証明書分散問題の近似可能性について

泉朋子 [1]，泉泰介 [2]，小野廣隆 [3]，和田幸一 [2]
[1] 名古屋工業大学産学官連携センター
[2] 名古屋工業大学大学院工学研究科
[3] 九州大学大学院システム情報科学研究院

証明書分散問題 (Minimum Certificate Dispersal Problem, MCD) とは，グラフ G と要求集合 R が与えられたときに，R に含まれるすべての要求を満たすよう各ノードに G の辺を割り当て，各ノードに割り当てられる辺の総数を最小化する問題である．要求とはグラフ G 上の異なる 2 つのノードの順序対で表され，要求 (u, v) を満たすにはノード u, v に割り当てる辺の和集合に G における u から v への経路が含まれる必要がある．MCD は与えられるグラフが強連結の場合においても NP-困難であることが既に示されている．本研究では，MCD の近似可能性について議論する．まず，強連結グラフにおいて MCD の近似率の下界が $\Omega(\log n)$ (n は G のノード数) であることを示し，さらに任意のグラフにおける MCD に対する多項式時間 $O(\log n)$-近似アルゴリズムが構成可能であることを示す．また，既存研究において多項式時間 2-近似アルゴリズムであると評価されていたアルゴリズムが，無向グラフを入力とする MCD に対しては多項式時間 3/2-近似アルゴリズムであることを示す．

On Approximability of the Minimum Certificate Dispersal Problem

Tomoko Izumi[1], Taisuke Izumi[2], Hirotaka Ono[3], Koich Wada[2]
[1]Center for Social Contribution and Collaboration, Nagoya Institute of Technology
[2]Graduate School of Engineering, Nagoya Institute of Technology
[3]Graduate School of Information Science and Electrical Engineering, Kyushu University

Assume that G is a graph and that R is a set of requests which is represented by a reachable ordered pair of nodes in G. The problem discussed in this paper requires us to assign edges to each node such that all requests in R are satisfied and the total number of edges all nodes have is minimized for a given G and R. To satisfy a request (u, v), a set of assigned edges to u and v must contain a path from u to v in G. This problem is called the *Minimum Certificate Dispersal problem (MCD)* and is NP-hard even if the input graph is restricted to a strongly connected one. In this paper, we consider approximability of MCD. We clarify an optimal approximability / inapproximability bound in terms of order: we prove the approximation ratio of MCD for strongly connected graphs is $\Omega(\log n)$ and MCD has a polynomial time approximation algorithm whose factor is $O(\log n)$ (n is the number of nodes in G). In addition, we prove that when a given graph is restricted to an undirected graph, the MCD algorithm proposed in [11] guarantees 3/2 approximation ratio.

1 Introduction

The problem discussed in this paper is, for a given directed graph $G = (V, E)$ and a set of *requests*, how to assign edges to each node such that all requests are satisfied and the total number of edges all nodes have is minimized. A request is represented by a reachable ordered pair of nodes in G. To satisfy a request (u, v), a set of assigned edges to u and v contains a path from u to v in G. This problem is formulated in [11] and called *the Minimum Certificate Dispersal problem (MCD)*. The given set R of requests is classified according to the elements of R: R is *subset-full* if there exists a subset V' of V such that R consists of all reachable pairs of nodes in V', and R is *full* if the subset V' is equal to V.

The problem is motivated by the requirement in public-key based security systems, which are known as a major technique for supporting secure communication in a distributed system [3, 4, 5, 6, 7, 10, 11]. The main problem of the systems is to make each user's public key available to others in such a way that its authenticity is verifiable. One of well-known approaches to solve this problem is based on public-key certificates. A public-key certificate contains public key of a user v encrypted by using private key of a user u. If a user u knows the public key of another user v, user u can issue a certificate from u to v. Any user who knows public key of u can use it to decrypt the certificate from u to v for obtaining public key of v. When a user w has communication request to send messages to a user v securely, w needs to know public key of v to encrypt the messages with it.

Table 1: Approximability / Inapproximability

Restriction on request		Arbitrary	Subset-full	Full
Hardness	[11]	(NP-Complete)	open	
	our paper	$\Omega(\log n)$		
Approximation ratio	[11]			2
	our paper	$O(\log n)$	1.5 (for undirected graphs)	

n is the number of nodes.

All certificates issued by the users in a network can be represented by a certificate graph: each node corresponds to a user and each directed edge corresponds to a certificate. For satisfying a communication request from a node w to v, node w needs to get node v's public-key. When the node w computes v's public-key, w uses a set of certificates stored in w and v in advance. Therefore, in a certificate graph, if a set of certificates stored in w and v contains a path from w to v, then the communication request from w to v is satisfied. In terms of cost to maintain certificates, the total number of certificates stored in all nodes must be minimized for satisfying all communication requests.

The minimum certificate dispersal with a restriction of available paths has discussed in [7]. That is, when a graph, a set of requests and a set of paths for each request are given, the problem is to assign the edges to each node such that all the requests are satisfied using the given paths and the total number of edges is minimized. They proved that the problem is NP-hard and proposed polynomial-time algorithms for the problem when a given graph is included in special graph classes. In their work, to assign edges to each node, only the restricted paths which are given for each request is allowed to be used. But in general case, there may exist several paths for each request in a graph.

MCD, with no restriction of available paths, is first formulated in [11]. In [11], it is proved that MCD is NP-hard even if the input graph is restricted to a strongly connected one. They proposed a polynomial-time 2-approximation algorithm MinPivot for strongly connected graphs when a set of request is full (see Table 1).

In this paper, we consider approximability of MCD. Table 1 shows our contribution in this paper. First, we clarify an optimal approximability / inapproximability bound for MCD in terms of order: we prove the lower bound of approximation ratio for MCD is $\Omega(\log n)$ by a reduction of the SET-COVER to MCD, where n is the number of nodes. This result provides a stronger inapproximability of MCD than the known result. Moreover, we show MCD has a polynomial time approximation algorithm whose factor is $O(\log n)$. The $O(\log n)$ approximation ratio is achieved by formulating MCD as a submodular set cover problem. In addition, we prove that when a given graph is restricted to an undirected graph, the algorithm MinPivot proposed in [11] guarantees $3/2$ approximation ratio. This approximation ratio is hold even if a given set of request is subset-full.

This paper is organized as follows. In Section 2, we define the Minimum Certificate Dispersal Problem (MCD). Section 3 and Section 4 present inapproximability and approximability of MCD respectively. In Section 5, we prove 3/2-approximation rate is achieved the algorithm MinPivot when a given graph is an undirected graph. Section 6 concludes the paper.

2 Minimum Certificate Dispersal Problem

In this section, we introduce several notations and define the minimum certificate dispersal problem (MCD).

Let $G = (V, E)$ be a directed graph, where V and E are the sets of nodes and edges in G respectively. An edge in E connects two distinct nodes in V. The edge from a node u to v is denoted by (u, v). The number of nodes and edges in G is denoted by n and m (i.e., $n = |V|, m = |E|$). A sequence of edges $p(v_0, v_k) = (v_0, v_1), (v_1, v_2), \ldots, (v_{k-1}, v_k)$ is called a *path* from v_0 to v_k of length k. For a path $p(v_0, v_k)$, v_0 and v_k are called the source and destination of the path respectively. The length of a path $p(v_0, v_k)$ is denoted by $|p(v_0, v_k)|$. For simplicity, we treat a path as the set of edges on the path when no confusion occurs. A shortest path $sp(u, v)$ from u to v is the one whose length is the minimum of all paths from

u to v. When there is more than one path with the minimum length from u to v, $sp(u, v)$ is defined as one of them chosen arbitrarily. The distance from u to v is the length of a shortest path from u to v, denoted by $d(u, v)$.

A *dispersal* D of a directed graph $G = (V, E)$ is a family of sets of edges indexed by V, that is, $D = \{D_v \subseteq E | v \in V\}$. We call D_v a local dispersal of v. A local dispersal D_v indicates the set of edges assigned to v. The *cost* of a dispersal D, denoted by $c.D$, is the sum of cardinalities of all local dispersals in D (i.e., $c.D = \Sigma_{v \in V} |D_v|$). A request is a reachable ordered pair of nodes in G. For a request (u, v), u and v are called the source and destination of the request respectively. A set R of requests is *subset-full* if there exists a subset of V such that R consists of all reachable pairs of nodes in V' ($i.e., R = \{(u, v) | u$ is reachable to v in G, $u, v \in V' \subseteq V\}$), and R is *full* if the subset V' is equal to V. We say a dispersal D of G *satisfies* a set R of requests if a path from u to v is included in $D_u \cup D_v$ for any request $(u, v) \in R$.

The *Minimum Certificate Dispersal Problem (MCD)* for a directed graph is defined as follows:

Definition 2.1
[Minimum Certificate Dispersal Problem (MCD)]
INPUT: A directed graph $G = (V, E)$ and a set R of requests
OUTPUT: A dispersal D of G satisfying R with minimum cost.

The minimum cost of a dispersal of G which satisfies R is called the *minimum dispersal cost* of G for R, and denoted by $c_{min}(G, R)$. For short, the cost $c_{min}(G, R)$ is also denoted by $c_{min}(G)$ when R is full. Let D^{Opt} be an optimal dispersal of G which satisfies R (i.e., D^{Opt} is one such that $c.D^{Opt} = c_{min}(G, R)$).

In this paper, we deal with MCD for undirected graphs in Section 5. For an undirected graph G, the edge between nodes u and v is denoted by (u, v) or (v, u). When an edge (u, v) is included in a local dispersal D_v, the node v has two paths from u to v and from v to u.

3 Inapproximability

It was shown in [11] that MCD for strongly connected graphs is NP-hard by a reduction from the VERTEX-COVER problem. In this section, we provide another proof of NP-hardness of MCD for strongly connected graphs, which implies a stronger inapproximability. Here, we show a reduction from the SET-COVER problem. For a collection $\mathcal{C}$ of subsets of a finite universal set U, $\mathcal{C}' \subseteq \mathcal{C}$ is called a *set cover* of U if every element in U belongs to at least one member of $\mathcal{C}'$. Given $\mathcal{C}$ and a positive integer k, SET COVER is the problem of deciding whether a set cover $\mathcal{C}' \subseteq \mathcal{C}$ of U with $|\mathcal{C}'| \leq k$ exists.

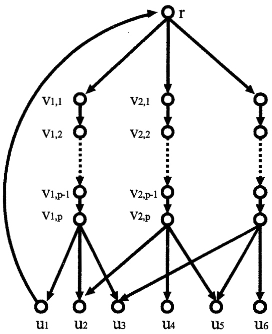


Figure 1: Reduction from SET-COVER

The reduction from SET-COVER to MCD is as follows: Given a universal set $U = \{1, 2, \ldots, n\}$ and its subsets $S_1, S_2, \ldots, S_m$ and a positive integer k as an instance $\mathcal{I}$ of SET-COVER, we construct a graph $G_\mathcal{I}$ including gadgets that mimic (a) elements, (b) subsets, and a special gadget: (a) Each element i of the universe set $U = \{1, 2, \ldots, n\}$, we prepare an element gadget u_i (it is just a vertex); let V_U be the set of element vertices, i.e., $V_U = \{u_i \mid i \in U\}$. (b) Each subset $S_j \in \mathcal{C}$, we prepare a directed path $(v_{j,1}, v_{j,2}, \ldots, v_{j,p})$ of length $p - 1$, where p is a positive integer used as a parameter. The end vertex $v_{j,p}$ is connected to the element gadgets that correspond to elements belonging to S_j. For example, if $S_1 = \{2, 4, 5\}$, we have directed edges $(v_{1,p}, u_2)$, $(v_{1,p}, u_4)$ and $(v_{1,p}, u_5)$. (c) The special gadget just consists of a base vertex r. This r has directed edges to all $v_{j,1}$'s of $i = 1, 2, \ldots, m$. Also r has an incoming edge from each u_i. See Figure 1 as an example of the reduction, where $S_1 = \{1, 2, 3\}, S_2 = \{2, 4, 5\}$ and $S_3 = \{3, 5, 6\}$. We can see that $G_\mathcal{I}$ is strongly connected. The set R of requests contains the requests from the base vertex r to all element vertices u_i, i.e., $R = \{(r, u_i) \mid u_i \in V_U\}$.

We can show the following lemma, although we omit the proof because it is straightforward.

Lemma 3.1
For the above construction of $G_\mathcal{I}$, the following holds:

(i) *If the answer of instance $\mathcal{I}$ of SET-COVER is* *yes, then $c_{min}(G,R) \leq pk+n$.*

(ii) *Otherwise, $c_{min}(G,R) \geq p(k+1)+n$.*

About the inapproximability of SET-COVER, it is known that SET-COVER has no polynomial-time approximation algorithm with factor better than $0.2267 \ln n$, unless $P = NP$ [1]. From this inapproximability, we rewrite Lemma 3.1 in terms of *gap-preserving reduction* [2] as follows:

Lemma 3.2
The above construction of $G_{\mathcal{I}}$ is a gap-preserving reduction from SET-COVER to MCD for strongly connected graphs such that

(i) *if $OPT_{SC}(\mathcal{I}) = \mathtt{min}$, then $c_{min}(G,R) \leq p \cdot \mathtt{min} + n$,*

(ii) *if $OPT_{SC}(\mathcal{I}) \geq \mathtt{min} \cdot c \ln n$, then $c_{min}(G,R) \geq (p \cdot \mathtt{min} + n)\left(c \ln n - \frac{cn \ln n - n}{p \cdot \mathtt{min} + n}\right)$,*

where $OPT_{SC}(\mathcal{I})$ denotes the optimal value of SET-COVER for $\mathcal{I}$ and $c = 0.2267$.

By taking p large enough, we have the following theorem:

Theorem 3.1
There exists no $(0.2267 \ln n - \varepsilon)$ factor approximation polynomial time algorithm of MCD for strongly connected graphs unless $P = NP$, where ε is an arbitrarily small positive constant.

It is not trivial (actually, it might be difficult) to extend the result to more restricted classes of strongly connected graphs, e.g., bidirectional graphs. However, we can still obtain some inapproximability result for bidirectional graphs, by slightly modifying the graph $G_{\mathcal{I}}$, though we omit the details.

Lemma 3.3
There is a gap-preserving reduction from VERTEX-COVER for graphs with degree at most 4 to MCD for bidirectional graphs such that

(i) *if $OPT_{VC}(\mathcal{I}) = \mathtt{min}$, then $c_{min}(G,R) \leq \mathtt{min} + n$,*

(ii) *if $OPT_{VC}(\mathcal{I}) \geq c \cdot \mathtt{min}$, then $c_{min}(G,R) \geq (\mathtt{min} + n)\left(c - \frac{(c-1)n}{\mathtt{min}+n}\right)$,*

where $OPT_{VC}(\mathcal{I})$ denotes the optimal value of VERTEX-COVER for $\mathcal{I}$, and $c = 79/78$.

In this lemma, $c = 79/78$ represents an inapproximability of VERTEX-COVER for graphs with degree at most 4 under the assumption $P \neq NP$ [8]. From this lemma, we obtain the following theorem:

Theorem 3.2
There exists no $(391/390 - \varepsilon)$ factor approximation polynomial time algorithm of MCD for bidirectional graphs unless $P = NP$, where ε is an arbitrarily small positive constant.

4 Approximability

In the previous section, we show that it is difficult to design a polynomial time approximation algorithm of MCD whose factor is better than $(0.2267 \ln n - \varepsilon)$, even if we restrict that the input graph is strongly connected. In this section, in contrast, we show that MCD has a polynomial time approximation algorithm whose factor is $O(\log n)$, which is applicable for general graphs. This implies that we clarify an optimal approximability / inapproximability bound in terms of order under the assumption $P \neq NP$.

The idea of $O(\log n)$-approximation algorithm is based on formulating MCD as a *submodular set cover problem* [9]: Let us consider a finite set N, a nonnegative cost function c_j associated with each element $j \in N$, and non-decreasing submodular function $f : 2^N \mapsto Z^+$. A function f is called *non-decreasing* if $f(S) \leq f(T)$ for $S \subseteq T \subseteq N$, and is called submodular if $f(S) + f(T) \geq f(S \cap T) + f(S \cup T)$ for $S, T \subseteq N$. For a subset $S \subseteq N$, the cost of S, say $c(S)$, is $\sum_{j \in S} c_j$.

By this f, c and N, the submodular set cover problem is formulated as follows: [**Minimum Submodular Set Cover (SSC)**]

$$\min \left\{ \sum_{j \in S} c_j : f(S) = f(N) \right\}.$$

It is known that the greedy algorithm of SSC has approximation ratio $H(\max_{j \in N} f(j))$ where $H(i)$ is the i-the harmonic number if f is integer-valued and $f(\emptyset) = 0$ [9]. Note that $H(i) < \ln i + 1$.

We here claim that our problem is considered a submodular set cover problem. Let $N = \bigcup_{u \in V} \{x_{e,u} \mid e \in E\}$. Intuitively, $x_{e,u} \in S \subseteq N$ represents that the local dispersal of u contains $e \in E$ in S, i.e., $e \in D_u$ in S. For $S \subseteq N$, we define $d_S(u,v)$ as the distance from u to v under the setting that each edge $e \in D_u \cup D_v$ of S has length 0 otherwise 1. That is, if all edges are included in $D_u \cup D_v$ of S, then $d_S(u,v) = 0$. If no edge is included in $D_u \cup D_v$ of S, then $d_S(u,v)$ is the length of a shortest path from u to v of G. Let $f(S) = \sum_{(u,v) \in R}(d_{\emptyset}(u,v) - d_S(u,v))$. This f is integer-valued and $f(\emptyset) = 0$. In the problem setting of MCD, we can assume that for any

$(u,v) \in R$, G has a (directed) path from u to v. (Otherwise, we have no solution). Then the condition $f(N) = f(S)$ means that all the requests are satisfied. Also cost c reflects the cost of MCD.

Then we have the following lemma:

Lemma 4.1
Function f defined as above is a non-decreasing submodular function.

Proof.
Since it is obvious that f is non-decreasing, we only show the submodularity of f. By the inductive property, it is sufficient to show that $f(S \cup \{x_{e,u}\}) + f(S \cup \{x_{e',v}\}) \geq f(S) + f(S \cup \{x_{e,u}, x_{e',v}\})$.

$$\begin{aligned}
& f(S \cup \{x_{e,u}\}) - f(S) \\
= \quad & \sum_{(i,j)\in R} (d_S(i,j) - d_{S\cup\{x_{e,u}\}}(i,j)) \\
= \quad & \sum_{(u,j)\in R} (d_S(u,j) - d_{S\cup\{x_{e,u}\}}(u,j)) \\
+ \quad & \sum_{(i,u)\in R} (d_S(i,u) - d_{S\cup\{x_{e,u}\}}(i,u)) \qquad (1)
\end{aligned}$$

$$\begin{aligned}
& f(S \cup \{x_{e',v}\}) - f(S \cup \{x_{e,u}, x_{e',v}\}) \\
= \quad & \sum_{(i,j)\in R} (d_{S\cup\{x_{e,u},x_{e',v}\}}(i,j) - d_{S\cup\{x_{e',v}\}}(i,j)) \\
= \quad & d_{S\cup\{x_{e,u},x_{e',v}\}}(v,u) - d_{S\cup\{x_{e',v}\}}(v,u) \\
+ \quad & d_{S\cup\{x_{e,u},x_{e',v}\}}(u,v) - d_{S\cup\{x_{e',v}\}}(u,v) \\
\geq \quad & -2. \qquad (2)
\end{aligned}$$

By the property of shortest paths, we can see that $d_{S\cup\{x_{e,u}\}}(v,u) - d_S(v,u) \leq d_{S\cup\{x_{e,u},x_{e',v}\}}(v,u) - d_{S\cup\{x_{e',v}\}}(v,u)$ and $d_{S\cup\{x_{e,u}\}}(u,v) - d_S(u,v) \leq d_{S\cup\{x_{e,u},x_{e',v}\}}(u,v) - d_{S\cup\{x_{e',v}\}}(u,v)$. By summing (1) and (2) up, we obtain $f(S \cup \{x_{e,u}\}) + f(S \cup \{x_{e',v}\}) \geq f(S) + f(S \cup \{x_{e,u}, x_{e',v}\})$. □

Notice that f can be computed in polynomial time.

By these, MCD is formulated as a submodular set cover problem. Since $\max_{x_{e,u}\in N} f(\{x_{e,u}\}) \leq |R| \max_{u,v} d_\emptyset(u,v) \leq n^3$, the approximation ratio of the greedy algorithm is $O(\log n)$. We obtain the following.

Theorem 4.1
There is a polynomial time algorithm with approximation factor $O(\log n)$ for MCD.

```
MinPivot ($G = (V, E), R$)
  $V' = \{v, w \in V | (v, w) \in R\}$
  for each node $u \in V$ do
    for each node $v \in V'$, store $sp(v, u)$ to $D_v$
    $D(u) = \{D_v | v \in V\}$
  output $\min_{u\in V}\{c.D(u)\}$
```

Figure 2: Algorithm MinPivot

5 3/2-approximation Algorithm

Zheng et al. have proposed a polynomial-time algorithm for MCD, called MinPivot , which achieves approximation ratio two when a set R of requests is full. In this section, we improve the approximation ratio of MinPivot under a certain kind of restriction. More precisely, it is shown that MinPivot is a 3/2-approximation MCD algorithm for undirected graphs even when R is subset-full.

5.1 Algorithm MinPivot

The algorithm MinPivot is designed for directed graphs and any set of requests. In this section, we focus MCD on undirected graphs, thus, we introduce simplified algorithm MinPivot for undirected graphs. A pseudo-code of MinPivot is shown in Figure 2.

In dispersals returned by MinPivot , some node is selected as the pivot. Each request is satisfied by a path via the selected pivot. The algorithm works as follows: it picks up a node u as a candidate of the pivot. Then, for each request $(v,w) \in R$, MinPivot constructs the shortest paths from v to the pivot u and from w to u. That is, the shortest path from v to u is stored in D_v, one from w to u is stored in D_w. Since there is a path from v to w via the pivot u in $D_v \cup D_w$ for each request (v,w), the dispersal satisfies R. For every pivot candidate, the algorithm MinPivot computes the corresponding dispersal as stated above. Finally, the minimum-cost one among all computed dispersals is chosen and returned.

In [11], the following theorem is proved.

Theorem 5.1
For an undirected graph G, MinPivot is a 2-approximation algorithm for MCD on G with a full request, and it completes in $O(nm)$ time.

5.2 Proof of 3/2-approximation

In this subsection, we prove the following theorem.

Theorem 5.2
For an undirected graph G and a subset-full request R, MinPivot is a 3/2-approximation algorithm.

Throughout this subsection, we assume that the request R is subset-full. The set of nodes included in requests in R is denoted by V_R, that is, $V_R = \{u, v|(u, v) \in R\}$. An output of the algorithm MinPivot for an undirected graph G with a request R is denoted by D^{MP}. From the algorithm MinPivot , the following proposition clearly holds.

Proposition 5.1
For an undirected graph G and a set R of requests, if D is a dispersal in which a local dispersal of every node in V_R contains a path from the node to a node u, then $c.D^{MP} \leq c.D$.

The idea of the proof is that we can construct a dispersal D with cost at most $\frac{3}{2} \cdot c.D^{Opt}$, in which there exists a node u such that every node v in V_R has a path from the node to a node u. From Proposition 5.1, it follows that the cost of the solution by MinPivot is bounded by $\frac{3}{2} \cdot c.D^{Opt}$.

In what follows, we show the construction of D. First, we introduce several notations and definitions necessary to the explanation: let x be a node in V_R with a minimum local dispersal in D^{Opt} (i.e., $|D_x^{Opt}| = \min\{|D_v^{Opt}||v \in V_R\}$). We may consider only the case that $|D_x^{Opt}| > 0$ holds because if $|D_x^{Opt}|$ is zero then MinPivot returns an optimal solution since each node v in V_R must has a path from v to x to satisfy a request (v, x). Then, D^{Opt} is equivalent to the solution computed by MinPivot whose pivot candidate is x. We define a rooted tree T from an optimal dispersal D^{Opt}. To define T, we first assign a *weight* to each edge: to any edge in D_x^{Opt}, the weight zero is assigned. All other edges are assigned the weight one. A rooted tree $T = (V, E_T)(E_T \subseteq E)$ is defined as the shortest path tree with root x (in terms of weighted graphs) that spans all nodes in V_R. Let $p_T(u, v)$ be a shortest path from a node u to a node v on the tree T. The weight of a path $p(u, v)$ is defined by the total weight of the edges on the path and denoted by $w.p(u, v)$. For each node v, let $p_T(v, v) = \phi$ and $w.p_T(v, v) = 0$.

Lemma 5.1
On the tree $T = (V, E_T)$ for an optimal dispersal D^{Opt}, $\sum_{v\in V_R} w.p_T(x, v) < c.D^{Opt}$.

Proof.
For the node x, $w.p_T(x, x) < |D_x^{Opt}|$ clearly holds since $|D_x^{Opt}| > 0$. For any other node v in V_R, the set R of requests necessarily includes (x, v) (remind that R is subset-full). To satisfy (x, v), in the optimal dispersal, $D_x^{Opt} \cup D_v^{Opt}$ includes a path $p(x, v)$, and thus, $p(x, v) \setminus D_x^{Opt} \subseteq D_v^{Opt}$. This implies $|p(x, v) \setminus D_x^{Opt}| \leq |D_v^{Opt}|$. Since any edge in D_x^{Opt} has weight zero and all other edges have weight one, the weight of $p(x, v)$ is equal to $|p(x, v) \setminus D_x^{Opt}|$. From the definition of $p_T(x, v)$, we obtain $w.p_T(x, v) \leq w.p(x, v) \leq |D_v^{Opt}|$.

In an optimal dispersal D^{Opt}, the local dispersal D_v^{Opt} of each node v in $V \setminus V_R$ has no edges since there is no request for v in R. Thus, it follows $\sum_{v\in V_R} w.p_T(x, v) < \sum_{v\in V_R} |D_v^{Opt}| = c.D^{Opt}$. □

We construct a desired dispersal D by adding some edges to each local dispersal D_v^{Opt} in the optimal dispersal. When the local dispersal D_v of every node $v \in V_R$ is constructed by adding all the edges in D_x^{Opt} to D_v^{Opt} (i.e., $D_v = D_v^{Opt} \cup D_x^{Opt}$), every local dispersal D_v contains a path from v to x since $D_v^{Opt} \cup D_x^{Opt}$ contains the path to satisfy the request (x, v). In this case, the cost of the dispersal D is at most twice as many as one of the optimal dispersal. Thus, from Proposition 5.1, we prove that the algorithm MinPivot is a 2-approximation algorithm. The idea of our proof of Theorem 5.2 is that we construct a dispersal D by adding each edge in D_x^{Opt} to at most $|V_R|/2$ local dispersals. For each edge e in D_x^{Opt}, let $C(e)$ be the number of nodes from which path to the node x on the tree T includes the edge e: $C(e) = |\{v \in V_R | e \in p_T(x, v)\}|$. The construction of the desired dispersal depends on whether any edge e in D_x^{Opt} satisfies $C(e) \leq |V_R|/2$ or not.

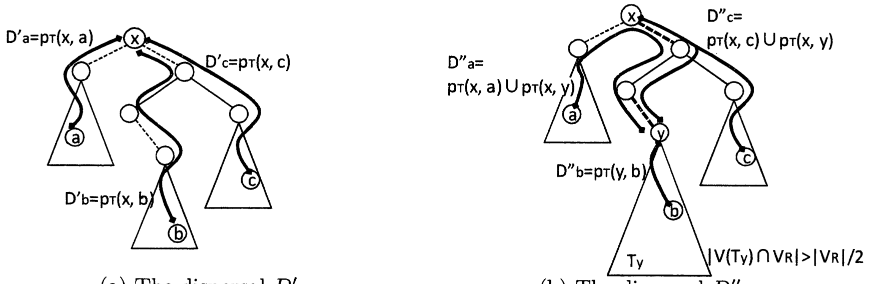
First, we explain the construction of dispersal D' in the case that $C(e) \leq |V_R|/2$ holds for any edge e in D_x^{Opt}: $D' = \{D'_v | v \in V\}$ where

- for the node v in V_R, $D'_v = p_T(x, v)$,
- for the node v in $V \setminus V_R$, $D'_v = \phi$.

Figure 3(a) shows one example of the dispersal D'. In the figure, the dotted edges represent edges included in D_x^{Opt} and the thick curves represent the local dispersal of each node.

Lemma 5.2
$c.D^{MP} \leq c.D' \leq \frac{3}{2} \cdot c.D^{Opt}$

Proof.
By the definitions, $|p_T(x, v)| = w.p_T(x, v) + |p_T(x, v) \cap D_x^{Opt}|$ holds. In addition, we obtain $\sum_{v\in V_R} |p_T(x, v) \cap D_x^{Opt}| = \sum_{e\in D_x^{Opt}} C(e)$ from the



(a) The dispersal D'

(b) The dispersal D''

Figure 3: Examples of the proposed dispersals. The dotted edges represent edges included in D_x^{Opt} and the heavy dotted edges represent edges included in $\hat{D}_x^{Opt}$

definition of $C(e)$. Thus, $c.D' = \sum_{v \in V_R} w.p_T(x,v) + \sum_{e \in D_x^{Opt}} C(e)$. From Lemma 5.1 and the assumption that $C(e) \leq |V_R|/2$, it follows that $c.D' \leq c.D^{Opt} + |D_x^{Opt}| \cdot \frac{|V_R|}{2}$. Now, the size of the local dispersal $|D_x^{Opt}|$ is the minimum of all local dispersals in D^{Opt}, and the local dispersal of the node not included in V_R is empty in D^{Opt}. Therefore, we obtain $|D_x^{Opt}| \cdot |V_R| \leq c.D^{Opt}$. It implies that $c.D' \leq c.D^{Opt} + \frac{1}{2} \cdot c.D^{Opt} \leq \frac{3}{2} \cdot c.D^{Opt}$. Since the local dispersal D'_v of v in V_R includes a path from x to v, $c.D^{MP} \leq c.D'$ holds by Proposition 5.1. □

We consider the case that there is an edge such that $C(e) > |V_R|/2$. Let T_v be a subtree of T induced by the node v and all of v's descendants, and $V(T_v)$ be a set of nodes in T_v. The set of edges in D_x^{Opt} such that $C(e) > |V_R|/2$ is denoted by $\hat{D}_x^{Opt}$. Let y be the node farthest from x of those adjacent to some edge in $\hat{D}_x^{Opt}$.

Lemma 5.3
All edges in $\hat{D}_x^{Opt}$ are on the path $p_T(x,y)$.

Proof.
If a path $p_T(x,w)$ from x to a node $w \in V_R$ contains an edge (u,v), then node w is a descendant of u and v. That is, $w \in V(T_v) \cap V_R$ holds. Thus, from the definition of $C(e)$, we have $C((u,v)) = |V(T_v) \cap V_R|$ for each edge $(u,v) \in D_x^{Opt}$ where u is the parent of v. Therefore, the edge (u,v) satisfies $C((u,v)) > |V_R|/2$ iff $|V(T_v) \cap V_R| > |V_R|/2$.

We prove the lemma by contradiction. Suppose for contradiction that there is an edge (u,v) such that $(u,v) \in \hat{D}_x^{Opt}$ and $(u,v) \notin p_T(x,y)$. Let v be a child of u on T. From $(u,v) \notin p_T(x,y)$, it follows that node v is not an ancestor of the node y on T. Since node y is the farthest node from x, from which the edge to its parent is contained in $\hat{D}_x^{Opt}$, node v is not a descendant of y. Thus, we obtain $V(T_v) \cap V(T_y) = \phi$. In addition, $C((u,v)) = |V(T_v) \cap V_R| > |V_R|/2$ holds. From $V(T_v) \cap V(T_y) = \phi$ and $|V(T_v) \cap V_R| > |V_R|/2$, we obtain $|V(T_y) \cap V_R| \leq |V_R|/2$. It contradicts the definition of the node y. □

In the case that there is an edge such that $C(e) > |V_R|/2$, a dispersal D'' is constructed such that every node in V_R has a path from itself to node y: $D'' = \{D''_v | v \in V\}$ where

- for the node v in $V_R \cap V(T_y)$, $D''_v = p_T(y,v)$,
- for the node v in $V_R \setminus V(T_y)$, $D''_v = p_T(x,v) \cup p_T(x,y)$,
- for the node v in $V \setminus V_R$, $D''_v = \phi$.

Figure 3(b) shows one example of the dispersal D''. The heavy dotted edges represent edges included in $\hat{D}_x^{Opt}$. We can see that local dispersal of each node contains a path from itself to the node y.

Lemma 5.4
$c.D^{MP} \leq c.D'' \leq \frac{3}{2} \cdot c.D^{Opt}$

Proof.
From the definition of the dispersal D'', we obtain $c.D'' \leq \sum_{v \in V_R \cap V(T_y)} |p_T(y,v)| + \sum_{v \in V_R \setminus V(T_y)} (|p_T(x,v)| + |p_T(x,y)|)$. Lemma 5.3 implies that the edge in $\hat{D}_x^{Opt}$ is contained by only nodes in $V_R \setminus V(T_y)$. Moreover, it implies that for each edge $e \in D_x^{Opt}$ that is not on $p_T(x,y)$, $e \in D_x^{Opt} \setminus \hat{D}_x^{Opt}$ and $C(e) \leq |V_R|/2$ hold. Since $|V_R \setminus V(T_y)| \leq |V_R|/2 < |V_R \cap V(T_y)|$, the following

inequalities can be obtained in the same way as the proof of Lemma 5.2:

$$\begin{aligned}
c.D'' &\le \sum_{v\in V_R\cap V(T_y)} w.p_T(y,v) + \sum_{v\in V_R\setminus V(T_y)} (w.p_T(x,v) \\
&\quad +w.p_T(x,y) + |\hat{D}_x^{Opt}|) + \sum_{e\in D_x^{Opt}\setminus \hat{D}_x^{Opt}} C(e) \\
&\le \sum_{v\in V_R\cap V(T_y)} w.p_T(y,v) + \sum_{v\in V_R\setminus V(T_y)} w.p_T(x,v) \\
&\quad +|V_R \setminus V(T_y)|\{w.p_T(x,y) + |\hat{D}_x^{Opt}|\} \\
&\quad + \sum_{e\in D_x^{Opt}\setminus \hat{D}_x^{Opt}} C(e) \\
&\le \sum_{v\in V_R\cap V(T_y)} (w.p_T(y,v) + w.p_T(x,y)) \\
&\quad + \sum_{v\in V_R\setminus V(T_y)} w.p_T(x,v) + \frac{|V_R|}{2}\cdot|\hat{D}_x^{Opt}| \\
&\quad +\frac{|V_R|}{2}\cdot|D_x^{Opt}\setminus \hat{D}_x^{Opt}| \\
&= \sum_{v\in V_R} w.p_T(x,v) + \frac{|V_R|}{2}\cdot|D_x^{Opt}| \le \frac{3}{2}\cdot c.D^{Opt}
\end{aligned}$$

Since the local dispersal D''_v of every node v in V_R includes a path from v to y, $c.D^{MP} \le c.D''$ holds by Proposition 5.1. □

From Lemma 5.2 and Lemma 5.4, Theorem 5.2 is proved.

6 Conclusions

In this paper, we have considered the approximability of MCD, which is the problem that for a given graph G and a set R of requests, requires us to assign edges to each node such that all requests in R are satisfied and the total number of edges all nodes have is minimized. We have shown that the approximation ratio of MCD is $\theta(\log n)$: the result of the lower bound $\Omega(\log n)$ is proved by the reduction of the SET-COVER to MCD, and one of the upper bound $O(\log n)$ is proved by formulating MCD as a submodular set cover problem. In addition, we have proved that when a given graph is restricted to an undirected graph, the algorithm MinPivot guarantees 3/2 approximation ratio even if a given set of request is subset-full.

Our future work is to determine the hardness of MCD when a given request is full or subset-full. We conjecture that when a given set of requests is restricted to a full one, MCD is P and the algorithm MinPivot returns an optimal solution. Now, we tackle this question and investigate graph classes which MinPivot returns optimal dispersals.

Acknowledgment This work is supported in part by the Japan Society for the Promotion of Science: Grant-in-Aid for Young Scientists(B)(19700058), and Hori Information Science Promotion Foundation.

References

[1] N. Alon, D. Moshkovitz, and S. Safra. Algorithmic construction of sets for k-restrictions. *ACM Transactions on Algorithms*, 2(2):153–177, April 2006.

[2] S. Arora and C. Lund. Hardness of approximation. In D. Hochbaum, editor, *Approximation Algorithms for NP-hard problems*, pages 399–446. PWS publishing company, 1995.

[3] S. Capkun, L. Buttyan, and J.-P. Hubaux. Self-organized public-key management for mobile ad hoc networks. *IEEE Transactions on Mobile Computing*, 2(1):52–64, March 2003.

[4] M. G. Gouda and E. Jung. Certificate dispersal in ad-hoc networks. In *in Proceeding of the 24th International Conference on Distributed Computing Systems (ICDCS'04)*, pages 616–623, March 2004.

[5] M. G. Gouda and E. Jung. Stabilizing certificate dispersal. In *in Proceeding of the 7th International Symposium on Self-Stabilizing Systems (SSS'05)*, pages 140–152, October 2005.

[6] J. Hubaux, L. Buttyan, and S. Capkun. The quest for security in mobile ad hoc networks. In *in Proceeding of the 2nd ACM international symposium on Mobile ad hoc networking and computing (Mobihoc'01)*, pages 146–155, October 2001.

[7] E. Jung, E. S. Elmallah, and M. G. Gouda. Optimal dispersal of certificate chains. In *in Proceeding of the 18th International Symposium on Distributed Computing (DISC'04)*, pages 435–449, October 2004.

[8] M. Karpinski. Approximating bounded degree instances of NP-hard problems. In *in Proceeding of the 13th Symposium on Fundamentals of Computations Theory (FCT'01)*, August 2001.

[9] L. A. Wolsey. An analysis of the greedy algorithm for the submodular set covering problem. *Combinatorica*, 2(4):385–393, 1982.

[10] H. Zheng, S. Omura, J. Uchida, and K. Wada. An optimal certificate dispersal algorithm for mobile ad hoc networks. *IEICE Transactions on Fundamentals*, E88-A(5):1258–1266, May 2005.

[11] H. Zheng, S. Omura, and K. Wada. An approximation algorithm for minimum certificate dispersal problems. *IEICE Transactions on Fundamentals*, E89-A(2):551–558, February 2006.