Performance Analysis and Improvement for the Construction of MCDS Problem in 3D Space

Jun Li[1], Xiaofeng Gao[1](✉), Guihai Chen[1], Fengwei Gao[1], and Ling Ding[2]

[1] Department of Computer Science and Technology, Shanghai Jiao Tong University, Shanghai 200240, China
lijun2009@sjtu.edu.cn, {gao-xf,gchen}@cs.sjtu.edu.cn, gfwsteven@gmail.com
[2] Institute of Technology, University of Washington Tacoma, Tacoma, USA
lingding@u.washington.edu

Abstract. In this paper, we discuss the problem of finding a minimum connected dominating set (MCDS) in 3-dimensional space, where the communication model is a unit ball graph (UBG). MCDS in UBG is proved to be an NP-complete problem, and currently the best approximation is 14.937 in [1]. However, their projection method during the approximation deduction process is incorrect, which overthrows its final bound completely. As a consequence, in this paper we will first propose a new projection method to overcome their problem, illustrate the cardinality upper bound of independent points in a graph (which will be used to analyze the approximation ratio), and then optimize the algorithms to select MCDS with prune techniques. The major technique we use is an adaptive jitter scheme, which solves the open question in this area.

1 Introduction

A connected dominating set (CDS) is widely used for many network applications. For instance, it can form a virtual backbone to take charge of routing and message transmission process. It can also be denoted as sink tunnels for data gathering or sensor detection. Hence, there are lots of researches which related to it in the literature [2,3]. A CDS is defined to be a subset of V in a given graph $G = (V, E)$, such that every vertex of V is either in this subset or adjacent to a vertex in this subset and this subset can induce a connected subgraph.

Most literature discussed CDS in two-dimensional space, and use a unit disk graph (UDG) to model the network. However, such model cannot precisely describe the non-flat area such as mountainous region or underwater environment. Correspondingly, we can use a unit ball graph (UBG) to model such a

This work has been supported in part by the National Natural Science Foundation of China (Grant No.61202024), Shanghai Pujiang Program (No.13PJ1403900), Shanghai Educational Development Foundation (Chenguang Grant No.12CG09), and the Natural Science Foundation of Shanghai (Grant No.12ZR1445000).

Z. Zhang et al. (Eds.): COCOA 2014, LNCS 8881, pp. 184–199, 2014.
DOI: 10.1007/978-3-319-12691-3_15

network in 3-dimensional space. In a UBG $G = (V, E)$, any two vertices are adjacent (or connected) if and only if the Euclidean distance between them is at most 1.

Since UBG can formulate a network environment more precisely than UDG, CDS in UBG can represent more applications than that in UDG. For instance, Wang and Li [4] constructed 3D landmark maps with vision data extracted from camera images, and then used 3D-CDS to improve data association in application of simultaneous localization and mapping (SLAM). Yang [5] implemented 3D-CDS as clusters to find an optimal topology control strategy in 3D wireless sensor networks. In all, it is significant to design fast algorithms for selecting an appropriate CDS set from a given network and analyze their performance. Typically, a CDS with minimum cardinality is the most efficient choice for practical use, and we refer it as MCDS.

Computing minimum CDS (MCDS) is a well-known NP-complete problem, and lots of approximation algorithms were proposed during last decade. Those algorithms often include two phases. Firstly, they choose an maximal independent set (MIS) from G. Second, they add some extra nodes from G to connect this MIS, usually by Steiner trees. An MIS in a graph $G = (V, E)$ is a subset $M \subseteq V$ such that any two vertices from M are not connected and we cannot add another vertex from $M \backslash V$ to form a bigger MIS. Easy to see, in UDG or UBG, the distance between any two vertices in M should be more than 1. To analyze the performance of those approximations, the ratio $mis(G)/mcds(G)$ plays an important role, where $mis(G)$ is the size of MIS the algorithm selected and $mcds(G)$ is the size of an optimal MCDS. In 2-dimensional situation, this approximation ratio has been widely studied. Based on the fact that the neighborhood area of any node can contain at most five independent points, Wan et al. [6] proposed that $mis(G) \leq 4mcds(G) + 1$. Later, Wu et al. [7] improved this ratio to 3.8 by proving that the neighborhood of any two adjacent nodes can contain at most 8 nodes. In [8], Gao et al. showed the bound can be at most 3.453 and Li et al. improved the ratio into 3.4305 in [9]. Recently, Du and Du [10] showed that $mis(G) \leq 3.399mcds(G) + 4.874$, which is the best result up to now.

Although finding minimum CDS in UBG is very similar as in UDG, the analysis of those approximation ratios in UBG are much harder. Because, instead of disk packing, sphere packing has more complicated properties. To the best of our knowledge, few papers studied the approximation ratio for MCDS problem in UBG. In the earlier stage, Hansen and Schmutz [11] discussed the expected size of a CDS in a random UBG and compared the performance of existing algorithms. Later, Butenko and Ursulenko [12] proved that the ratio of $mis(G)/mcds(G)$ in UBG is at most 11 by using the well-known fact that a sphere can touch at most twelve spheres of the same size, which induced an approximation ratio of 22 for MCDS in UBG. Zhong et al. [13] claimed that such ratio could be reduced to 16. Zou et al. [14] further reduced this ratio to $13 + \ln 10$. Recently, Kim et al. [1] referred the idea in [7] and tried to answer how many independent points can be contained in two adjacent unit balls. Finally, they improved the ratio of $mis(G)/mcds(G)$ into 10.917 by showing that there are at most 22 independent points in two adjacent unit balls, and finally got an approximation ratio of MCDS as 14.937, which is the best result up to now.

However, after careful investigation, we find that during the deduction process in [1], one of the intermediate assertion is incorrect, which overthrows the final result completely. The main technique they implemented in their proof is a projection method to the ball surface and then applying some famous graph theories, and the problem comes under some scenarios when the projection result cannot guarantee the distance lower bound of two independent points. Researchers later found that designing a projection method to guarantee the distance lower bound is not an easy step, and it remains an open question in recent years [15]. As a consequence, in this paper we will first introduce a new projection method to guarantee the distance bound, and then illustrate the bound of $mis(G)/mcds(G)$ with some new analyzing techniques. Since the mistake in [1] only influences the selection of MIS, we will only focus on the first phase of MCDS construction. Next, we will further optimize the algorithms for MCDS selection in [1] with prune process and validate the efficiency of our design by numerical experiments.

The rest of the paper is organized as follows: Sect. 2 illustrates the problem in [1] with counter examples. Section 3 introduces our new projection method to analyze the ratio of $mis(G)/mcds(G)$. Section 4 discusses how to improve MCDS algorithm while Sect. 5 exhibits the simulation results with different parameter settings. Finally, Sect. 6 summarizes this paper.

2 Independent Points in Two Adjacent Unit Balls

Similar with the analysis in [7], once we have the answer of *"two-ball problem"*, we can deduce a better upper bound for the ratio $mis(G)/mcds(G)$ and reduce the overall approximation ratio.. *two-ball problem* means the problem of "how many independent points can be contained in two adjacent unit balls". Here two adjacent unit ball means the Euclidean distance between two balls with unit radius is at most 1, while any two points are called independent points if and only if their Euclidean distance is at least 1.

Actually, what Kim et al. did in [1] follows this idea. However, their method have an unavoidable error. In Subsect. 2.1 we will review their method to prove two-ball problem, and then in Subsect. 2.2 we will precisely point out where their problem lies and provide an counter example to validate our claim.

2.1 Review Kim's Method in [1]

In [1], Kim et al. referred the idea in [7] and improved the ratio of $mis(G)/mcds(G)$ into 10.917 by showing that there are at most 22 independent points in two adjacent unit balls. Their answer to the two-ball problem is the most important contribution in their paper. In order to solve the two-ball problem, Kim et al. extended the approach for solving the famous Gregory-Newton problem [16]. They considered two adjacent unit balls, say, B_1 and B_2 with centers u_1 and u_2. To get an upper bound of MIS in these two adjacent balls, they assumed that the Euclidean distance between u_1 and u_2 is equal to 1, since the total volume of $B_1 \cap B_2$ is larger when the distance between these two adjacent nodes increases, and consequently more independent nodes can be contained in $B_1 \cap B_2$. They then divided all the independent nodes into two categories: the nodes located in $(B_1 \cup B_2) \backslash (B_1 \cap B_2)$ and the nodes in $B_1 \cap B_2$. They mainly focused on the former part and claimed that the size of MIS in this region is at most 20 with a special "projection" method.

Their projection method is a mapping rule to project all the independent nodes in $(B_1 \cup B_2) \backslash (B_1 \cap B_2)$ to the surface of $B_1 \cup B_2$. The detailed description can be shown as follows: For each independent node v, if $v \in B_1 \backslash B_2$ (respectively, $B_2 \backslash B_1$), then draw a radial from u_1 (respectively, u_2) going through v, and intersect the outer surface of B_1 (denoted by $Sur(B_1)$, and respectively $Sur(B_2)$) at point P. By this mapping rule, they got the projection points set $\{P_1, P_2, \cdots, P_t\}$, where t is the size of MIS in $(B_1 \cup B_2) \backslash (B_1 \cap B_2)$.

Next, for any two points P_i and P_j, if their Euclidean distance (denoted by $d(P_i, P_j)$) is between 1 and $3 \arccos(1/7)\pi$, they made a curve from P_i to P_j on $Sur(B_1 \cup B_2)$ in the specified way as shown in Sect. 3.2.1 in [1]. These curves partition $Sur(B_1 \cup B_2)$ into some tiny faces. By analyzing the lower bounds of those faces' areas and using Euler's formula, they proved that $t \leq 20$. Combined with the fact that a unit ball can pack at most 12 independent nodes [16], they finally concluded that the number of MIS in the union of two adjacent unit balls is at most 22.

2.2 The Problem of Kim's Method with Counter Examples

After careful investigation, we find that in Sect. 3.2 of [1], one of the intermediate assertion is incorrect, which overthrows the final result completely. This assertion says: **"According to their mapping rule, on $\mathbf{Sur(B_1 \cup B_2)}$, for any $\mathbf{P_i}$ and $\mathbf{P_j}$, $\mathbf{d(P_i, P_j) > 1}$."** This assertion is a foundation of their work. With this assertion, they could conclude that no two curves on $Sur(B_1 \cup B_2)$ can intersect, which is a declaration to guarantee the correctness of the lower bound for the tiny faces' areas on $Sur(B_1 \cup B_2)$.

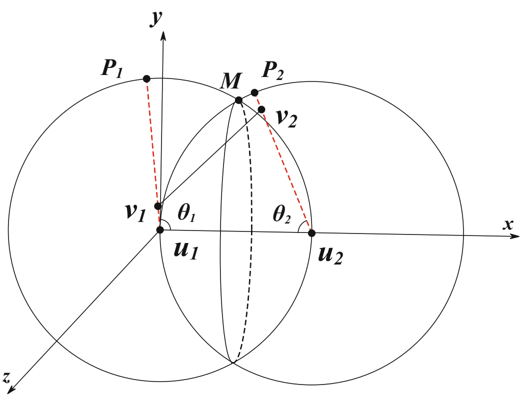

Fig. 1. An Example to show that under some cases $d(P_1, P_2) < 1$

However, although in many cases this assertion seems correct, it is not valid for every possible scenario. Let us provide an example to show why it is incorrect. In this example, v_1 and v_2 are two independent nodes located in two different balls and their projection points are P_1 and P_2 respectively, as Fig. 1 shows. Specially, we can let v_1, v_2, u_1 and u_2 locate in a same plane. Moreover, we set $\theta_1 + \theta_2 \le \pi$, where θ_1, θ_2 denote $\angle v_1 u_1 u_2$ and $\angle v_2 u_2 u_1$ respectively.

Now we prove that $d(P_1, P_2) \le 1$ in this situation. Since all points in this case are on a same plane, we can convert this case as a 2D plane in Fig. 2. Consider $\Delta P_1 M P_2$ in Fig. 2, by the Law of Cosines, we have

$$d^2(P_1, P_2) = d^2(P_1, M) + d^2(M, P_2) - 2d(P_1, M)d(M, P_2)\cos\angle P_1 M P_2.$$

From Fig. 2, we find that $\angle P_1 M P_2 = \angle T_1 M T_2 + \angle T_1 M P_1 + \angle T_2 M P_2$, where $\angle u_1 M T_1 = \angle u_2 M T_2 = \pi/2$. Here $T_1 M$ and $T_2 M$ are tangent lines to $disk(u_1)$ and $disk(u_2)$ respectively ($disk(u)$ is the cycle centered at u with radius 1). According to Alternate Segment Theorem,

$$\angle T_1 M P_1 = \angle P_1 u_1 M/2, \ \angle T_2 M P_2 = \angle P_2 u_2 M/2.$$

Therefore, $\angle P_1 M P_2 = (\theta_1 + \theta_2)/2 + \pi/3$. Also, it is easy to get $\angle P_1 u_1 M = \theta_1 - \pi/3$. Then,

$$d(P_1, M) = 2\sin(\theta_1/2 - \pi/6), \ d(P_2, M) = 2\sin(\theta_2/2 - \pi/6).$$

Hence, the Euclidean distance between P_1 and P_2 is:

$$d^2(P_1, P_2) = 4\cos^2(\frac{\theta_1 + \theta_2}{2}) - 4\cos(\frac{\theta_1 + \theta_2}{2})\cos(\frac{\theta_1 - \theta_2}{2}) + 1.$$

Since $\theta_1 + \theta_2 \le \pi$ in this case, $0 \le \cos(\frac{\theta_1 + \theta_2}{2}) \le \cos(\frac{\theta_1 - \theta_2}{2})$. Therefore, $d^2(P_1, P_2) \le 1$, which is a counter example for Kim's assertion.

Actually, we can also get a lower bound for $d(P_1, P_2)$ when v_1 and v_2 move to points u_1 and M respectively. In that case, $d(P_1, P_2)$ is equal to $2\sin(\pi/12) \approx 0.5176$, which is far less than 1.

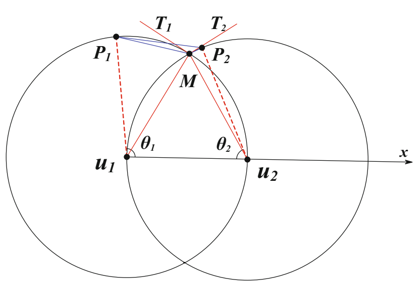

Fig. 2. An counter example in 2D

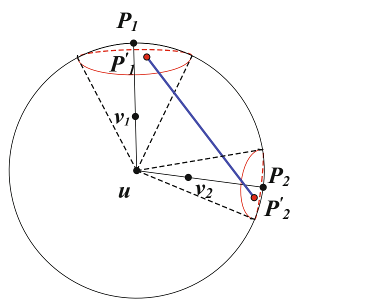

Fig. 3. The new projection of P'_1 and P'_2

3 Our New Projection Method

To achieve an approximation ratio of 10.917, we have to guarantee $d(P_1', P_2') \geq 1$ for any pair of independent points on the surface of two adjacent unit balls. In this section we introduce a new projection method to overcome Kim's flaw. The main idea of our method is an adaptive jitter scheme in the projection process. We will first introduce the motivation of our new projection rule. Then, we will give the definition of the new projection and prove $d(P_1', P_2') \geq 1$ holds under this projection.

Before introducing our new method, we need to introduce some notations and definitions, which are frequently used in the rest of this section. For any two nodes P_1 and P_2 in a three dimensional space, $d(P_1, P_2)$ denotes the Euclidean distance between them, while $|P_1P_2|$ denotes the length of geodesic arc between P_1 and P_2. $disk(v)$ denotes the unit disk with center 1. Let $Sur(B)$ denote the outer surface of a geometric object B. In order to apply geometric principle to solve our problem, we allow the distance between a pair of independent nodes to be equal to one. (Actually, we will often use this critical distance in the coming part of analysis.) Let B_1 and B_2 be two adjacent unit balls with centers u_1 and u_2, and $d(u_1, u_2) = 1$. For any vertex v, the plane going through point v, u_1 and u_2 is called the principal plane of v. In addition to "principal plane", we also define a plane called "normal plane", which is the perpendicular bisector of segment u_1u_2, and denote by L the intersection of the normal plane with $Sur(B_1 \cup B_2)$. Besides, we call the projection in Kim's method "direct projection".

According to our observation, when $d(P_1, P_2) < 1$ occurs, at least one of v_1 and v_2 is much closer to its dominator (u_1 or u_2), which can be found from Fig. 1. Without loss of generality, we say the closer node is v_1. In the unit ball B_1, we know the size of its MIS is at most 12. But if we want to put all the 12 independent nodes in B_1, the efficient way is to put all of them on $Sur(B_1)$. Consequently, if v_1 is much closer to u_1, it will greatly affect the total number of MIS in B_1, and the size of MIS in $B_1 \cup B_2$ will also be affected. We consider this MIS number decrease as the sacrifice to shorten $d(P_1, P_2)$. In order to quantitatively describe and use this property, we provide Lemma 1 as follows.

Lemma 1. *A unit ball B with center u contains two independent points v_1, v_2 and $d(u, v_1) = r_1$, $d(u, v_2) = r_2$. Their direct projection points are P_1 and P_2. P_1' (respectively, P_2') is an arbitrary point in the cycle region on $Sur(B_1)$ (respectively, $Sur(B_2)$) with center P_1 (respectively, P_2) and spherical radius* $\arccos(r_1/2) - \pi/3$ *(respectively,* $\arccos(r_2/2) - \pi/3$*) as Fig. 3 shows. Then $d(P_1', P_2') \geq 1$.*

Proof. First, we consider the 2D situation as Figs. 4 and 5 show. In Fig. 4, point v_1 is in $disk(u)$. M_1, M_2 are intersection points of uv_1' perpendicular bisector with $disk(u)$. Since all independent nodes with v_1 are outside $disk(v_1)$, nodes in $disk(u)$ which are independent with v_1 cannot locate above line M_1M_2. From Lemma 1, it is easy to know the available region of P_1' on $disk(u)$ is from P_1^l to P_1^r, where $\left|P_1^lP_1\right| = |P_1P_1^r| = \arccos(r_1/2) - \pi/3$ (shown in Fig. 4). Further,

$\angle M_2uv_1 = \arccos(r_1/2)$. Hence, $|P_1^r M_2| = \pi/3$ and $d(P_1^r, M_2) = 1$. Similarly, $d(P_1^l, M_1) = 1$. Based on the location of v_2, there are two situations to discuss.

Case 1: If v_2 is on the circle of $disk(u)$, then P_2, P_2' and v_2 are the same point. Since v_2 must locate below M_1M_2 and P_1' is on the arc between P_1^l and P_1^r, it is obvious to conclude that $d(P_1'P_2') \geq 1$.

Case 2: When v_2 is not on the circle of $disk(u)$ (as Fig. 5 shows), P_1' (P_2', respectively) locates between P_1^l and P_1^r (P_2^l and P_2^r, respectively). Next, we will prove that $d(P_1', P_2')$ is minimum when P_1' is at point P_1^r and P_2' is at point P_2^l.

As Fig. 5 shows, segment uM is the midperpendicular of $P_1^rP_2^l$. And lines $P_1^rT_1$ and $P_2^lT_2$ are parallel to uM. Then, all points in the arc from P_1^l to P_1^r and the arc from P_2^l to P_2^r are outside the parallel lines $P_1^rT_1$ and $P_2^lT_2$. Therefore, $d(P_1', P_2') \geq d(P_1^rP_2^l)$. On the other side, $d(P_1^rP_2^l) \geq 1$ is equivalent to $\left|P_1^rP_2^l\right| \geq \pi/3$. Combining with Law of Cosines, we have

$$\begin{aligned}\left|P_1^rP_2^l\right| &= \angle P_1^rP_2^l = \angle P_1uP_2 - \angle P_1uP_1^r - \angle P_2uP_2^l \\ &= \arccos\left(\frac{r_1^2 + r_2^2 - 1}{2r_1r_2}\right) - \left[\arccos\left(\frac{r_1}{2}\right) - \pi/3\right] - \left[\arccos\left(\frac{r_2}{2}\right) - \pi/3\right].\end{aligned}$$

When r_1 is given, it can be proved that the value of $\left|P_1^rP_2^l\right|$ decreases with the value of r_2 increases. Thus, when r_2 equals one, $\left|P_1^rP_2^l\right|$ will be minimized, which is exactly **Case 1** where v_2 is on the circle of $disk_1(u)$. Therefore, we have $d(P_1', P_2') \geq 1$.

Similarly, it is easy to extend the 2-dimensional situation to 3-dimensional situation. □

According to Lemma 1, we come up with a new region called "Effective Projection Region" to describe the extra feasible moving space of P_1' or P_2'.

Definition 1 (Effective Projection Region). *For node v in a unit ball B, its direct projection point is P. The region on $Sur(B)$ whose center is point P and spherical radius is $\arccos(r/2) - \pi/3$ is called v's effective projection region, where r is the Euclidean distance between v and B's center.*

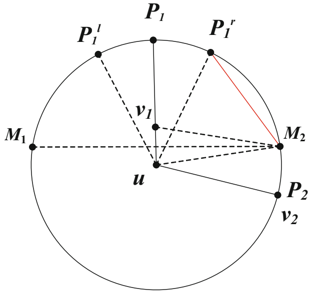

Fig. 4. Projection region with $r_2 = 1$

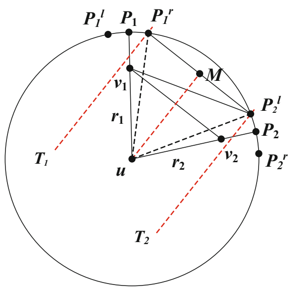

Fig. 5. Projection region with $r_2 < 1$

Obviously, in 3D space, for any two independent nodes v_1 and v_2 in a unit ball B, their direct projection points are P_1 and P_2. According to Lemma 1, if we arbitrarily move P_1 and P_2 along $Sur(B)$ inside their effective projection regions, $d(P_1, P_2)$ will always ≥ 1. Next, we will discuss the situation where v_1 and v_2 are in two different balls. Before that, we first define our new projection rule.

Definition 2 (Region Projection). *For any independent node v, its direct projection point is P, then any point which locates in v's effective projection region is a Region Projection point of v.*

Definition 3 (Final Projection). *For any point v in $(B_1 \cup B_2) \backslash (B_1 \cap B_2)$, the v's principal plane intersects with L and we assume M is the closer intersection point to v. P' is v's final projection if and only if it satisfies conditions as follows:*

(1) P' is a region projection point of v.
(2) P' is on v's principal plane.
(3) Among all the points satisfying (1) and (2), P' is the farthest from M.

Theorem 1. *For any two independent nodes v_1 and v_2 in $(B_1 \cup B_2) \backslash (B_1 \cap B_2)$, their final projection points are P_1' and P_2'. Then, $d(P_1', P_2') \geq 1$.*

Next, we will prove the correctness of Theorem 1. To make it simple, we first discuss the two-dimensional situation as a special case in Sect. 3.1. Afterwards, we generalize our conclusion for three-dimensional situation in Sect. 3.2.

By Lemma 1, if v_1 and v_2 are in the same unit ball, $d(P_1', P_2') \geq 1$. Thus, we only need to consider the situation when v_1, v_2 are in different balls. Without loss of generality, let v_1 in B_1 and v_2 in B_2.

3.1 Proof of Theorem 1 in 2-Dimensional Space

When u_1v_1 and u_2v_2 are in the same principal plane, our problem turns into a 2D problem as shown in Fig. 6.

In Fig. 6, P_1 and P_2 are the direct projection points; θ_1 and θ_2 denote $\angle P_1u_1u_2$ and $\angle P_2u_2u_1$; r_1 and r_2 denote $d(v_1, u_1)$ and $d(v_2, u_2)$ respectively. In addition, let α, β and γ denote $|P_1'M|$, $|P_2'M|$ and $\angle P_1'MP_2'$. To simplify our

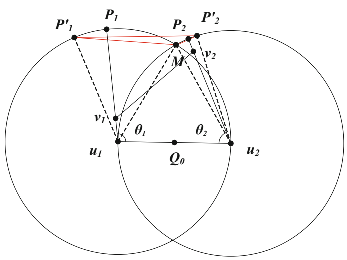

Fig. 6. Discussion in 2-dimension

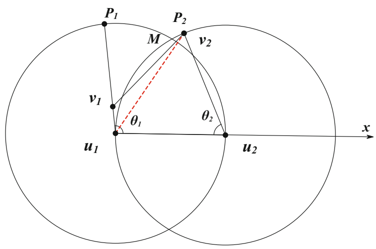

Fig. 7. Value of r_1 in 2D space

problem, we assume $d(v_1, v_2) = d(u_1, u_2) = 1$. In $\Delta P_1' M P_2'$, it is easy to figure out that $P_1'M = 2sin\frac{\alpha}{2}$, $P_2'M = 2sin\frac{\beta}{2}$, and $\gamma = \frac{2\pi}{3} + \frac{\alpha+\beta}{2}$. By cosine law, we have

$$\begin{aligned} d^2(P_1', P_2') &= \left(2sin\frac{\alpha}{2}\right)^2 + \left(2sin\frac{\beta}{2}\right)^2 - 2\left(2sin\frac{\alpha}{2}\right)\left(2sin\frac{\beta}{2}\right)cos\angle P_1'MP_2' \\ &= 4cos^2(\frac{\alpha+\beta}{2} + \frac{\pi}{3}) - 4cos(\frac{\alpha+\beta}{2} + \frac{\pi}{3})cos(\frac{\alpha-\beta}{2}) + 1. \end{aligned} \tag{1}$$

Let $x = \cos(\frac{\alpha+\beta}{2} + \frac{\pi}{3})$ and $y = \cos(\frac{\alpha-\beta}{2})$. Construct function $f(x, y) = 4x^2 - 4yx + 1$, where $y \geq 0$. Easy to see that $x \leq 0 \Rightarrow f(x, y) \geq 1$. That means $\alpha + \beta \geq \frac{\pi}{3}$. Moreover, $f(x, y) \geq 1 \Leftrightarrow x \leq 0 \quad or \quad x \geq y$. If $x \geq y$, then $\cos(\frac{\alpha+\beta}{2} + \frac{\pi}{3}) \geq \cos(\frac{\alpha-\beta}{2}) \geq 0$. Hence, $\alpha + \beta \leq \frac{\pi}{3}$ and $\frac{\alpha+\beta}{2} + \frac{\pi}{3} \leq \frac{\alpha-\beta}{2} \Rightarrow \beta + \frac{\pi}{3} \leq 0$, which is impossible. Therefore, $d(P_1', P_2') \geq 1$ is equivalent to $\alpha + \beta \geq \frac{\pi}{3}$.

According to Definition 3, we have $\alpha = \theta_1 + \arccos(r_1/2) - 2\pi/3$, and $\beta = \theta_1 + \arccos(r_1/2) - 2\pi/3$. Thus, our goal is to prove

$$\theta_1 + \theta_2 + \arccos(\frac{r_1}{2}) + \arccos(\frac{r_2}{2}) \geq \frac{5\pi}{3}. \tag{2}$$

As $\arccos(r_1/2) \geq \pi/3$ and $\arccos(r_2/2) \geq \pi/3$, we only need to consider the case where $\theta_1 + \theta_2 < \pi$.

Note that $d(v_1, v_2) = 1$ and $\theta_1 + \theta_2 < \pi$, v_1v_2 can not be parallel to u_1u_2. Without loss of generality, assume v_1 is more closer to u_1u_2 as Fig. 6 shows. Because $\theta_1 + \theta_2 < \pi$, it can be proved that $\angle v_1P_2u_2 < \pi/2$. Thus, $d(v_1, P_2) > 1$. If we move v_2 to P_2 and keep anything else unchanged, the new state will produce shorter $P_1'P_2'$. In that state, $r_2 = 1$.

Hence, it is sufficient to prove Eq. (2) under condition $r_2 = 1$. Since $d(v_1, v_2) = 1$, we can figure out the relation between θ_1 and r_1, θ_2. As Fig. 7 shows, we build a polar coordinates on u_1. In $\Delta v_2u_1u_2$, it is easy to figure out that $d(v_2, u_1) = 2\sin(\theta_2/2)$. Then, the coordinates of v_1 and v_2 are (θ_1, r_1) and $(\pi/2 - \theta_2/2, 2\sin(\theta_2/2))$. By mathematic knowledge, we have

$$\theta_1 = \pi - \frac{\theta_2}{2} - \arcsin\left(\frac{r_1^2 - 1 + 4\sin^2\frac{\theta_2}{2}}{4r_1\sin\frac{\theta_2}{2}}\right). \tag{3}$$

With Eq. (3), the value of $|P_1'M|$ is a function of r_1 and θ_2, and we denote it as $\alpha(r_1, \theta_2)$. By analyzing the sign of $\partial\alpha(r_1, \theta_2)/\partial r_1$, it can be proved that, when θ_2 is fixed, $\alpha(r_1, \theta_2)$ is minimum when r_1 is minimum or maximal. Consequently, when θ_2 is given, the value of $d(P_1', P_2')$ is minimum when r_1 is minimum or maximal. Therefore, what we need to do next is to verify $d(P_1', P_2') \geq 1$ for the two situations where r_1 is minimum and maximum.

Case 1 (r_1 is minimal): For this case, we first need to figure out the minimal value of r_1 when θ_2 is fixed. Considering v_1 locates in $disk(u_1)\backslash disk(u_2)$, it is obvious that, when v_1 locates on the intersection of $disk(v_2)$ with $disk(u_2)$, r_1 reaches minimal. Then, $d(v_1, u_2) = 1$, $\angle v_1u_2v_2 = \pi/3$, $\theta_1 = 2\pi/3 - \theta_2/2$ and $r_1 = 2\cos\theta_1$. Consequently, $\theta_1 + \theta_2 + \arccos(r_1/2) + \arccos(r_2/2) = 5\pi/3$ which meets inequality (2). Hence, in this case, $d(P_1', P_2') \geq 1$.

Case 2 (r_1 is maximal): It is easy to figure out that the maximal value of r_1 is 1 when v_1 locates on the boundary of $disk(u_1)$. In this case, P_1' and v_1 are the same. $d(P_1', P_2') = d(v_1, v_2) = 1$ which also meets the requirement in Theorem 1.

3.2 Proof of Theorem 1 in 3-Dimensional Space

When u_1v_1 and u_2v_2 are in different principal planes, we can follow the ideas in 2-dimension situation and give the same result.

First we explore the equivalent condition for $d(P_1', P_2') \geq 1$. In Fig. 8, δ denotes the dihedral angle between those two principal planes, Γ_1 and Γ_2; α and β denote $|P_1'M_1|$ and $|P_2'M_2|$ respectively. Similar as analysis in Sect. 3.1, we have $d(P_1', M) = 2\sin\frac{\alpha}{2}$, and $d(P_2', M) = 2\sin\frac{\beta}{2}$. Using method of analytical geometry, we have

$$d^2(P_1', P_2') = 4\cos^2\left(\frac{\alpha+\beta}{2} + \frac{\pi}{3}\right) - 4\cos\left(\frac{\alpha+\beta}{2} + \frac{\pi}{3}\right)\cos\left(\frac{\alpha-\beta}{2}\right) + 1$$
$$+ \left[\cos(\alpha-\beta) - \cos\left(\alpha+\beta+\frac{2\pi}{3}\right)\right](1-\cos\delta). \qquad (4)$$

Note that, when $\delta = 0$, Eq. (4) turns to Eq. (1). Since positive δ also contributes to $d(P_1', P_2')$ from Eq. (4), then $d(P_1', P_2')$ will be definitely larger than 1 when $\theta_1 + \theta_2 > \pi$. Therefore, we still just need to consider the situation where $\theta_1 + \theta_2 < \pi$.

Let $x = \cos((\alpha+\beta)/2 + \pi/3)$, $y = \cos(\alpha-\beta)/2$, and $z = \cos\delta$. Thus,

$$d^2(P_1', P_2') = 2(1+z)x^2 - 4yx + 2(1-z)y^2 + 1.$$

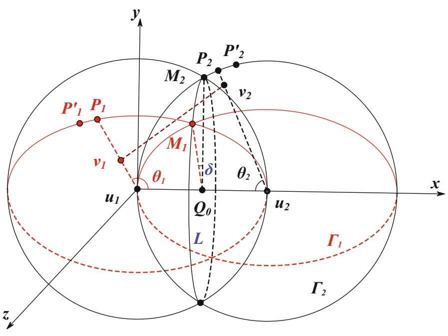

Fig. 8. Discussion in 3-dimensional space

Construct function $g(x,y) = 2(1+z)x^2 - 4yx + 2(1-z)y^2 + 1$. It is easy to find the solution of inequality $g(x,y) \geq 1$ and its effective solution is

$$\frac{x}{y} \leq \frac{1-z}{1+z}. \tag{5}$$

Hence, $d(P_1', P_2') \geq 1$ is equivalent to Eq. (5).

Similar with 2D situation we discussed in Sect. 3.1, we can also conclude that the value of $d(P_1', P_2')$ is smaller under condition $r_2 = 1$ and it is sufficient to prove Theorem 1 under this condition. Besides, when δ and θ_2 are fixed, the value of $d(P_1', P_2')$ is minimum when r_1 is minimum or maximum. Thus, we just need to verify Eq. (5) for these two situations.

Case 1 (r_1 is minimal): For this case, we first need to figure out the minimal value of r_1 when θ_2 and δ are fixed. Considering v_1 locates in $B_1 \backslash B_2$, it is obvious that, when v_1 locates on the intersection of $disk(u_2)$ on plane Γ_1 with the unit ball whose center locates on v_2, r_1 reaches minimal. In this state, we use θ_1^{min} to denote the current θ_1. Then, we have $r_1^{min} = 2\cos\theta_1^{min}$ and $2\sin 2\theta_1^{min} \sin\theta_2 \cos\delta - 2\cos 2\theta_1^{min} \cos\theta_2 = 1$. Hence, when θ_2 and δ are given, we can figure out r_1^{min} and θ_1^{min}.

Besides, in this situation the ranges of θ_2 and δ we need consider are as follows:

$$\begin{cases} \delta \in [0, \arccos\frac{1}{3}] \\ \theta_2 \in [\frac{\pi}{3}, 2\arctan(\frac{1-2\tan^2(\delta/2)}{\sqrt{3}}) + \frac{\pi}{3}]. \end{cases} \tag{6}$$

By numerical method, we can verify Eq. (5) under conditions (6).

Case 2 (r_1 is maximal): It is easy to figure out that the minimal value of r_1 is 1 when v_1 locates on $Sur(B_1)$. In this case, P_1' and v_1 are the same. $d(P_1', P_2') = d(v_1, v_2) = 1$ which meets the requirement in Theorem 1.

In conclusion, according to the analysis in Sect. 3.1 and Sect. 3.2, Theorem 1 always follows. Therefore, we can use our new projection rule to fix the incorrectness in [1].

4 MCDS Construction Improvement

So far, we have analyzed the approximation ratio of the MIS in UBG. In this section, we will introduce two prune methods to improve Kim's CDS construction algorithm. The following are some notations used in this section:

(1) For node u, $N(u) = \{v | v \in V(G) \backslash u \text{ and } d(u,v) \leq 1\}$, $N[u] = N(u) \cup \{x\}$.
(2) For node set C, $N(C) = (\cup_{v \in C} N(v)) \backslash C$.
(3) For u and C, $M_{u,C} = \{v | d(u,v) \leq 1,\ v \text{ is a MIS node and } v \notin C\}$.
(4) For u, $dom(u) = \{v | d(u,v) \leq 1 \text{ and } v \text{ is a CDS node}\}$.

4.1 Algorithm for Computing CDS

The algorithm introduced by Kim is formally described in Algorithm 1, which has two steps. It firstly generates an MIS M such that every node in M is two hops away from its nearest node in M. We can use Butenko and Ursulenko's algorithm to compute such MIS. The second step is to connect this MIS. Kim used a greedy strategy, which starts with the original node and repeats round by round. In each round, it picks a node v adjacent to the connected component C computed in the previous rounds that makes $|M_{v,C}|$ maximal, and add it to C, It terminates when all the points in MIS are connected.

4.2 Improve the Generated CDS

In this 2-step algorithm, there is some redundancy in the given CDS C. Firstly, through the computing of MIS, some nodes which have only one neighbor may be added to the MIS in order to maintain the properties of MIS. But when the connectors are added, those points would be useless for the whole CDS, and it is better to adjust it to non-CDS nodes. Also, the redundancy may occur in the inner side of the CDS due to the increased density of CDS nodes. Since the redundancy occurs after the algorithm terminates, we can add two more steps afterward to reduce CDS size with the help of prune techniques.

Algorithm 1. C-CDS-UBG($G(V, E)$)

```
Set M = Φ, B = Φ, V' = V.
Pick a root r ∈ V' that r has the biggest degree in V'.
Set M = {r}, B = N(r), V' = V'\N[r].
while V' ≠ Φ do
    Pick a node u ∈ N(B) such that |N(u) ∩ V'| is maximized.
    Set M = M ∪ {u}, B = B ∪ (M ∩ V'), and V' = V'\N(u).
end while
Set C = {r} and M' = M = {r}.
while M' ≠ Φ do
    Pick a node v ∈ N(C) such that |M_{v,C}| = max{|M_{u,c}|u ∈ N(C)}.
    Set C = C ∪ {v} ∪ M_{v,C} and M' = M'\M_{v,C}.
end while
Return C.
```

Notice that once remove a CDS node, the remaining CDS must maintain all its original properties. Thus for a CDS node $u \in G$, it could be removed iff:

(1) Every point that u dominates must have at least one alternative dominator.
(2) $G(C\backslash\{u\})$ is connected.

Correspondingly, we design two prune methods to reduce CDS size. The first method is to reduce some leaf nodes instantly. We use postorder traversal to traverse the CDS tree and reduce redundant points in it, as shown in Algorithm 2.

Lemma 2. *For any CDS C, After Algorithm 2 is executed, C is also a CDS.*

Algorithm 2. CUTLEAF(u)

1: Set $P = dom(u)$.
2: **while** $P \neq \Phi$ **do**
3: Pick a node $x \in P$.
4: **if** x has not been visited **then**
5: $CUTLEAF(x)$.
6: **end if**
7: Set $P = P \backslash \{x\}$.
8: **end while**
9: **if** ($|dom(v)| \geq 2$ for all $v \in N(u)$ in graph G) and $|dom(u)| = 1$ **then**
10: $C = C \backslash \{u\}$.
11: **end if**
12: Return C.

Proof. Let C' be the CDS after Algorithm 2 is executed. If $C' = C$, then Lemma 2 holds. Otherwise, let C_0 be the initial CDS, C_i be the CDS after the i-th reduction and u_i be the node reduced in this iteration. We show that if C_i is a CDS, then C_{i+1} is a CDS, for $i \geq 0$. According to Line 9, all $N(u_{i+1})$ has at least 2 dominators. Also, $|dom(u_{i+1}) = 1|$ ensures that only one CDS node is adjacent to u_{i+1}. So $C_{i+1} = C_i \backslash \{u_{i+1}\}$ is also connected. Hence, C_{i+1} is a CDS. Since C_0 is a CDS, recursively after Algorithm 2 is executed, C' is also a CDS.

Algorithm 3. CUTINSIDE($G(V, E)$,C)

1: Let G' be the subgraph generated by C, compute all the congestion nodes $C' \in C$.
2: Pick up a node $u \in V$ s.t. $u \notin C'$ and ($|dom(v)| \geq 2$ for all $v \in N(u)$ in G).
3: $C = C \backslash \{u\}$. Return.

Lemma 3. *For any CDS C, After Algorithm 3 is executed, C is also a CDS.*

Proof. Similar to Lemma 2, Line 2 ensures that all $N(u)$ has at least 2 dominators. Moreover, u is not a congestion node ensures that $C \backslash \{u\}$ is connected.

Algorithm 4. R-C-CDS-UBG($G(V, E)$)

1: P =C-CDS-UBG($G(V, E)$), Pick a root $r \in V'$ that r has the biggest degree in V'.
2: Run CUTLEAF(r).
3: Run CUTINSIDE($G(V, E)$,P) for several(30) times.
4: Return P.

With two algorithms above, Algorithm 4 is an improvement for computing CDS.

Theorem 2. *The time and space complexity of Algorithm 4 R-C-CDS-UBG is $O(n^2)$, where n is the number of nodes in a given input UBG.*

Proof. Since it is necessary to store the graph, the space complexity of Algorithm 4 is $O(n^2)$. Then we show that the time complexity is $O(n^2)$.

Firstly, the input time complexity for Algorithm 4 is $O(n)$. For the first step of Algorithm 1 (Lines 1–7), each round of the while loop add one point to the MIS, so the while loop ends in $O(n)$ rounds. In each round, a node x should be picked. Since we can store and update it instantly, the time complexity of node selection is $O(n)$ each round. Thus the time complexity of the whole loop is $O(n^2)$.

For the second step of Algorithm 1 (Lines 8–13), since each round of the while loop joint at least one MIS point to the CDS, and the MIS has $O(n)$ nodes, the loop ends in $O(n)$ rounds. During each round, we use an array to store $|M_{v,C}|$, and the maintenance time complexity is $O(n)$, since only points in $N(N(v))$ would change its $|M_{v,C}|$. Also, the time complexity to select a v is $O(n)$ each round. Hence, the time complexity of Algorithm 1 is $O(n^2)$.

For Algorithm 2, we can store $|dom(u)|$ for each $u \in G$. Once a CDS node v is reduced, only $dom(N(v))$ nodes need to change, so the maintenance complexity is $O(n)$. Next, each edge in G will be visited for a constant times, so the overall time complexity if $O(n^2)$. For Algorithm 3, we can use the Tarjan's strongly connected components algorithm for computing congestion set C', whose time complexity is $O(n^2)$. Hence, the time complexity of Algorithm 3 is $O(n^2)$.

For Algorithm 4, it runs Algorithm 1 once, Algorithm 2 once, and Algorithm 3 for a constant time, each with time $O(n^2)$. Therefore, the time complexity of Algorithm 4 is $O(n^2)$.

5 Simulation Results

In this section, we compare Algorithm 4 with Kim's Algorithm 1 to solve MCDS in UBGs. For the simulations, we deploy wireless nodes in a $20 \times 20 \times 20$ three-dimensional virtual space. We also ensure that the graph induced by all nodes is connected. The number of nodes varies from 100 to 1000 by increasing 100. We use 1 as the maximum transmission range of the nodes. Through the random graph generation process, we control the lower bound of distance between two nodes at 0.25, 0.5, 0.75. Thus, we can have graphs with different node density. In Fig. 9, we use R-C-CDS-UBG to identify our algorithm, and C-CDS-UBG to identify Kim's.

Figure 9 shows the comparison of the performance between two algorithms when the lower bound of distance between two points is 0.25, 0.5, and 0.75 respectively. Through the figure, we can see that whatever the graph is, our algorithm can give a better answer than Kim's averagely. the ratio between our answer and Kim's is nearly 0.78. Also, through the comparison, we can figure out that in most situations, our improvement is steady.

Figure 10 shows a sample of UBG which has 500 points. The first picture is the result by Kim's algorithm with 208 CDS points, while the second is the

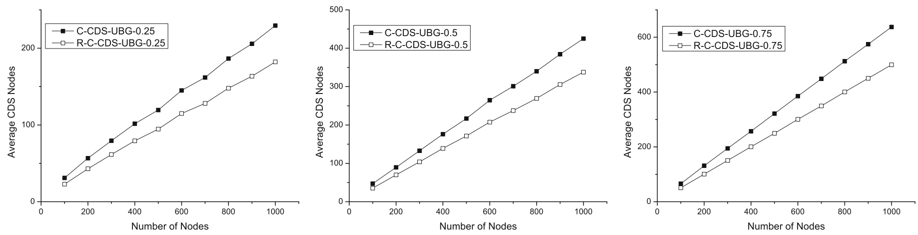

Fig. 9. Comparison with different parameter settings

(a) CDS by Kim's Alg. 1. |CDS|=208 (b) CDS by Prune Alg. 4. |CDS| = 155

Fig. 10. An example solution with $n = 500$ points.

result by prune algorithm with 155 CDS points. In this example our algorithm reduces CDS size by 25 %, and it can be found in the graph that lots of the boundary nodes are dropped to make the CDS much smaller.

6 Conclusion

In this paper, we first pointed out the problem in Kim's method [1]. Then, we proposed a new projection method for solving the *two-ball problem*. With this new projection method, we successfully improved the ratio of $mis(G)/mcds(G)$ in UBG into 10.917. Moreover, we also optimized the algorithm for minimum connected dominating set selection in [1] with prune process and validate the efficiency of our design by numerical experiments.

References

1. Kim, D., Zhang, Z., Li, X., Wang, W., Wu, W., Du, D.Z.: A better approximation algorithm for computing connected dominating sets in unit ball graphs. IEEE Trans. Mob. Comput. **9**(8), 1108–1118 (2010)
2. Kouider, M., Vestergaard, P.D., et al.: Generalized connected domination in graphs. Theoret. Comput. Sci. **8**(1), 57–64 (2006)
3. Gaspers, S., Liedloff, M., et al.: A branch-and-reduce algorithm for finding a minimum independent dominating set. Theoret. Comput. Sci. **14**(1), 29–42 (2012)

4. Wang, X., Li, P.: Improved data association method in binocular vision-SLAM. In: ICICTA, vol. 2, pp. 502–505. IEEE (2010)
5. Yang, D.: An immunity-based ant colony optimization topology control algorithm for 3D wireless sensor networks. Sens. Transducers J. **150**(3), 125–129 (2013)
6. Wan, P.J., Alzoubi, K.M., Frieder, O.: Distributed construction of connected dominating set in wireless ad hoc networks. In: INFOCOM, vol. 3, pp. 1597–1604. IEEE (2002)
7. Wu, W., Du, H., Jia, X., Li, Y., Huang, S.C.H.: Minimum connected dominating sets and maximal independent sets in unit disk graphs. Theoret. Comput. Sci. **352**(1), 1–7 (2006)
8. Gao, X., Wang, Y., Li, X., Wu, W.: Analysis on theoretical bounds for approximating dominating set problems. Discret. Math. Algorithms Appl. **1**(01), 71–84 (2009)
9. Li, M., Wan, P.J., Yao, F.: Tighter approximation bounds for minimum CDS in unit disk graphs. Algorithmica **61**(4), 1000–1021 (2011)
10. Du, Y.L., Du, H.W.: A new bound on maximum independent set and minimum connected dominating set in unit disk graphs. J. Combin. Optim. 1–7 (2013). doi:10.1007/s10878-013-9690-0
11. Hansen, J.C., Schmutz, E.: Comparison of two CDS algorithms on random unit ball graphs. In: ALENEX/ANALCO, pp. 206–211 (2005)
12. Butenko, S., Ursulenko, O.: On minimum connected dominating set problem in unit-ball graphs. Preprint Submitted to Elsevier Science (2007)
13. Zhong, X., Wang, J., Hu, N.: Connected dominating set in 3-dimensional space for ad hoc network. In: WCNC, pp. 3609–3612. IEEE (2007)
14. Zou, F., Li, X., Kim, D., Wu, W.: Construction of minimum connected dominating set in 3-dimensional wireless network. In: Li, Y., Huynh, D.T., Das, S.K., Du, D.-Z. (eds.) WASA 2008. LNCS, vol. 5258, pp. 134–140. Springer, Heidelberg (2008)
15. Du, D.-Z., Wan, P.-J.: Connected Dominating Set: Theory And Applications. Springer Optimization and Its Applications, vol. 77. Springer, Heidelberg (2012)
16. Zong, C.: Sphere Packings. Springer, New York (1999)