $(p+1)^2$ cells such that each cell is an $m \times m$ small square, excluding the top and right boundary edges. Then, this partition of $\overline{Q}$ is denoted by $p(0, 0)$. Second, In general, the partition $p(a, b)$ can be obtained by shifting the left-lower corner of $p(0, 0)$ from $(-m, -m)$ to $(-m + a, -m + b)$.

For each cell e of size $m \times m$ of $p(0, 0)$, we assume $e = \{(x, y)|im \leq x < (i+1)m, jm \leq y < (j+1)m\}$. Then, we can define its central area I_e and boundary area B_e as follows

$$I_e = \{(x, y)|im + 1 \leq x \leq (i+1)m - 1, jm + 1 \leq y \leq (j+1)m - 1\},$$
$$B_e = e - \{(x, y)|im + k \leq x \leq (i+1)m - k, jm + k \leq y \leq (j+1)m - k\}.$$

We shall use $G[I_e]$ to denote the subgraph of G induced by the vertices in I_e and use $comp(G[I_e])$ to denote the set of connected component in $G[I_e]$.

Note that, for each cell e, the central area I_e and the boundary area B_e have an overlap area of width $k - 1$. This ensures the output of Algorithm 1 (see Sect. 3) is a k-path vertex cover. If we add some new vertices by step 5 of the Algorithm 1, the connectedness of the output of the Algorithm 1 can also be ensured.

3 Algorithm overview

In this section, we present our PTAS for MkPCVCP in UDGs. Before doing so, we need a constant approximation for MkPCVCP, which is similar to the well-known 2-approximation for the minimum vertex cover.

Initially, let $A \leftarrow \emptyset$. At each iteration, we simply choose a path $P = (x_1, x_2, \ldots, x_k)$ on k vertices in G, and set $A \leftarrow A \cup \{x_1, \ldots, x_k\}$, then let $G \leftarrow G - \{x_1, \ldots, x_k\}$, repeated the process until there is no path of length $k - 1$ is left.

It is clearly the above algorithm gives a k-approximation for MkPVC. Next, we show that we can modify it into a k^2-approximation for MkPCVCP. We need the following lemma.

Lemma 3.1 *Let G be a connected graph and C be a k-path vertex cover of G. Then, there exist two connected components C_1 and C_2 of the induced subgraph $G[C]$ such that $dist(C_1, C_2) \leq k$.*

Proof Let C_1 and C_2 be two connected components of $G[C]$ with shortest distance in G. Suppose that $\Pi = (v_1, v_2, \ldots, v_t)$ is the shortest path in G connecting C_1 and C_2. If $t \geq k+2$, consider the subpath $(v_2, v_3, \ldots, v_{t-1})$, since C is a k-path vertex cover of G. Thus, there must exist a vertex v in the subpath that belongs to C. Let C_3 be the connected component of $G[C]$ which contains v. Then, we have

$$dist(C_1, C_3) < dist(C_1, C_2),$$

which contradicts with our choice of C_1 and C_2. □

Lemma 3.2 *Let $G = (V, E)$ be a connected graph. There is a polynomial time k^2-approximation for MkPCVCP.*

Proof Let C^* be an optimal solution for MkPCVCP of G. Suppose A_c is the output of the k-approximation mentioned above, and $\widehat{C}$ is an optimal solution for MkPVCP of G. For any k-path in A_c, there must exist a vertex belong to $\widehat{C}$. Then, we must have

$$|A_c| \leq k\widehat{C} \leq k|C^*|,$$

since the size of an optimal solution for MkPVCP cannot exceed the size an optimal solution for MkPCVCP.

Moreover, if A_c is not connected, we can reduce the number of connected component of A_c by one through adding at most $k-1$ vertices into A_c, until A_c becomes connected. So, we need to add at most $(t-1)(k-1)$ vertices into A_c to get a k-PCVC set C_0, where t $(t \leq |A_c|)$ is the number of connected component of A_c in graph G. Thus, we have

$$|C_0| \leq |A_c| + (t-1)(k-1) \leq k|A_c| \leq k^2|C^*|.$$

So, there is a k^2-approximation for MkPCVCP. □

Algorithm 1 (PTAS for MkPCVCP-UDG)

Input: A connected unit disk graph $G = (V, E)$ with $|V| = n$, a positive integer $k \geq 2$ and a real number $\varepsilon > 0$.

1. Let $m \leftarrow \lceil \frac{40(k-1)k^3}{\varepsilon} \rceil$.
2. Let $C_0 \subseteq V$ be a k^2-approximation to the MkPCVC for G.
3. For $a \leftarrow 0$ to $m-1$ do.
   (a) Let $C_0(a) \leftarrow \{v \in C_0 | v \text{ lies in the boundary area of } p(a, a)\}$.
   (b) Choose a^* such that $|C_0(a^*)| = min_{a \in \{0,1,\ldots,m-1\}} |C_0(a)|$.
4. For any component $H \in comp(G[I_e])$, use exhausted search to find a minimum k-PCVC C_H of H. Set $C[e] = \bigcup_{H \in comp(G[I_e])} C_H$.
5. For each cell e of $p(a^*, a^*)$. If there exists a connected component $H \in comp(G[I_e])$ such that $C_H \bigcap C_0 = \emptyset$, find a path $P_e(H)$ which connects C_H and $C_0(a^*)$ with the minimum length. Set $C_e = \bigcup_{H \in comp(G[I_e])} P_e(H)$; else, Set $C_e = \emptyset$.
6. Output $C \leftarrow C_0(a^*) \bigcup (\bigcup_{e \in p(a^*, a^*)} C[e]) \bigcup (\bigcup_{e \in p(a^*, a^*)} C_e)$.

Theorem 3.3 *The output C of algorithm 1 is a k-path connected vertex cover for unit disk graph G.*

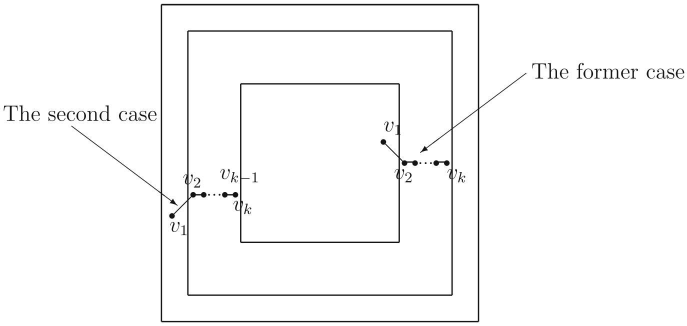
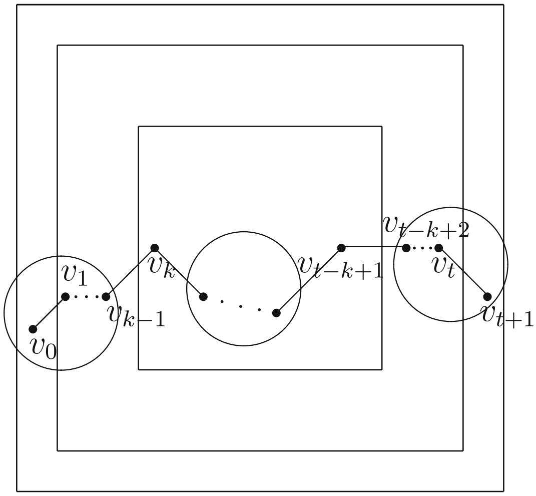


Fig. 1 Two cases that a path lies in a cell e

Fig. 2 The path $(v_1, \ldots, v_{t-1}, v_t)$ is in $H \in$ *comp* $(G[I_e])$



Proof First, for any path $(v_1, v_2, \ldots, v_k)$ with length $k-1$, the Euclidean distance between v_i and v_{i+1} is less than or equal to 1 for $i = 1, 2, \ldots, k-1$. It follows that the path $(v_1, v_2, \ldots, v_k)$ belong to either the central area of e or the boundary area of e, since the central area and boundary area have an overlap area with width $k-1$ for each cell e. In the former case, the path is in a component $H \in G[I_e]$. So, the path is covered by $C[e] \subseteq C$. In the second case, we also see that the path is covered by $C_0(a^*)$ (see Fig. 1). Thus, any path with length $k-1$ in G is covered by C, and C is a k-path vertex cover of G.

Second, we prove that the subgraph $G[C]$ induced by C is connected.

Let H_1 and H_2 be two distinct connected components in $G[C_0(a^*)]$ with shortest distance in $G[C_0]$. Since C_0 is connected, there is a path $\Pi = (v_0, v_1, \ldots, v_t, v_{t+1})$ of $G[C_0]$ connecting H_1 and H_2 through the central area of a cell e. Since the central area and the boundary area of each cell have an overlap with width $k-1$, we assume without loss of generality that $\{v_0, v_1, v_2, \ldots, v_{k-1}\} \subseteq V(H_1)$, $\{v_{t-k+2}, \ldots, v_t, v_{t+1}\} \subseteq V(H_2)$ and $\{v_k, v_{k+1}, \ldots, v_{t-k+1}\} \subseteq I_e \setminus B_e$. Then, we see that $\{v_1, \ldots, v_{k-1}, v_{t-k+2}, \ldots, v_t\} \subseteq I_e \bigcap B_e$. So, the path $(v_1, v_2, \ldots, v_{t-1}, v_t)$ is in $H \in comp(G[I_e])$. According to our Algorithm 1, there exists a connected component $C_H \in C[e]$ connecting H_1 and H_2 (see Fig. 2).

For each cell e of $p(a^*, a^*)$ and each connected component $H \in comp(G[I_e])$, there are two cases needed to be considered.

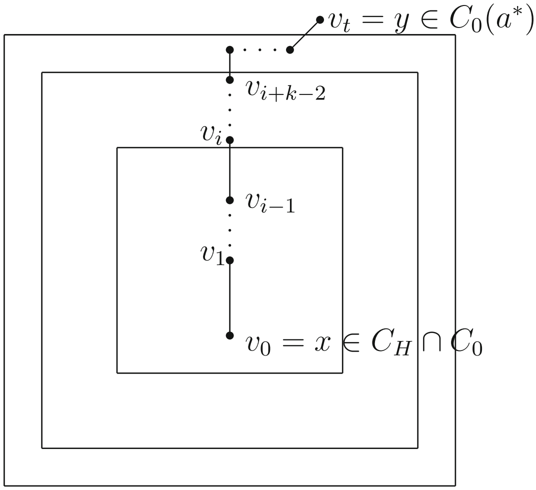


Fig. 3 C_H is connected with $C_0(a^*)$

(1) If there exists a connected component $H \in comp(G[I_e])$ such that $C_H \bigcap C_0 \neq \emptyset$, there must exist a vertex x in C_H belong to C_0. Then, there exists a path $P = (v_0, v_1, \ldots, v_t) \subseteq C_0$ connecting an other vertex $y \in C_0(a^*)$ which is belong to the other parts of G outside of e, since $G[C_0]$ is connected in G. Suppose $v_0 = x$, $v_t = y$ and $\{v_0, v_1, \ldots, v_{t-1}\} \subseteq e$. Let i be the index such that v_i is the first vertex on P with $v_i \in B_e$. Then, we see that $v_{i-1} \in I_e \setminus B_e, v_i, \ldots, v_{i+k-2} \in I_e$ and thus $v_{i-1}, v_i, \ldots, v_{i+k-2} \in I_e$. So, There must exist a vertex in $\{v_{i-1}, v_i, \ldots, v_{i+k-2}\}$ belong to C_H. Thus, C_H is connected with $C_0(a^*)$ (see Fig. 3).

(2) If there exists a connected component $H \in comp(G[I_e])$ such that $C_H \bigcap C_0 = \emptyset$, by step 5 of our Algorithm 1, there must exist a path $P_e(H)$ connecting C_H and $C_0(a^*)$ with minimum length.

For each case, we get the conclusion that C_H is connected with $C_0(a^*)$. So, we have that C is a k-path connected vertex cover for unit disk graph G. □

4 Time complexity

In this section, we give an analysis of the time complexity of Algorithm 1. Clearly, the step 4 of Algorithm 1 using exhausted search is the most time consuming part. So, we need to prove that the step 4 of Algorithm 1 can also be executed in polynomial time, for any fixed $k \geq 2$ and $\varepsilon > 0$.

Lemma 4.1 *If G is a UDG. Then, for any vertex v of G, there are at most 5 independent vertices in $N(v)$, where $N(v)$ is the set of vertices adjacent with v in G.*

Lemma 4.2 *Let H be a connected subgraph induced by some vertices in a cell e in $p(a^*, a^*)$. Assume that H does not contain a k-path. Let Δ be the maximum degree of vertices in H. Then, we have $\Delta \leq 6k$.*

Proof Let v be a vertex with maximum degree Δ in H. Divide the disk centered at v with diameter two into six parts. By the pigeonhole, there must exist a part of the disk has least $\lfloor \frac{\Delta}{6} \rfloor$

vertices. So, we have $\lfloor \frac{\Delta}{6} \rfloor < k$, for otherwise the connected component H would contain a k-path. Thus, we have $\Delta \le 6k$. □

Lemma 4.3 *For any $m \times m$ cell e of the partition $p(a, a)$, let H be a connected subgraph which does not contain a k-path in cell e. Then we have $|V(H)| \le g(k)$, where $g(k) = 4(6k)^k$.*

Proof Let Δ be the maximum degree of vertices in H. Then, by the above Lemma 4.2, we have $\Delta \le 6k$ and the diameter of the subgraph $G[H]$ induced by H is less than or equal to $k-1$, since H does not contain a k-path. So, we have

$$\begin{aligned} |V(H)| &\le k - 1 + 2[(\Delta - 2) + (\Delta^2 - 2) + \cdots + (\Delta^{\lceil \frac{k-3}{2} \rceil} - 2)] \\ &\le 2(\Delta + \Delta^2 + \cdots + \Delta^{\lceil \frac{k-3}{2} \rceil}) = 2\frac{\Delta(1 - \Delta^{\lceil \frac{k-3}{2} \rceil + 1})}{1 - \Delta} \le 2\frac{\Delta(\Delta^{\frac{k+1}{2}} - 1)}{\Delta - 1} \\ &\le 2\frac{\Delta(\Delta^{\frac{k+1}{2}} - 1)}{\frac{\Delta}{2}} \le 4\Delta^{\frac{k+1}{2}} \le 4\Delta^k \le 4(6k)^k. \end{aligned}$$

□

Lemma 4.4 *The number of independent unit disks in an $m \times m$ cell e is at most $\lceil \frac{4(m+1)^2}{\pi} \rceil$.*

Proof Enlarge the cell e to an $(m+1) \times (m+1)$ cell by adding a boundary with width 0.5. Then, all the disks in cell e lie in the $(m+1) \times (m+1)$ cell. Since each unit disk occupies area $\frac{\pi}{4}$, the result follows from the independence assumption. □

Theorem 4.5 *The time complexity of our Algorithm 1 is $n^{O(f(k)/\epsilon^2)}$, where n is the number of vertices in the graph and $f(k) = k^4 g(k)$.*

Proof For any connected component $H \in comp(G[I_e])$, consider the subgraph $G[V(H) \setminus C_H]$ induced by $V(H) \setminus C_H$. We know that $G[V(H) \setminus C_H]$ does not contain any k-path. Without a loss of generality, we assume $V(H) \setminus C_H = C_1 \bigcup C_2 \bigcup \cdots \bigcup C_s$, where C_i $(i = 1, 2, \ldots, s)$ are connected components of $G[V(H) \setminus C_H]$ and s is less than the potential of the maximum independent unit disk set. By Lemma 4.4, we have

$$s \le \left\lceil \frac{4(m+1)^2}{\pi} \right\rceil.$$

According to Lemma 4.3, we have

$$|V(H) \setminus C_H| = \left| C_1 \bigcup C_2 \bigcup \cdots \bigcup C_s \right| \le g(k) \left\lceil \frac{4(m+1)^2}{\pi} \right\rceil.$$

Now, we use the following strategy to compute C_H: enumerate the induced subgraphs of H with no more than $g(k)\lceil \frac{4(m+1)^2}{\pi} \rceil$ vertices to find out all induced subgraphs whose connected components does not contain a k-path. Then take complements and find out the one which is connected with minimum number of vertices.

The above exhausted search for C_H takes time at most

$$\sum_{i=0}^{g(k)\lceil \frac{4(m+1)^2}{\pi} \rceil} \binom{n_H}{i} = n_H^{O(g(k)m^2)},$$

where n_H is the number of vertices in H. Therefore, the total running time of Algorithm 1 is at most

$$\sum_{e,H} n_H^{O(g(k)m^2)} = \left(\sum_{e,H} n_H\right)^{O(g(k)m^2)} = n^{O(g(k)m^2)} = n^{O(f(k)/\epsilon^2)}.$$

□

5 Performance analysis

The following theorem shows that our Algorithm 1 is a PTAS.

Theorem 5.1 *Let C^* be an optimal solution to MkPCVCP in unit disk graph G, and C be the output of Algorithm 1. Then, we have $|C| \leq (1+\varepsilon)|C^*|$.*

Proof First, we prove that

$$|C_0(a^*)| \leq \frac{4k^3}{m}|C^*|.$$

According to the partition and shifting technique, the location of the graph G relative to the grid of the partition changes when the partition shifts (towards northwest). Without loss generality, we may imagine that the grid of the partition is fixed, but actually the graph G is moving (towards southwest). Thus, for each vertex v in C_0, it belongs to at most $4k$ of the sets $C_0(0), C_0(1), \ldots, C_0(m-1)$. Therefore, we have

$$\sum_{a=0}^{m-1} |C_0(a)| \leq 4k|C_0| \leq 4k \times k^2|C^*| = 4k^3|C^*|.$$

and thus, we have

$$|C_0(a^*)| \leq \frac{4k^3}{m}|C^*| \tag{1}$$

Second, we modify C^* into $\overline{C^*}$ by adding some vertices such that $\overline{C^*}$ satisfied the following condition:

(C_1) For each connected component H of $G[I_e]$, $\overline{C^*} \bigcap V(H)$ is a k-PCVC of H.

For each cell e, let $\overline{C_e^*} = C^* \bigcap I_e$. Obviously, we have $\overline{C_e^*} = C^* \bigcap I_e$ is a k-PVC of $G[I_e]$. Suppose there exists a connected component H of $G[I_e]$ such that the condition (C_1) is not satisfied. By Lemma 1, there are two connected components H_1 and H_2 of $G[\overline{C_e^*} \bigcap V(H)]$ such that H_1 and H_2 can be connected through adding at most $k-1$ vertices in $V(H) \setminus \overline{C_e^*}$. Now, add these vertices into $\overline{C_e^*}$. If the new $\overline{C_e^*}$ still does not satisfy condition (C_1). Then, continue above process to merge connected components. Without loss of generality, we assume that this has been done t times before $\overline{C_e^*}$ satisfied condition (C_1), and thus:

$$|\overline{C_e^*}| \leq \left|C^* \bigcap e\right| + (k-1)t \tag{2}$$

Third, we can proved that

$$\left|C_0(a^*) \bigcap e\right| \geq \frac{t}{5} \tag{3}$$

Suppose that the connected components are merged in the order H_1 and H_2, H_3 and $H_4, \ldots, H_{2t-1}$ and H_{2t}. For simplicity, we assume that all the above H_j's are distinct connected components. Let $x_i^1, x_i^2, \ldots, x_i^{k-1}$ be $k-1$ adjacent vertices in $V(H_{2i-1}) \bigcap B_e \bigcap I_e$, such that $x_i^1, x_i^2, \ldots, x_i^{k-1}$ is adjacent with a vertex $x_i^k \in B_e \backslash I_e$. Then, there must exist a vertex in $\{x_i^1, x_i^2, \ldots, x_i^k\}$ belonging to C_0. Let $w_i = x_i^j$, $j = 1, 2, \ldots, k$, if $x_i^j \in C_0$. Note that $x_i^j \in B_e$. Hence $w_i \in C_0(a^*) \bigcap e$. Let $\{x_i^1, x_i^2, \ldots, x_i^k\} \backslash \{w_i\}$ be charged on w_i. However, a vertex may be charge more than once as w_i's. For example, it is possible that there are two independent vertices x_i, x_j covered by the same vertex of C_0, since they belong to different components of $G[C^* \bigcap I_e]$. Then, by Lemma 4.1, such a vertex charges at most 5 times as w_i's. Thus, we have

$$5(k-1)\left|C_0(a^*) \bigcap e\right| \geqslant (k-1)t$$

and inequality (3) follows. So, we have

$$|\overline{C_e^*}| \leq \left|C^* \bigcap e\right| + 5(k-1)\left|C_0(a^*) \bigcap e\right| \tag{4}$$

Last, for each path $P_e(H)$, we know that, by the step 5 of Algorithm 1, $P_e(H)$ connects C_H and $C_0(a^*)$ with minimum cardinality. Since the central area I_e and the boundary area B_e have an overlap area with width $k-1$. So, there exists a path $P = (v_0, v_1, \ldots, v_t)$ connecting C_H and $B_e \setminus I_e$. Let $v_0 \in C_H$ and $v_t \in B_e \setminus I_e$. Then, there must exist a vertex $x \in \{v_0, v_1, \ldots, v_{k-1}\}$ belonging to $C_0(a^*)$. So, by the minimality of the path $P_e(H)$. We have

$$|P_e(H)| \leq k-2.$$

Charge the path $P_e(H)$ to the vertex x. Then, using a similar arguments as above, we have

$$\left|\bigcup_{e \in p(a^*, a^*)} C_e\right| \leq \sum_{e \in p(a^*, a^*)} 5(k-2)\left|C_0 \bigcap B_e \bigcap I_e\right| \leq 5(k-2)|C_0(a^*)|$$

Note that in Algorithm 1, $C[e]$ is the subset satisfying the condition (C_1) with minimum cardinality, for each cell e. It follows that $|C[e]| \leq |\overline{C_e^*}|$. Thus, we have

$$\begin{aligned}
\left|\bigcup_{e \in p(a^*, a^*)} C[e]\right| &= \sum_{e \in p(a^*, a^*)} |C[e]| \\
&\leq \sum_{e \in p(a^*, a^*)} |\overline{C_e^*}| \\
&\leq |C^*| + 5(k-1)|C_0(a^*)| \\
&\leq |C^*| + \frac{20(k-1)k^3}{m}|C^*|.
\end{aligned}$$

Thus, by Eqs. 1–5, we have

$$
\begin{aligned}
|C| &= \left| C_0(a^*) \bigcup \left(\bigcup_{e \in p(a^*,a^*)} C[e] \right) \bigcup \left(\bigcup_{e \in p(a^*,a^*)} C_e \right) \right| \\
&\le |C_0(a^*)| + \sum_{e \in p(a^*,a^*)} |C[e]| + \sum_{e \in p(a^*,a^*)} |C_e| \\
&\le 5(k-1)|C_0(a^*)| + |C^*| + \frac{20(k-1)k^3}{m}|C^*| \\
&\le |C^*| + \frac{40(k-1)k^3}{m}|C^*| \\
&\le (1+\varepsilon)|C^*|.
\end{aligned}
$$

This completes the proof. □

References

1. Brešar, B., Kardoš, F., Katrenič, J., Semanišin, G.: Minimum k-path vertex cover. Discrete Appl. Math. **159**, 1189–1195 (2011)
2. Cheng, X., Huang, X., Li, D., Wu, W., Du, D.: A polynomial-time approximation scheme for minimum connected dominating set in ad hoc wireless networks. Networks **42**, 202–208 (2003)
3. Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. Freeman, San Francisco (1978)
4. Gollmann, D.: Protocol Analysis for Concrete Environments, EUROCAST 2005. LNCS, vol. 3643, pp. 365–372. Springer, Heidelberg (2005)
5. Kratochvil, K.: Intersection graphs of noncrossing arc-connected sets in the plane. In: Proceedings of Symposium on Graph Drawing, GD'96. LNCS, vol. 1190, pp. 257–270 (1997)
6. Menezes, A., van Oorshot, P., Vanstone, S.: Handbook of Applied Cryptography. CRC Press, Boca Raton (1996)
7. Novotný, M.: Formal Analysis of Security Protocols for Wireless Sensor Networks, accepted to Tatra Mountains Mathematical Publications (2010)
8. Novotný, M.: Design and analysis of a generalized canvas protocol. In: Proceedings of WISTP 2010. LNCS 6033, pp. 106–121. Springer (2010)
9. Tu, T., Zhou, W.: A factor 2 approximation algorithm for the vertex cover P_3 problem. Inf. Process. Lett. **111**, 683–686 (2011)
10. Vogt, H.: Integrity preservation for communication in sensor networks. Technical report 434, Institute for Pervasive Computing, ETH Zürich (2004)
11. Vogt, H.: Exploring message authentication in sensor networks. In: Castelluccia, C., Hartenstein, H., Paar, C., Westhoff, D. (eds.) ESAS 2004. LNCS, vol. 3313, pp. 19–30. Springer, Heidelberg (2005)
12. Vogt, H.: Increasing attack resiliency of wireless ad hoc and sensor netmorks. In: Proceedings of the 2nd International Workshop on Security in Distributed Computing Systems (ICDCSW 2005), vol. 2, pp. 179–184. IEEE Computer Society, Washington (2005)
13. Wang, L.S., Jiang, T.: An approximation scheme for some steiner tree problems in the plane. Networks **28**, 187–193 (1996)
14. Zhang, Z., Gao, X., Wu, W.: PTAS for connected vertex cover in unit disk graphs. Theor. Comput. Sci. **410**, 5398–5402 (2009)