A greedy approximation algorithm for the minimum (2, 2)-connected dominating set problem

Yash P. Aneja
Odette School of Business
University of Windsor
Windsor, Canada

Asish Mukhopadhyay
School of Computer Science
University of Windsor
Windsor Canada

Md. Zamilur Rahman
School of Computer Science
University of Windsor
Windsor Canada

Abstract Using a connected dominating set (CDS) to serve as the virtual backbone of a wireless sensor network (WSN) is an effective way to save energy and reduce the impact of broadcasting storms. Since nodes may fail due to accidental damage or energy depletion, it is desirable that the virtual backbone is fault tolerant. This could be modeled as a k-connected, m-fold dominating set ((k,m)-CDS). Given a virtual undirected network $G = (V, E)$, a subset $C \subset V$ is a (k,m)-CDS of G if (i) $G[C]$, the subgraph of G induced by C, is k-connected, and (ii) each node in $V \backslash C$ has at least m neighbors in C. We present a two-phase greedy algorithm for computing a (2, 2)-CDS that achieves an asymptotic approximation factor of $(3 + \ln(\Delta + 2))$, where Δ is the maximum degree of G. This result improves on the previous best known performance factor of $(4 + \ln \Delta + 2 \ln(2 + \ln \Delta))$ for this problem.

1 Introduction

Suppose $G = (V, E)$ is a connected graph. A subset C of V is a said to be a connected dominating set (CDS) of G if $G[C]$, the induced graph on C, is connected and every vertex v in $V \setminus C$ is a neighbor of C (connected by an edge to some vertex $u \in C$). Nodes in C are called *dominators*, and nodes in $V \backslash C$ are called *dominatees*. To save energy and reduce interference, it is desirable that the CDS size is as small as possible. Computing a minimum CDS is a well known NP-hard problem [3]. By showing that finding a minimum set cover is a special case of finding a minimum CDS, Guha and Khullar [4] established that a minimum CDS can not be approximated within $\rho \ln n$ for any $0 < \rho < 1$ unless $NP \subset DTIME(N^{O(\log \log n)})$. In the same paper Guha and Khullar [4] proposed a two-phase greedy algorithm, with an approximation factor of $(3 + \ln \Delta)$ for fining a minimum sized CDS. Subsequently, Ruan et. al. [5] used a potential function approach to come up with a single phase greedy algorithm improving the approximation ratio to $(2 + \ln \Delta)$. There are, in the literature, several approximation algorithms for finding a minimum CDS for a general graph [2].

To make a virtual backbone more robust to deal with frequent node failures in WSNs, researchers have suggested using a (k, m)-CDS. As mentioned in the abstract, $C \subset V$ is a (k, m)-CDS if every node in $V \backslash C$ is adjacent to at least m nodes in C, and $G[C]$, the subgraph induced by C, is k-connected. The k-connectedness means that $|C| > k$ and $G[C \backslash X]$ is connected for any $X \subset C$ with $|X| < k$. In other words, no two vertices of $G[C]$ are separated by removal of fewer than k other vertices of C. With such a C, messages can be shared by the whole network, where every node in $V \backslash C$ can tolerate up to $m - 1$ faults (node failures) on its dominators, and the virtual backbone $G[C]$ can tolerate up to $k - 1$ faults.

Zhou et al. [7], using a more complex potential function than the one in Ruan et al. [5], provide a single phase $(2 + \ln(\Delta + m - 2))$-approximation algorithm for the minimum $(1, m)$-CDS problem in a general graph.

Shi et al. [6], using a two-phase approach, provide a $(\alpha + 2(1 + \ln \alpha))$-approximation algorithm for the minimum $(2, m)$-CDS, where $m \geq 2$ and α is the approximation ratio for the computation of a $(1, m)$-CDS. Using the solution obtained for the minimum $(1, m)$-CDS problem, they augment the connectivity of $G[C]$ by merging blocks (a block is defined as a maximal connected subgraph without a cut-vertex) of $G[C]$ recursively. When $m = 2$, this approximation ratio becomes $(4 + \ln \Delta + 2 \ln(2 + \ln \Delta))$.

In this paper, we present a different two-phase approach to the $(2, 2)$-CDS problem. The first phase ends up obtaining a C such that it is a *2-fold* dominating set, and all connected components of $G[C]$ are biconnected (2-connected). The second phase, at each iteration, needs two nodes from $V \backslash C$ to reduce the number of these biconnected components by at least one. This results in an algorithm with an asymptotic approximation factor of $(3 + \ln(\Delta + 2))$. By a simple modification of the potential function, our approach provides a $(3 + \ln(\Delta + m))$-approximation algorithm for computing a $(2, m)$-CDS.

For related and earlier work, the reader may refer to the papers by [6] and [7].

2   Main results

Let $G = (V, E)$ be a *biconnected* graph. For a $C \subset V$, define $p(C)$ to be the number of (connected) components of $G[C]$, the subgraph induced by C. Define $G\langle C \rangle$ to be the *spanning* subgraph of G, with vertex set V, and edge set $\{e \in E : e$ has at least one end in $C\}$. Let $q(C)$ represents the number of components of $G\langle C \rangle$. For each node $v \in V$, define $m_C(v)$ as:

$$m_C(v) = \begin{cases} 0, & \text{if } v \in C, \text{or adjacent to at least 2 nodes in } C \\ 1, & \text{if } v \in V \setminus C, \text{ and adjacent to at most 1 node in } C. \end{cases}$$

Let $m(C) = \sum_{v \in V} m_C(v)$. Thus $m(C)$ represents the number of nodes in $V \backslash C$ which have at most one neighbor in C. Note that for $m = 2$, $q(C)$ and $m(C)$ are defined exactly as in [7]. Again, as in [7], we assign a color to each node in V relative to a given C as follows. All nodes in C are colored black, nodes in $V \backslash C$ which have at least two neighbors in C are colored gray, nodes in $V \backslash C$ that have exactly one neighbor in C are colored red, and all other nodes are colored white.

Given C, we define $\hat{p}(C)$ to be

$$\begin{aligned} \hat{p}(C) &= \max_{x \in C} p(C \setminus \{x\}) \\ &= p(C \setminus \{x_C\}) \end{aligned} \tag{1}$$

in an attempt to capture the bi-connectivity deficit of $G[C]$.

A node x_C for which the maximum in (1) is attained is called a critical node of C. Note that if $G[C]$ is biconnected then $\hat{p}(C) = 1$, in which case every node in C can be viewed as a critical node.

Finally, we use the functions, $\hat{p}(C), q(C)$ and $m(C)$ to define a potential function, $f(C)$, on C as:

$$f(C) = \hat{p}(C) + q(C) + m(C) \tag{2}$$

and the difference function $\Delta_y f(C)$ by

$$\Delta_y f(C) = f(C) - f(C \cup \{y\}),$$

where $y \in V$. We can also, equivalently, write

$$\Delta_y f(C) = \Delta_y \hat{p}(C) + \Delta_y q(C) + \Delta_y m(C)$$

Result. Function $f(C)$ is monotonically non-increasing. That is, $\Delta_y f(C) \geq 0$ for every v in V. We need to consider three cases:

Proof: Several cases arise.

1. Suppose y is gray. This means that $\hat{p}(C \cup \{y\}) \leq \hat{p}(C)$. Clearly $m(C \cup \{y\}) \leq m(C)$, and $q(C \cup \{y\}) \leq q(C)$. Thus, $f(C \cup \{y\}) \leq f(C)$, and hence $\Delta_y f(C) \geq 0$.

2. Suppose y is red. It is then connected to only one node in C. As y is added to C, its m-value goes down by one and its q-value cannot increase. Its $\hat{p}$ value may increase by 1. Thus, $f(C \cup \{y\}) \leq f(C)$.

3. Suppose y is white. As y is added to C, its m-value goes down by one, q-value goes down by at least one, and $\hat{p}$-value goes up by one. Hence, $f(C \cup \{y\}) \leq f(C)$.

The following characterization of the structure of a biconnected graph [1] is useful for us.

Definition 2.1 *Given a graph H, we call a path P an H-path if P meets H exactly in its ends.*

For example consider a biconnected graph that is a cycle H of three nodes: x_1, x_2, and x_3. Then a path P of three nodes x_1, x_4, and x_3 is an H-path of H. Adding this H-path to cycle H, keeps it biconnected. The following proposition formalizes this observation and is illustrated in Fig. 1

Proposition 2.2 *[1] A graph H is biconnected if and only if it can be constructed from a cycle by successively adding H-paths to graphs H already constructed.*

Suppose C^* is a minimum $(2,2)$-CDS of G. Since it is biconnected, using the above proposition we can list the nodes in an order such that each sublist starting from the beginning is essentially a "*path*", where the first node of this "*path*" might correspond to a biconnected subgraph of C^*. Let us illustrate this with the following example 2 of $G[C^*]$.

We can list 8 nodes of this graph as the following list with sublists: $((1,2,3,4),5,6,7),8)$. Node 2 is adjacent to node 1, node 3 is adjacent to only node 1. Node 4, however, is adjacent to both nodes 2 and node 3. So $(1,2,3,4)$ corresponds to a biconnected graph (cycle), and is now designated as a "single meta-node" in our list. Next, the H-path $(4,5,6,7,2)$ is added to this subgraph, resulting in another biconnected subgraph. Finally, adding the H-path $(7,8,4)$ results in $G[C^*]$. The next lemma exploits this interpretation of a biconnected graph as a "path".
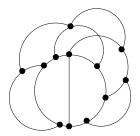
3

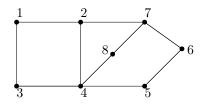Figure 1: *H-path construction of 2-connected graphs*

Figure 2:

Lemma 2.3 *For any two subsets $A, B \subseteq V$ and any node $y \in V$, if B is a "path" then*

$$\Delta_y f(A \cup B) \leq \Delta_y f(A) + 1 \tag{3}$$

Proof: The result is obvious if $y \in A$. Suppose $y \in B \backslash A$. Then the above result follows as $\Delta_y f(A) \geq$

0 for all $y \in V$. Thus we assume from here on that $y \notin A \cup B$. Define $\mu(f) = \Delta_y f(A \cup B) - \Delta_y f(A)$. It is useful to write $\mu(f)$ as:

$$\mu(f) = \mu(\hat{p}) + \mu(q) + \mu(m).$$

We first look at $\mu(m)$. Define S to be set of nodes which are neighbors of y which are white with respect to A and red with respect to B. Let $|S| = s$. We first want to show that:

$$\mu(m) = \begin{cases} s - 1, & \text{if } y \text{ is gray for } A \cup B, \ but \text{ not gray for } A \\ s, & \text{otherwise.} \end{cases} \tag{4}$$
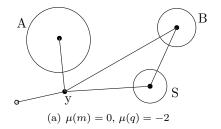
It is easy to formalize and establish this result by looking at the following two example figures: case (i): $\mu(m) = s - 1$, case (ii): $\mu(m) = s$.

In both 3(a) and 3(b) of Figure 3, $|S| = s = 1$. In Fig. 3(a), $m(A) = 4$, $m(A \cup \{y\}) = 3$. Hence $\Delta_y m(A) = 1$. $m(A \cup B) = 2$, $m(A \cup B \cup \{y\}) = 1$. Hence $\Delta_y m(A \cup B) = 1$. So $\mu(m) = \Delta_y m(A \cup B) - \Delta_y m(A) = 0$.

In Fig. 3(b), $m(A) = 4$, $m(A \cup \{y\}) = 3$. Hence $\Delta_y m(A) = 1$. $m(A \cup B) = 3$, $m(A \cup B \cup \{y\}) = 1$, $\Delta_y m(A \cup B) = 2$. Hence $\mu(m) = \Delta_y m(A \cup B) - \Delta_y m(A) = 1$.

We now look at $\mu(q)$. We want to show that:

$$\mu(q) \leq \begin{cases} -s, & \text{if } y \text{ is adjacent to } B \\ -(s - 1), & \text{otherwise.} \end{cases} \tag{5}$$

4

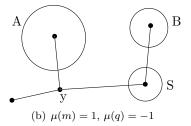(a) $\mu(m) = 0$, $\mu(q) = -2$ (b) $\mu(m) = 1$, $\mu(q) = -1$

Figure 3: Two cases covering the computation of $\mu(m)$

Let $N_A(y)$ be the set of components of $G\langle A\rangle$ that are adjacent with node y in G (the component of $G\langle A\rangle$ containing node y, if any, is not counted). Then $\Delta_y q(A) = |N_A(y)|$. Hence $\mu(q) = |N_{A\cup B}(y)| - |N_A(y)|$. Again, it is easy to formalize and establish the above result by looking at the above two example figures, in Figure 3, covering the cases: 3(a), $\mu(q) \leq -s$, and 3(b): $\mu(q) \leq -(s-1)$.

In Figure 3(a), $\Delta_y q(A) = |N_A(y)| = 3$, $\Delta_y q(A \cup B) = N_{A\cup B}(y) = 1$. Hence $\mu(q) = 1 - 3 = -2$.

In Figure 3(b), $\Delta_y q(A) = N_A(y) = 2$, $\Delta_y q(A \cup B) = |N_{A\cup B}(y)| = 1$. Hence $\mu(q) = -1$. Zhou et al. [7] have established the above two results in a more general setting.

Now we focus on $\mu(\hat{p}) = \Delta_y \hat{p}(A \cup B) - \Delta_y \hat{p}(A)$.

Let r be a critical node of $G[A]$. Let A_r be the set of nodes in the component of $G[A]$ containing node r. [Note that if $G[A]$ is connected then $A_r = A$.]

We define three constants α, β, and γ in $G[A]$ as follows. Let

$\alpha = p(A_r\backslash\{r\})$, the number of components in $G[A_r\backslash\{r\}]$.

$\beta = $ The number of components $G[A\backslash A_r]$ which are adjacent to node y in $G[A]$.

$\gamma = $ The number of components of $G[A_r\backslash\{r\}]$ which are are adjacent to node y in $G[A]$. ($\gamma \leq \alpha$).
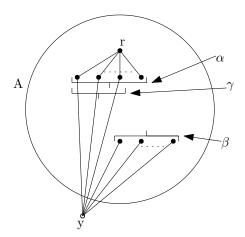
Refer to Fig. 4 for an illustration.

Figure 4: Illustrating the parameters α, β and γ

Result-1: $\Delta_y \hat{p}(A) = \min\{\alpha, \beta + \gamma\} - 1$.

Proof: Referring to the figure above, note that $\hat{p}(A) = \alpha + \beta$. Now let us calculate $\hat{p}(A \cup \{y\})$. Whichever of the two nodes, node r or node y, whose removal results in the higher number of components in $G[A \cup \{y\}]$ is the critical node . Now if we remove node r, the resulting number of components will be $(\alpha - \gamma) + 1$. If we remove node y then this number is $\beta + 1$. Hence

$$\hat{p}(A \cup \{y\}) = \max\{\alpha - \gamma, \beta\} + 1$$

Hence,

$$\begin{aligned}
\Delta_y \hat{p}(A) &= \hat{p}(A) - \hat{p}(A \cup \{y\}) \\
&= \alpha + \beta - [\max\{\alpha - \gamma, \beta\}] + 1 \\
&= \alpha + \beta + \min\{\gamma - \alpha, -\beta\} - 1 \\
&= \min\{\beta + \gamma, \alpha\} - 1
\end{aligned}$$

Returning to $\mu(\hat{p}) = \Delta_y \hat{p}(A \cup B) - \Delta_y \hat{p}(A)$, we use *result-1* to make some assertions about $\mu(\hat{p})$. As we mentioned earlier, we can assume that B is a set of nodes which form a "path". Since B is a "path", adding B to A does not create a new critical node in $G[A \cup B]$.

Result-2: Suppose y is not adjacent to B, then $\mu(\hat{p}) = 0$.

Proof: Since y is not adjacent to B, adding B to A does not change β and γ values. α value may increase. Hence, $\min\{\beta + \gamma, \alpha\}$ does not change, implying $\mu(\hat{p}) = 0$.

Result-3: If y is adjacent to B, then $\mu(\hat{p}) \leq 1$.

Proof: If B is not adjacent to r, then β goes up by 1, α and γ do not change. Hence $\mu \leq 1$. If B is adjacent to r, then both α and γ go up by 1, but β does not change. Hence $\mu(\hat{p}) = 1$. Hence we have the third inequality:

$$\mu(\hat{p}) \quad \begin{cases} = 0, & \text{if } y \text{ is not adjacent to } B, \\ \leq 1, & \text{otherwise.} \end{cases} \tag{6}$$

Combining the three inequalities (4), (5), and (6), proves our Lemma 2.3.

Lemma 2.4 *Let $G = (V, E)$ be a biconnected graph. Then, C is a 2-fold dominating set if $\Delta_y f(C) = 0$ for every $y \in V$.*

Proof: The following claims establish the proof.

Claim-1. $C \neq \emptyset$.

Suppose $C = \emptyset$. We have $\hat{p}(\emptyset) = 0$, $q(\emptyset) = |V|$, $m(\emptyset) = |V|$. Since G is biconnected, every node in G has degree at least 2. Pick any node y. So $C = \{y\}$, and $\hat{p}(\{y\}) = 0$, $q(\{y\}) = |V| - |N_G(y)|$, $m(\{y\}) = |V|$. Hence $\Delta_y f(C) > 0$, a contradiction.

Claim-2. $|C| \geq 3$. Its proof is straightforward.

Claim-3. $m(C) = 0$. This claim would imply that C is a 2-fold CDS.

Suppose $m(C) > 0$. This means that there is at least one node y which is red or white with respect to C. Suppose that y is a white node. This means that y is an isolated node in $G\langle C \rangle$, and hence accounts for one component in computing $q(C)$. Adding y to C implies $\Delta_y q(C) \geq 1$, and $\Delta_y m(C) = 1$. Since $\Delta_y \hat{p}(C) \geq -1$, we have $\Delta_y f(C) \geq 1$, a contradiction. So assume that there are no white nodes. Suppose y is red. This mean that y is adjacent to only one node in C. Since G is biconnected, y is adjacent to another node $y_1 \notin C$. So y_1 is either red or gray. Suppose y_1 is red.

6

Adding y to C makes y_1 gray. Hence $\Delta_y m(C) = 2$. Since $\Delta_y q(C) \geq 0$, and $\Delta_y \hat{p}(C) \geq -1$, we have $\Delta_y f(C) \geq 1$, a contradiction. So assume y_1 is gray. Then $\Delta_{y_1} \hat{p}(C) \geq 0$, $\Delta_{y_1} m(C) = 1$, since y is now gray in $G[C \cup \{y_1\}]$, $\Delta_{y_1} q(C) \geq 0$, implying $\Delta_{y_1} f(C) > 0$, a contradiction. This proves the claim.

Claim-4. Every (connected) component in $G[C]$ is biconnected.

To prove this, suppose C_1 is a component of C which is not biconnected. Hence C_1 has a critical vertex x such that $\hat{p}(C_1) = p(C_1 \setminus \{x\}) = t \geq 2$. Since G is biconnected, there exists a gray node y that is connected to two different components of $G[C_1 \setminus \{x\}]$. Hence $\hat{p}(C_1 \cup \{y\}) \leq t - 1$, implying $\Delta_y \hat{p}(C) \geq 1$, $\Delta_y q(C) \geq 0$, and hence $\Delta_y f(C) > 0$, a contradiction.

When $\Delta_y f(C) = 0$, $\forall\ y \in V \setminus C$, we say that phase-I of the algorithm has ended. A formal description of the Phase I algorithm is given below. At the end of Phase I, $G[C]$ has t biconnected components, $t \geq 1$. If $t = 1$, there is nothing more to do. Again, since G is biconnected, if C_1 and C_2 are any two components of $G[C]$, there must exist at least two nodes y_1 and y_2 in $V \setminus C$ such that both y_1 and y_2 are connected to both C_1 and C_2, making $G[C \cup \{y_1, y_2\}]$ having one less component than $G[C]$.

So, if at the end of phase-I, we have t components in $G[C]$, we need to add at most $2t$ nodes to C to obtain a $(2,2)$-CDS.

Algorithm 1 Greedy Algorithm for approximate (2,2)-MCDS : Phase 1

1: Set $C = \emptyset$
2: **while** $\Delta_y f(C) > 0$ **do**
3:     Pick a vertex $v \in V - C$ that causes the maximum reduction in $\Delta_y f(C)$
4:     Set $C = C \cup \{y\}$
5: **end while**
6: return C

Theorem 2.5 *The greedy algorithm with potential function f for $(2,2)$-CDS is bounded by the approximation ratio $(3 + \ln(\Delta + 2))$, where Δ is the maximum degree of G.*

Proof: Assume $|V| = n$. Let $C_G = \{x_1, x_2, \ldots x_g\}$, in the order of nodes selected by the algorithm (phase-I). For $0 \leq i \leq g$, let $C_i = \{x_1, \ldots, x_i\}$. In particular, C_g is the output of the algorithm. Suppose C^* is a minimum $(2,2)$-CDS with $\theta = |C^*|$. Since $G[C^*]$ is biconnected, we can arrange the elements C^* as $y_1, \ldots, y_\theta$ such that for each $j \geq 2$, $C_{j-1}^* = \{y_1, \ldots, y_{j-1}\}$ can be written as a "path", such that y_j is connected to y_{j-1}, and perhaps to the first node (or meta-node) of this path. If y_j is also connected to the first "node", then $G[y_1, \ldots, y_j]$ is biconnected, and considered as a single meta-node. Let $C_0 = C_0^* = \emptyset$. Since $f(C^*) = 2$, we have

$$f(C_{i-1}) - 2 = f(C_{i-1}) - f(C_{i-1} \cup C^*)$$

$$= \sum_{j=1}^{\theta} \Delta_{y_j}(C_{i-1} \cup C_{j-1}^*)$$

$$\leq \sum_{j=1}^{\theta} \left(\Delta_{y_j}(C_{i-1}) + 1 \right)$$

7

By the pigeonhole principle, there exists a node y_j in C^* *such that*

$$\Delta_{y_j} f(C_{i-1}) + 1 \geq \frac{f(C_{i-1}) - 2}{\theta}$$

Since phase-I follows greedy strategy,

$$\Delta_{x_i} f(C_{i-1}) \geq \Delta_y f(C_{i-1}) \geq \frac{f(C_{i-1}) - 2}{\theta} - 1$$

or

$$\text{or } f(C_i) \leq f(C_{i-1}) - \frac{f(C_{i-1}) - 2}{\theta} + 1$$

Denote $a_i = f(C_i) - 2$. Then, we can equivalently write

$$a_i \leq a_{i-1} - \frac{a_{i-1}}{\theta} + 1 \tag{7}$$

Since all a_i's are integers, we have

$$a_i \leq a_{i-1} - \left\lceil \frac{a_{i-1}}{\theta} \right\rceil + 1$$

Now $a_i > \theta$ implies $\left\lceil \frac{a_{i-1}}{\theta} \right\rceil \geq 2$, which means $a_i < a_{i-1}$. So long as $a_i > 2 + \theta$, phase-I continues. Now, we can write inequality 7 as:

$$a_i \leq a_{i-1} \left(1 - \frac{1}{\theta} \right) + 1, \text{ whose solution, as in [4, 5, 7], is}$$

$$a_i \leq a_0 \left(1 - \frac{1}{\theta} \right)^i + \sum_{j=0}^{i-1} \left(1 - \frac{1}{\theta} \right)^j$$

So after $\theta \ln(a_0/\theta)$ iteration, as in [4], $a_i < 2\theta$. Since phase-I continues as long as $a_i > \theta$, after at most θ iterations $a_i \leq \theta$ since each iteration of phase-I reduces a_i' by at least one unit. Suppose phase-I ends at this stage. At this stage $f(C_i) \leq \theta + 2$. Thus C has at most $\theta + 2$ biconnected components, and needs at most $2\theta + 4$ additional nodes in C to obtain a $(2, 2)$-CDS, resulting in a bound of

$$\theta \ln(a_0/\theta) + \theta + 2\theta + 4 = \theta \left[\ln \frac{a_0}{\theta} + 3 + \frac{4}{\theta} \right]$$

Asymptotically, $4/\theta$ can be ignored. So the asymptotic approximation factor is $3 + \ln(\frac{a_0}{\theta}) = 3 + \ln(\frac{2n}{\theta})$. To bound $\frac{2n}{\theta}$, we proceed as follows.

Taking $i = 1$, $C_0 = \emptyset$. Then $\hat{p}(\emptyset) = 0$, $q(\emptyset) = n$, $m(\emptyset) = n$. So $f(\emptyset) = 2n$. $f(x_1) = \hat{p}(\{x_1\}) + q(\{x_1\}) + m(\{x_1\})$, $\hat{p}(\{x_1\}) = 0$, $q(\{x_1\}) = n - |N_G(x_1)| - 1$, $m(\{x_1\}) = n - 1$. This implies that $f(x_1) = 2n - 2 - |N_G(x_1)| = 2n - 2 - \Delta$. Hence

$$\Delta_{x_1} f(\emptyset) = |N_G(x_1)| + 2 = \Delta + 2$$

8

Now

$$f(C_1) \leq f(C_0) - \frac{f(C_0) - 2}{\theta} + 1 \text{ or} \tag{8}$$

$$\frac{2n - 2}{\theta} \leq \Delta + 2 \text{ or} \tag{9}$$

$$\frac{2n}{\theta} \leq \Delta + 2 + \frac{2}{\theta} \tag{10}$$

So the approximation ratio asymptotically becomes $3 + \ln(\Delta + 2)$.

3   Conclusion

In this paper, we proposed a $(3 + \ln(\Delta + 2))$-approximation algorithm for the $(2, 2)$-connected dominating set for a general graph. This algorithm can easily be generalized for the $(2, m)$-CDS problem, for $m \geq 2$, resulting in a $(3 + \ln(\Delta + m))$-approximation algorithm.

References

[1] Reinhard Diestel. *Graph theory; 2nd ed.* Graduate texts in mathematicals. Springer, Heidelberg, 2000. Record from the Electronic Library of Mathematics/European Mathematical Society.

[2] Ding-Zhu Du, Ker-I Ko, and Xiaodong Hu. *Design and Analysis of Approximation Algorithms.* Springer Publishing Company, Incorporated, 2011.

[3] Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman & Co., New York, NY, USA, 1979.

[4] Sudipto Guha and Samir Khuller. Approximation algorithms for connected dominating sets. *Algorithmica*, 20(4):374–387, 1998.

[5] Lu Ruan, Hongwei Du, Xiaohua Jia, Weili Wu, Yingshu Li, and Ker-I Ko. A greedy approximation for minimum connected dominating sets. *Theor. Comput. Sci.*, 329(1-3):325–330, 2004.

[6] Yishuo Shi, Yaping Zhang, Zhao Zhang, and Weili Wu. A greedy algorithm for the minimum 2-connected m-fold dominating set problem. *J. Comb. Optim.*, 31(1):136–151, 2016.

[7] Jiao Zhou, Zhao Zhang, Weili Wu, and Kai Xing. A greedy algorithm for the fault-tolerant connected dominating set in a general graph. *J. Comb. Optim.*, 28(1):310–319, 2014.