Available online at www.sciencedirect.com
SCIENCE DIRECT®

Theoretical Computer Science 332 (2005) 1–17

Theoretical Computer Science

www.elsevier.com/locate/tcs

k-Center problems with minimum coverage

Andrew Lim[a], Brian Rodrigues[b], Fan Wang[a], Zhou Xu[a,*]

[a] *Department of Industrial Engineering and Engineering Management, Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong*

[b] *School of Business, Singapore Management University, Singapore*

Received 25 September 2003; received in revised form 5 May 2004; accepted 23 August 2004
Communicated by O. Ibarra

Abstract

In this work, we study an extension of the k-center facility location problem, where centers are required to service a minimum of clients. This problem is motivated by requirements to balance the workload of centers while allowing each center to cater to a spread of clients. We study three variants of this problem, all of which are shown to be $\mathcal{NP}$-hard. In-approximation hardness and approximation algorithms with factors equal or close to the best lower bounds are provided. Generalizations, including vertex costs and vertex weights, are also studied.
© 2004 Elsevier B.V. All rights reserved.

Keywords: Approximation algorithm; k-center problem; Minimum coverage

1. Introduction

The k-center problem is a well-known facility location problem and can be described as follows: Given a complete undirected graph $G = (V, E)$, a metric $d : V \times V \to \mathbb{R}_+$ and a positive integer k, we seek a subset $U \subseteq V$ of at most k centers which minimizes the maximum distances from points in V to U. Formally, the objective function is given by

$$\min_{U \subseteq V, |U| \leqslant k} \max_{v \in V} \min_{r \in U} d(v, r).$$

* Corresponding author. Tel.: +852-2358-8642; fax: +852-2358-0062.
E-mail address: xuzhou@ust.hk (Z. Xu).

0304-3975/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.tcs.2004.08.010

As a typical example, we may want to set up k service centers (e.g., police stations, fire stations, hospitals, polling centers) and minimize the maximum distances between each client and these centers. The problem is known to be $\mathcal{NP}$-hard [4].

A factor ρ-approximation algorithm for a minimization problem is a polynomial time algorithm which guarantees a solution within at most ρ times the optimal cost. For the k-center problem, Hochbaum and Shmoys presented a factor 2-approximation algorithm and proved that no factor better than 2 can be achieved unless $\mathcal{P} = \mathcal{NP}$ [5]. Approximation algorithms for other k-center problems, where vertex costs are considered or when vertex weights are used have been extensively studied [3,6,11]. More recently, Bar-Ilan, Kortsarz and Peleg investigated an interesting generalization of capacitated k-center problem where the number of clients for each center was restricted to a service capacity limit or maximum load [1]. Their work was improved recently by Khuller and Sussmann [9]. On the other hand, to ensure that backup centers are available for clients, Krumke developed a “fault tolerant” k-center problem, where the objective was to minimize maximum distances as before, but where each client is required to be covered by a minimum number of centers [10]. Approximation algorithms for these problems were improved and extended in [8,2].

In these studies, no provision was made to ensure that centers provide a minimum coverage of clients. In the fault-tolerant problem, the client demand side of the problem is guaranteed coverage by a minimum number of centers (less than k), yet, on the supply side, there is no guarantee that each center services a minimum number of clients. In realistic applications however, such coverage is a common requirement. For example, in planning the location of hospitals, it would be expected that each hospital services a minimum number of neighborhoods. This would impose a balanced workload among hospitals and allow for economies of scale. Moreover, in cases when each center is equipped to provide a variety of services, a spread of clients covered is more likely to benefit service providers and clients alike. For example, where warehouses stock a variety of products, it would be beneficial if each services a spread of customers whose demands are more likely to include the range of products available. In this work, we address these provisions by extending the basic k-center problem to include a minimum coverage requirement. We allow coverage by different centers to overlap allowing clients to choose from a number of centers. In the problem, we minimize distances as in the basic k-center problem and require that every vertex in V is covered by one of the at most k selected centers in U. Further, each center in U must cover at least q vertices in V, where q is a non-negative integer, at most as large as $|V|$, which defines the minimum coverage for each center.

We call this a q-all-coverage k-center problem, with an objective function given by

$$\min_{U \subseteq V, |U| \leqslant k} \max \left(\max_{v \in V} \min_{r \in U} d(v, r), \max_{r \in U} d_q(V, r) \right),$$

where $d_q(V, r)$ is the distance to r from its qth closest vertex in V. Note that because $r \in V$, its closest vertex is r itself.

The left sub-figure of Fig. 1 shows an instance of a 3-all-coverage 2-center problem, where each of the two centers, denoted by filled triangles, cover three vertices (including itself) within a distance l_1.

Further, two variations to this problem will be studied. The first is a q-coverage k-center problem, for which only vertices in $V - U$ are counted in the coverage of every center

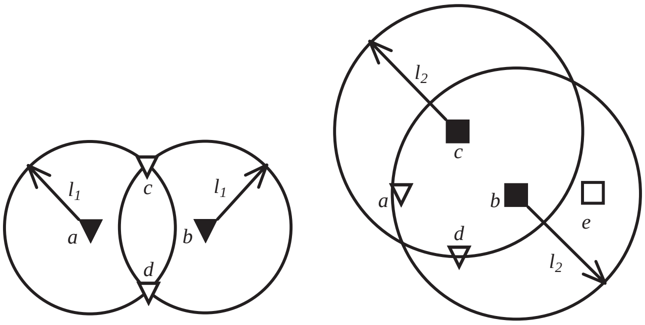

Fig. 1. Instances of k-center and k-supplier problems.

in U. Its objective function is

$$\min_{U \subseteq V, |U| \leqslant k} \max \left(\max_{v \in V-U} \min_{r \in U} d(v,r), \max_{r \in U} d_q(V-U, r) \right),$$

where $d_q(V-U, r)$ is the distance of r from its qth closest vertex in $V-U$. For example, in the left sub-figure of Fig. 1, the two centers only satisfy the 2-coverage 2-center problem within l_1, because centers themselves are not counted in their own coverage.

The second is a q-coverage k-supplier problem for which V is partitioned into two disjoint subsets: S, a supplier set, and D, a demand set. The problem is then to find a subset U of at most k centers in S to minimize distances, where not only is every demand point in D covered by a center in U, but every center in U must cover at least q demands in D. Here, the objective function is

$$\min_{U \subseteq S, |U| \leqslant k} \max \left(\max_{v \in D} \min_{r \in U} d(v,r), \max_{r \in U} d_q(D, r) \right),$$

where $d_q(D, r)$ is the distance of r from its qth closest demands in D.

The right sub-figure of Fig. 1 gives an instance of the 2-coverage 2-supplier problem. Among the three suppliers denoted by rectangles, two filled ones are selected to be centers, each of which covers two demand points, distinguished by triangles, within a distance l_2.

Additionally, these three problems can be generalized by the inclusion of vertex costs and vertex weights, as has been done for the basic k-center problem. To include costs, we define a cost $c(v)$ for each vertex v in V, where we now require $\sum_{r \in U} c(r) \leqslant k$. This cost generalization is useful, for example, in the case of building centers where the cost for centers can vary and when there is a limited budget as is the case in practice.

To extend the problems by including weights, we take $w(v)$ be the weight of each vertex v in V so that the weighted distance to vertex v from vertex u in V is $w(u, v) = d(u, v) \cdot w(v)$. For any vertex $v \in V$ and $X \subseteq V$, we let $w_q(X, v)$ to be the qth closest weighted distance of v from the vertices in X. With this, the three variants can be generalized to weighted models by replacing distances d and d_q in the objective functions with the weighted distances w and w_q, respectively. Weighted distances can be useful, for example, when $1/w(v)$ is modelled to be the response speed of the center at vertex v, which then makes $w(u, v) = d(u, v) \cdot w(v)$ its response time.

Finally, by considering both vertex costs and vertex weights, we study the most general extensions for the three new problems.

Throughout this paper, *OPT* denotes the optimal value of the objective function. We assume that the complete graph $G = (V, E)$ is directed, where $V = \{v_1, \ldots, v_n\}$ and $E = V \times V = \{e_1, \ldots, e_m\}$, where $m = n^2$, where each vertex $v \in V$ has a self-loop $(v, v) \in E$ with distance $d(v, v) = 0$. For each edge $e_i \in E$, let $d(e_i)$ and $w(e_i)$ denote its distance and its weighted distance, respectively.

For any graph *H*, a vertex v is said to *dominate* a vertex *u* in *H*, if and only if v equals *u* ($v = u$) or v has an edge from *u* in *H*. Based on this, we let $deg(v)$ denote the number of vertices dominated by v in *H*. If *H* is undirected, $deg(v)$ is the *degree* of a vertex v, i.e., the number of adjacent edges including the possible self-loop (v, v). If *H* is undirected, let $I(H)$ denote its *maximal independent set* [4], in which no two different vertices share an edge and no vertex outside $I(H)$ can be included while preserving its independence. When *H* has more than one maximal independent set, let $I(H)$ denote any one of them unless we explicitly construct $I(H)$.

We present approximation algorithms for the three problems considered in this paper and their generalizations. Our methods extend from the threshold technique used for the basic *k*-center problem [6], and are designed to address the new minimum coverage constraints included.

The paper is organized as follows. In the next section, we summarize the main results of this work and, in subsequent sections, we provide approximation hardness and algorithms for the three problems: the *q*-all-coverage *k*-center problem, the *q*-coverage *k*-center problem, and the *q*-coverage *k*-supplier problem. For each problem considered, approximation algorithms are provided for the basic case and for its weight, cost, and weight plus cost generalizations. In Section 6, we provide a conclusion.

2. Main results

Our main results are summarized in Table 1. In the table, [a] indicates the best possible approximation factors have been achieved, which are shown to be 2, 2 and 3 for the three problems, respectively, unless $\mathcal{P} = \mathcal{NP}$. These optimal results include the basic cases of all the three problems considered, and the weight and the cost generalizations of the *q*-coverage *k*-supplier problem. Moreover, for the weight and the cost generalizations of the other two problems, approximation algorithms are provided with constant factors, all of which are close to their best possible approximation factor of 2. Especially, for the cost generalization of the *q*-all-coverage *k*-center problem indicated by [b] in Table 1, a 3-approximation algorithm is achieved which matches the best known approximation factor for the cost generalization of the classical *k*-center problem [6].

Further, the approximation algorithms for the cost generalizations of the three problems can be extended to solve their weight plus cost generalizations. Let β denote the ratio between the maximum and the minimum value of weights. Their approximation factors are consistent with those of their cost generalizations, which hold when $\beta = 1$.

Table 1
Summary of approximation factors

	Basic	Weights	Costs	Weights + costs
q-All-coverage K-center	2^a	3	3^b	$2\beta+1$
q-Coverage K-center	2^a	4	4	$3\beta+1$
q-Coverage K-supplier	3^a	3^a	3^a	$2\beta+1$

[a] Achieves the best possible approximation factor unless $\mathcal{P}=\mathcal{NP}$.
[b] Matches the best known approximation factor.
β is the ratio between the maximum value and the minimum value of weights.

3. q-All-coverage k-center problems

The following hardness result for the q-all-coverage k-center problem can be proved by extending the reduction from the *Domination Set* problem [4] used for the classical k-center problem [7].

Theorem 1. *Given any fixed non-negative integer q, there is no* $(2-\varepsilon)$*-approximation algorithm for the q-all-coverage k-center problem, unless* $\mathcal{NP}=\mathcal{P}$.

The best possible approximation factor of 2 can be achieved by Algorithm 1. We first sort edges in E by order of non-decreasing distances, i.e., $d(e_1)\leqslant d(e_2)\leqslant\cdots\leqslant d(e_m)$. Let $G_i=(V,E_i)$, where $E_i=\{e_1,\dots,e_i\}$ for $1\leqslant i\leqslant m$. Thus, if G_i has a set U of at most k vertices that dominate all vertices in G_i, and each vertex of U dominates at least q vertices (including itself) in G_i, then U provides at most k centers to the problem with at most $d(e_i)$ distance. Let i^* denote the smallest such index. So $d(e_i^*)=OPT$ is the optimal distance.

To find a lower bound for OPT, construct an undirected graph H_i. H_i contains an edge (u,v), where $u,v\in V$ might be equal if and only if there exists a vertex $r\in V$ with $deg(r)\geqslant q$ and both (u,r) and (v,r) are in G_i. It is clear that the self loop (v,v) remains in H_i for each $v\in V$, and that if $(v,u)\in G_i$ then $(v,u)\in H_i$ since (v,v) and (u,v) are in G_i. As any two vertices dominated by the same vertex of G_i are adjacent in H_i, H_{i^*} satisfies the following:

(1) for each vertex $v\in V$, $deg(v)\geqslant q$ in H_{i^*}, including its self-loop;
(2) the size of its maximal independent set $|I(H_{i^*})|\leqslant k$.

Accordingly, suppose that the threshold j is the minimum index i leading H_i to satisfy the above two conditions, then we have $j\leqslant i^*$, which gives $d(e_j)\leqslant OPT$.

Finally, selecting vertices in H_j, we have $|I(H_j)|\leqslant k$. So, centers in $I(H_j)$ dominate all vertices of V in H_j, and each $v\in I(H_j)$ dominates at least $deg(v)\geqslant q$ vertices (including itself) of V in H_j. By the triangle inequalities, we know $d(u,v)\leqslant 2d(e_j)\leqslant 2OPT$, for every (u,v) in H_j. So the set U gives at most k centers with at most $2OPT$ distance, which establishes the following theorem for the approximation factor of Algorithm 1.

Algorithm 1 Basic q-all-coverage k-center

1: Sort edges so that $d(e_1) \leqslant d(e_2) \leqslant \cdots \leqslant d(e_m)$, and construct $H_1, H_2, \ldots, H_m$
2: Compute a maximal independent set, $I(H_i)$, in each graph H_i, where $1 \leqslant i \leqslant m$
3: Find the threshold j, denoting the smallest index i, such that $|I(H_i)| \leqslant k$, and for each $v \in V$, $deg(v) \geqslant q$ in H_i
4: Return $I(H_j)$.

Theorem 2. *Algorithm* 1 *gives an approximation factor of* 2 *for the q-all-coverage k-center problem.*

3.1. Any q with weights

From Algorithm 1, we have a 3-approximation Algorithm 2 for the weighted case of the q-all-coverage k-center problem. Firstly, sort edges by non-decreasing weighted distances, i.e., $w(e_1) \leqslant w(e_2) \leqslant \cdots \leqslant w(e_m)$ and let $G_i = (V, E_i)$ where $E_i = \{e_1, \ldots, e_i\}$. The graph H_i for G_i contains an edge (u, v), where $u, v \in V$ might be equal if and only if there exists a vertex r which has both (u, r) and (v, r) in G_i, which implies $w(u, r) \leqslant w(e_i)$ and $w(v, r) \leqslant w(e_i)$. To bound the optimum weighted distance (OPT), find the threshold j which denotes the minimum index i such that the degree of each vertex in H_i is at least q and the size of its maximal independent set $I(H_i)$ is at most k. Hence, it can be ensured that $w(e_j) \leqslant OPT$. Finally, consider each vertex $v \in V$. Among all $u \in V$ with $w(v, u) \leqslant w(e_i)$, let $g_i(v)$ denote the vertex having the smallest weight, i.e., the least weighted neighbor of v in G_i. Shifting every $v \in I(H_j)$ to $g_j(v)$, we obtain the set U which guarantees an approximation factor of 3 given by the following theorem:

Algorithm 2 Weighted q-all-coverage k-center

1: Sort edges so that $w(e_1) \leqslant w(e_2) \leqslant \cdots \leqslant w(e_m)$, and construct $H_1, H_2, \ldots, H_m$
2: Compute a maximal independent set, $I(H_i)$, in each graph H_i, where $1 \leqslant i \leqslant m$
3: Find the threshold j which denotes the smallest index i such that $|I(H_i)| \leqslant k$ and $deg(v) \geqslant q$ in H_i for each $v \in V$
4: Shift vertices in $I(H_j)$ to their least weighted neighbors in G_j, giving $U = \{g_j(v) | v \in I(H_j)\}$
5: Return U;

Theorem 3. *Algorithm* 2 *gives an approximation factor of* 3 *for the weighted q-all-coverage k-center problem.*

Proof. Firstly, by $|I(H_j)| \leqslant k$ and $U = \{g_j(v) | v \in I(H_j)\}$, we have $|U| \leqslant k$. Next, as shown in Fig. 2, for any vertex $u \in V$, there must exist v in $I(H_j)$ with (u, v) in H_j, which gives a vertex $r \in V$ with both (u, r) and (v, r) in G_j. Hence, $w(u, r) \leqslant w(e_j)$ and $w(v, r) \leqslant w(e_j)$. Since $w(g_j(v)) \leqslant w(r)$ and $w(v, g_j(v)) \leqslant w(e_j)$, u is covered by $g_j(v) \in U$ within $w(u, g_j(v)) \leqslant (d(u, r) + d(v, r) + d(v, g_j(v)))w(g_j(v)) \leqslant 3w(e_j)$. Furthermore,

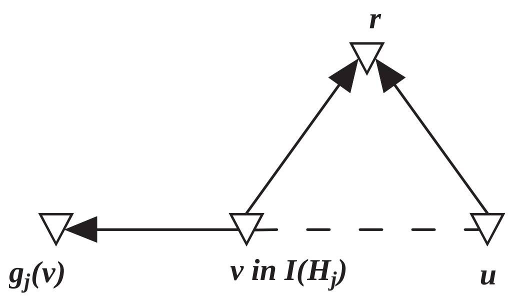

Fig. 2. Diagram for the proof of Theorem 3.

the degree of any vertex $v \in I(H_j)$ is at least q, which implies at least q vertices like u, equivalent or adjacent to v in H_j, can be covered by $g_j(v) \in U$ within $3w(e_j)$. Since $w(e_j) \leqslant OPT$, the approximation factor is 3 for Algorithm 2. □

3.2. Any q with weights and costs

We now give a $(2\beta + 1)$-approximation Algorithm 3 for the most general case, where vertices have both weights and costs. If only cost is considered, a 3-approximation can be achieved where $\beta = 1$.

Algorithm 3 is similar to Algorithm 2 except that a new set U_i is constructed by shifting each $v \in I(H_i)$ to $s_i(v)$, where $s_i(v)$ is the vertex who has the lowest cost among all $u \in V$ with $w(v, u) \leqslant w(e_i)$. Hence $s_i(v)$ is called the cheapest neighbor of v in G_i and we take $U_i = \{s_i(v) | v \in I(H_i)\}$ and $c(U_i)$ to denote the total costs of vertices in U_i. Because no two vertices in $I(H_i)$ are dominated by a common vertex in G_i, the index i^* with $w(e_{i^*}) = OPT$ leads H_{i^*} to satisfy the following:
(1) for each vertex $v \in V$, $deg(v) \geqslant q$ in H_{i^*}, including its self-loop;
(2) $c(U_{i^*}) \leqslant k$.
Finding the threshold j to be the minimum index i which causes H_i to satisfy the above two conditions, we have $j \leqslant i^*$ and $w(e_j) \leqslant OPT$. Furthermore, we will prove that the U_j provides at most k cost centers ensuring an approximation factor of $(2\beta + 1)$ in the following:

Algorithm 3 Weighted and cost q-all-coverage k-center)

1: Sort edges so that $w(e_1) \leqslant w(e_2) \leqslant \cdots \leqslant w(e_m)$, and construct $H_1, H_2, \ldots, H_m$
2: Compute a maximal independent set, $I(H_i)$, in each graph H_i, where $1 \leqslant i \leqslant m$
3: Let $U_i = \{s_i(v) | v \in I(H_i)\}$, where $s_i(v)$ is the cheapest neighbor of v in G_i
4: Find threshold j, denoting the smallest index i, such that $c(U_i) \leqslant k$, and for each vertex $v \in V$, $deg(v) \geqslant q$ in H_i
5: Return U_j.

Theorem 4. *Algorithm* 3 *gives an approximation factor of* $(2\beta + 1)$ *for the weighted and cost q-all-coverage k-center problem*.

Proof. Because $c(U_j) \leqslant k$ and $w(e_j) \leqslant OPT$ have been shown, we need only show that the objective function distance given by U_j is at most $(2\beta + 1)w(e_j)$. On one hand, for any vertex $u \in V$, there exists a vertex $v \in I(H_j)$ adjacent to u in H_j. This implies there is a vertex $r \in V$ with both $w(u, r) \leqslant w(e_j)$ and $w(v, r) \leqslant w(e_j)$. Since $w(r) \leqslant \beta w(s_j(v))$ and $w(v, s_j(v)) \leqslant w(e_j)$, we know that v is covered by $s_j(v) \in U_j$ within $w(u, s_j(v)) \leqslant (d(u, r) + d(v, r) + d(v, s_j(v)))w(s_j(v)) \leqslant (2\beta + 1)w(e_j)$. On the other hand, because the degree of any vertex $u \in I(H_j)$ is at least q, there are at least q vertices like u, equivalent or adjacent to v in H_j, covered by $s_j(v) \in U_j$ within at most $(2\beta + 1)w(e_j)$ weighted distance. Since $w(e_j) \leqslant OPT$, the approximation factor is $(2\beta + 1)$ for Algorithm 3. $\square$

4. q-Coverage k-center problems

Compared with the q-all-coverage k-center problem, the q-coverage k-center problem has an additional stipulation: for each selected center v, at least the q vertices covered by v should be outside the set of selected centers.

To determine its hardness, we provide the following theorem, which can be shown by a modified reduction from the *Domination Set* problem [4] used for Theorem 1.

Theorem 5. *Given any fixed non-negative integer q, there is no* $(2 - \varepsilon)$*-approximation algorithm for the q-coverage k-center problem, unless* $\mathcal{NP} = \mathcal{P}$.

The best possible approximation factor of 2 can be achieved for the q-coverage k-center problem by Algorithm 4 which is similar to Algorithm 1. The only difference is that the threshold j, found here, must cause the degree $deg(v)$ to be at least $q + 1$ in H_j instead of q for each vertex $v \in V$, since self-loops might exist but each center should be adjacent to q vertices other than itself. The approximation factor of 2 is proved by the following theorem:

Algorithm 4 Basic q-coverage k-center

1: Sort edges so that $d(e_1) \leqslant d(e_2) \leqslant \cdots \leqslant d(e_m)$, and construct $H_1, H_2, \ldots, H_m$
2: Compute a maximal independent set, $I(H_i)$, in each graph H_i, where $1 \leqslant i \leqslant m$
3: Find threshold j and denote the smallest index i, such that $|I(H_i)| \leqslant k$, and that for each $v \in V$, $deg(v) \geqslant q + 1$ in H_i
4: Return $I(H_j)$.

Theorem 6. *Algorithm* 4 *gives an approximation factor of* 2 *for the q-coverage k-center problem*.

Proof. By the same analysis for Theorem 2, we know that $d(e_j) \leqslant OPT$, and that $I(H_j)$ provides at most k centers which cover all the vertices within at most $2d(e_j)$. Since each vertex $v \in I(H_j)$ is adjacent to at least q vertices other than itself in H_j, to prove its q-coverage within $2d(e_j)$ we need only show that no two vertices in $I(H_j)$ are adjacent to each other in H_j. This is obvious, since $I(H_j)$ is an independent set of H_j. Since $2d(e_j) \leqslant 2OPT$, the approximation factor is 2. □

4.1. *Any q with weights*

The weighted case of the *q*-coverage *k*-center problem can be solved by Algorithm 5, which is more intricate than the previous algorithms and can be described as follows:

First, after sorting the *m* edges, an undirected graph P_i, instead of H_i, is constructed from G_i for $1 \leqslant i \leqslant m$. The construction is as follows: Let Q_i be the subset of $v \in V$ with $deg(v) \geqslant q + 1$. For any $u, v \in V$, where *u* and v might be equal, an edge (u, v) is in P_i if and only if there exits $r \in Q_i$ so that both (u, r) and (v, r) are in G_i. Consider the index i^*, where $w(e_{i^*}) = OPT$. Because each selected center must dominate at least q vertices other than itself, and no two vertices in $I(P_{i^*})$ are dominated by the same vertex in G_{i^*}, we observe that

(1) each vertex of *V* is dominated by at least one vertex of Q_{i^*} in G_{i^*};
(2) the size of $I(P_{i^*})$ can be at most as large as *k*, i.e. $|I(P_{i^*})| \leqslant k$.

Accordingly, define the threshold *j* to be the smallest index *i*, such that Q_i dominates all vertices of *V*, and $|I(P_i)| \leqslant k$. The two observations above imply $w(e_j) \leqslant OPT$.

Second, shift vertices in $I(P_j)$ are as follows. For each vertex $v \in I(H_j)$, let $p(v)$ denote the smallest weighted vertex, among all $u \in Q_j$ with an edge (v, u) in G_j. This gives $U' = \{p(v) | v \in I(P_j)\}$.

Now, consider an undirected graph, $H' = (U', E')$, where, for any two vertices *u* and v in $I(P_j)$, an edge $(p(u), p(v)) \in E'$ if and only if either $(p(v), p(u))$ or $(p(u), p(v))$ is in G_j. Its maximal independent set, denoted by $I(H')$, can be obtained greedily by Algorithm 6. It is easily seen that for any vertex $u \in U' - I(H')$, there exists a vertex $v \in I(H')$ with $(u, v) \in E'$ and $w(v) \leqslant w(u)$, where v could be the vertex that marks *u* in Algorithm 6.

Now, we prove that $I(H')$ provides at most *k* centers ensuring a 4-approximation factor to establish the following theorem.

Theorem 7. *Algorithm* 5 *gives an approximation factor of* 4 *for the weighted q-coverage k-center problem.*

Proof. Noting $w(e_j) \leqslant OPT$ and $|I(H')| \leqslant |U'| \leqslant |I(P_j)| \leqslant k$, we need only prove the following two facts:

(1) each $p(v) \in I(H')$ covers at least q vertices $u \in V - I(H')$ within $w(u, p(v)) \leqslant 4w(e_j)$, where $v \in I(P_j)$;
(2) each $u \in V - I(H')$ is covered by a certain vertex $p(v) \in I(H')$ within $w(u, p(v)) \leqslant 4w(e_j)$, where $v \in I(P_j)$.

Algorithm 5 Weighted q-coverage k-center

```
Sort edges so that w(e_1) ⩽ w(e_2) ⩽ ··· ⩽ w(e_m).
For each 1 ⩽ i ⩽ m, let Q_i = {v | deg(v) ⩾ q + 1}.
Construct graphs, P_1, P_2, ..., P_m.
Compute a maximal independent set, I(P_i) for each graph P_i where 1 ⩽ i ⩽ m.
Find the threshold j, denote the smallest index i, such that Q_i dominates all vertices of
  V in G_i, and |I(P_i)| ⩽ k.
For each vertex v ∈ I(P_j), let p(v) denote the lowest weighted vertex among all
  vertices u ∈ Q_j with an edge (v, u) in G_j.
Shift vertices in I(P_j) by U' = {p(v) | v ∈ I(P_j)}.
Construct H' = (U', E') from G_j, where for any two vertices u and v in I(P_j), an
  edge (p(u), p(v)) ∈ E' if and only if either (p(v), p(u)) or (p(u), p(v)) is in G_j.
Call Algorithm 6 to obtain I(H'), a maximal independent set of H', insuring that for
  any vertex u ∈ U' − I(H'), there exists a vertex v ∈ I(H') with (u, v) ∈ E' and
  w(v) ⩽ w(u).
Return I(H').
```

Algorithm 6 Maximal Independent set of $H' = (U', E')$ with weights w

```
U ← ∅;
while U' ≠ ∅ do
    Choose the vertex v, which has the smallest weight w(u) among all u ∈ U';
    U ← U + {v} and U' ← U' − {v};
    Mark all the vertices u ∈ U' adjacent to v, i.e (u, v) ∈ E', by U' ← U' − {u};
end while
Return U which is a maximal independent set of H'.
```

On one hand, consider each $p(v) \in I(H')$, where $v \in I(P_j)$. Because $I(H') \subseteq U' \subseteq Q_j$, we know $p(v) \in Q_j$, and so, there exist at least q vertices, other than $p(v)$, which are dominated by $p(v)$ in G_j. Moreover, each vertex u of these q vertices is not in $I(H')$, because otherwise, the edge $(u, p(v))$ in G_j implies an edge $(u, p(v))$ in H', contradicting to the independence of $I(H')$. Note that $w(u, p(v)) \leqslant w(e_j) \leqslant 4w(e_j)$. Fact 1 is proved.

On the other hand, consider each vertex $u \in V - I(H')$. As shown in Fig. 3, because $I(P_j)$ is a maximal independent set of P_j, there exists a vertex $t_1 \in I(P_j)$ with an edge (u, t_1) in P_j. (Note that if u is in $I(P_j)$, a self loop (u, u) must be in P_j because all vertices of V are dominated by Q_j). Thus, we know that $p(t_1)$ is in H'. Since $I(H')$ is a maximal independent set of H', there exists a vertex $p(t_2) \in I(H')$ for $t_2 \in I(P_j)$, with $w(p(t_2)) \leqslant w(p(t_1))$ and an edge $(p(t_1), p(t_2))$ in H'. So $w(p(t_1), p(t_2)) \leqslant w(p(t_2), p(t_1))$, implying $(p(t_1), p(t_2))$ is in G_j. Because (u, t_1) is in P_j, there exits a vertex $a \in Q_j$ dominating both u and t_1 in G_j, leading $w(p(t_1)) \leqslant w(a)$. Noting that weighted distances of (u, a), (t_1, a), $(t_1, p(t_1))$, and $(p(t_1), p(t_2))$ are all at most $w(e_j)$, we have $w(u, p(t_2)) \leqslant (d(u, a) + d(t_1, a) + d(t_1, p(t_1)) + d(p(t_1), p(t_2)))w(p(t_2)) \leqslant 4w(e_j)$. This proves Fact 2 and completes the proof. □

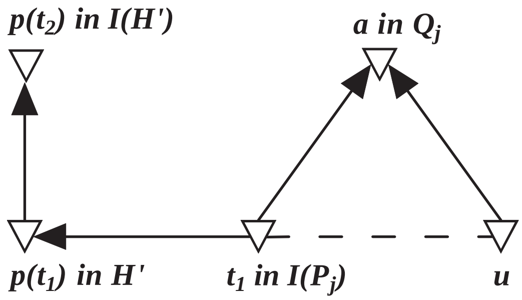

Fig. 3. Diagram of the proof for Theorem 7.

4.2. Any q with weights and costs

As shown in Algorithm 7, the basic idea employed to solve the q-coverage k-center problem with weights and costs is to combine and modify Algorithm 3 and Algorithm 4. For the problem here, we construct $H_1, \ldots, H_m$ first and sort edges $e_1, \ldots, e_m$ by their non-decreasing weighted distances.

However, to find the threshold j we need a new approach. For $1 \leqslant i \leqslant m$, an undirected graph H_i' is generated in the following manner: For any two vertices $u, v \in V$, the edge (u, v) is in H_i', if and only if there exists a vertex $r \in V$, such that either (u, r) is in G_i and (v, r) is in H_i, or (v, r) is in G_i and (u, r) is in H_i. We then compute $I(H_i')$, a maximal independent set of H_i', and shift each vertex $v \in I(H_i')$ to its lowest cost neighbor $s_i(v)$ in G_i; this forms the set $U_i' = \{s_i(v) | v \in I(H_i')\}$.

Now, we find the threshold j, which is the minimal index i, giving $deg(v) \geqslant q + 1$ in H_i for each vertex $v \in V$ and $c(U_i') \leqslant k$, where $c(U_i')$ denotes the total cost of vertices in U_i'. Observing that H_i' is a subgraph of H_i, we know $I(H_i')$ is also an independent set of H_i. By similar arguments for Algorithms 3 and 4, we derive $w(e_j) \leqslant OPT$.

To obtain the approximation factor, we prove that U_j' gives at most k cost centers within at most $(3\beta + 1)OPT$ weighted distance as follows.

Algorithm 7 Weighted and cost q-coverage k-center

1: Sort edges so that $w(e_1) \leqslant w(e_2) \leqslant \cdots \leqslant w(e_m)$, and construct $H_1, H_2, \ldots, H_m$.
2: Construct $H_1', H_2', \ldots, H_m'$.
3: Compute a maximal independent set, $I(H_i')$, in each graph H_i', where $1 \leqslant i \leqslant m$.
4: Let $U_i' = \{s_i(v) | v \in I(H_i')\}$, where $s_i(v)$ is the cheapest neighbor of v in G_i.
5: Find j, denoting the smallest index i, such that $c(U_i') \leqslant k$, and that for each vertex $v \in V$, $deg(v) \geqslant q + 1$ in H_i.
6: Return U_j'.

Theorem 8. *Algorithm* 7 *gives an approximation factor of* $3\beta + 1$ *for the weighted and cost q-coverage k-center problem.*

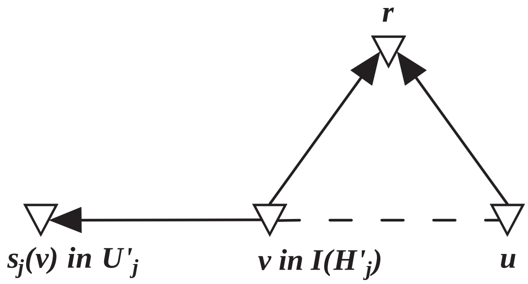

Fig. 4. Diagram for the proof of Theorem 8: the center $s_j(v) \in U'_j$ can cover any vertex u adjacent to $v \in I(H'_j)$ within $(3\beta + 1)w(e_j)$ weighted distance.

Proof. We have obtained $w(e_j) \leqslant OPT$ and $c(U'_j) \leqslant k$. To show q-coverage for each $s_j(v) \in U'_j$, where $v \in I(H'_j)$, we estimate the weighted distance, $w(u, s_j(v))$, for any vertex u equivalent or adjacent to v in H_j but other than $s_j(v)$. As shown in Fig. 4, since there exists a vertex r with $w(u, r) \leqslant w(e_j)$ and $w(v, r) \leqslant w(e_j)$, noting $w(v, s_j(v)) \leqslant w(e_j)$, we can obtain $w(u, s_j(v)) \leqslant (d(u, r) + d(v, r) + d(v, s_j(v)))w(s_j(v)) \leqslant (2\beta + 1)w(e_j) \leqslant (3\beta + 1)OPT$.

Moreover, the vertex u cannot be in U'_j, because otherwise assuming $u = s_j(v')$, where v' is in $I(H'_j)$ but other than v. Since $(v', u) \in G_j$ and $(v, u) \in H_j$, we have $(v', v) \in H'_j$, leading contradiction to the independence of $I(H'_j)$.

Therefore, since $deg(v) \geqslant q + 1$ in H_j, we have that $V - U'_j$ contains at least q such vertices as u, equivalent or adjacent to v in H_j, to be covered by $s_j(v)$ within $(3\beta + 1)OPT$ weighted distance.

Now we prove that any vertex $u \in V - U'_j$ is covered by a certain vertex in U'_j within $(3\beta + 1)OPT$ weighted distance. Because $I(H'_j)$ is a maximal independent set of H'_j, there exists a vertex $v \in I(H'_j)$ with $(u, v) \in H'_j$. This implies a vertex $r \in V$, having either $(u, r) \in G_j$ and $(v, r) \in H_j$, or $(v, r) \in G_j$ and $(u, r) \in H_j$. These two possible cases can both be proved to satisfy $w(u, s_j(v)) \leqslant (3\beta + 1)OPT$ as follows.

For the first case, if $(u, r) \in G_j$ and $(v, r) \in H_j$, as shown in the left of Fig. 5, then $w(u, r) \leqslant w(e_j)$, and there exists a vertex t with $w(v, t) \leqslant w(e_j)$ and $(r, t) \leqslant w(e_j)$. Noting $w(v, s_j(v)) \leqslant w(e_j)$, we can estimate the weighted distance $w(u, s_j(v))$ by $w(u, s_j(v)) \leqslant (d(u, r) + d(r, t) + d(v, t) + d(v, s_j(v)))w(s_j(v)) \leqslant (3\beta + 1)w(e_j) \leqslant (3\beta + 1)OPT$.

For the second case, if $(v, r) \in G_j$ and $(u, r) \in H_j$, as shown in the right of Fig. 5, then $w(v, r) \leqslant w(e_j)$, and there exists a vertex t with $w(u, t) \leqslant w(e_j)$ and $w(r, t) \leqslant w(e_j)$. Noting $w(v, s_j(v)) \leqslant w(e_j)$, we can also estimate the weighted distance $w(u, s_j(v))$ by $w(u, s_j(v)) \leqslant (d(u, t) + d(r, t) + d(v, r) + d(v, s_j(v)))w(s_j(v)) \leqslant (3\beta + 1)w(e_j) \leqslant (3\beta + 1)OPT$.

Noting that $v \in I(H'_j)$ implies $s_j(v) \in U'_j$, we obtain that U'_j gives at most k cost centers with at most $(3\beta + 1)OPT$ weighted distance. □

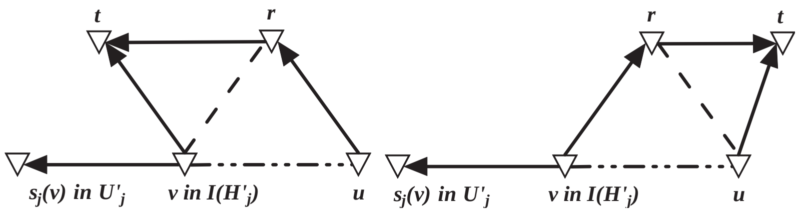

Fig. 5. Diagram for the proof of Theorem 8: two cases if the edge (u, v) is in H'_j.

In addition, for the *q*-coverage *k*-center problem with cost only, Algorithm 7 has an approximation factor of 4 when $\beta = 1$.

5. *q*-Coverage *k*-supplier problems

The *q*-coverage *k*-supplier problem partitions the vertex set *V* into the supplier set *S* and the demand set *D* that are disjoint. Hence, at most *k* centers need be selected from *S*, to minimize the distance within which all the vertices in set *D* are covered by centers each of which must cover at least *q* suppliers in *D*. In order to determine its hardness, we present the following theorem which can be proved by a reduction of *Minimum Cover* problem [4].

Theorem 9. *Given any fixed non-negative integer q, there is no* $(3 - \varepsilon)$*-approximation algorithm for the q-coverage k-center problem, unless* $\mathcal{NP} = \mathcal{P}$.

Proof. See Appendix A.

The best possible approximation factor of 3 can be achieved for the *q*-coverage *k*-supplier problem, even for its weighted extension and its cost extension. In the rest of this section, we provide a 3-approximation algorithm for the weighted case first which is applicable for the basic case by specifying $w(u) = 1$ for each supplier $u \in S$. Then, we design a $(2\beta + 1)$-approximation algorithm for the weighted and cost case, which ensures a factor of 3 for the cost only case when $\beta = 1$.

5.1. *Any q with weights*

The approximation approach is formulated in Algorithm 8. As before, edges are sorted non-decreasingly, i.e., $w(e_1) \leqslant w(e_2) \leqslant \cdots \leqslant w(e_m)$. We have subgraphs $G_1, G_2, ..., G_m$, where $G_i = (V, E_i)$, $V = S \cup D$ and $E_i = \{e_1, ..., e_i\}$. To obtain the threshold index, a new graph L_i is constructed on the demand set *D* for each G_i as follows. For each two demands $u, v \in D$, where *u* may equal to v, an edge (u, v) is in L_i if and only if there exists a supplier $r \in S$ with both (u, r) and (v, r) in G_i. Hence, self-loops of all the vertices in *V* are still in L_i. Let $I(L_i)$ denote a maximal independent set of L_i. We find *j* to be the threshold index, which is the smallest index *i*, with $deg(v) \geqslant q$ in L_i for $v \in D$, and $|I(L_i)|$

$\leqslant k$. Since i^*, the edge index of the optimal solution satisfies the above two conditions, and no two demands in $I(L_i)$ have edges from the same supplier in G_i for $1 \leqslant i \leqslant m$, we have $j \leqslant i^*$ leading to $w(e_j) \leqslant OPT$.

We shift each demand $v \in I(L_j)$ to its cheapest supplier $g_j(v)$ with the lowest weight among suppliers having an edge from v in G_j. This forms the center set $U = \{g_j(v) | v \in I(L_j)\}$, which provides at most k centers with at most a $3 \cdot OPT$ weighted distance. To see this, we prove the following theorem.

Algorithm 8 Weighted *q*-coverage *k*-supplier

1: Sort edges so that $w(e_1) \leqslant w(e_2) \leqslant \cdots \leqslant w(e_m)$, and construct $L_1, L_2, \ldots, L_m$.
2: Compute a maximal independent set, $I(L_i)$, in each graph L_i, where $1 \leqslant i \leqslant m$.
3: Find j, denoting the smallest index i, such that $|I(L_i)| \leqslant k$, and that for each vertex $v \in D$, its degree $\deg(v) \geqslant q$ in L_i.
4: For each demand $v \in I(L_j)$, let $g_j(v)$ denote the lowest weighted supplier among all suppliers $u \in S$ with an edge $(v, u) \in G_j$.
5: Let $U = \{g_j(v) | v \in I(L_j)\}$.
6: Return U.

Theorem 10. *Algorithm* 8 *has an approximation factor of* 3 *for the weighted q-coverage k-center problem.*

Proof. Note $|U| \leqslant |I(L_j)| \leqslant k$ and $w(e_j) \leqslant OPT$. To obtain the approximation factor of 3, we need only show the following two facts:

(1) each demand $u \in D$ is covered by a vertex in U within at most $3w(e_j)$;
(2) each supplier $g_j(v) \in U$, where $v \in I(L_j)$, covers at least q demands in D within at most $3w(e_j)$.

To show fact 1, consider each demand $u \in D$. As shown in Fig. 6, since $I(L_j)$ is a maximal independent set of L_j, there exists a vertex $v \in I(L_j)$ which has an edge (u, v) in L_j. This implies there is a supplier $r \in S$ with (u, r) and (v, r) in G_j. So both $w(u, r)$ and $w(v, r)$ are not more than $w(e_j)$. Noting $w(g_j(v)) \leqslant w(r)$ and $w(v, g_j(v)) \leqslant w(e_j)$, we can estimate the weighted distance from u to $g_j(v) \in U$ by $w(u, g_j(v)) \leqslant (d(u, r) + d(v, r) + d(v, g_j(v)))w(g_j(v)) \leqslant 3w(e_j)$.

Fact 2 is verified since for each supplier $g_j(v) \in U$ where $v \in I(L_j)$, the degree of v is at least q in L_j. Hence, at least q demands, equivalent or adjacent to v, are covered by $g_j(v)$ within $3w(e_j)$ by the same reasons for fact 1. By $w(e_j) \leqslant OPT$, the approximation factor is 3. □

5.2. *Any q with weights and costs*

Algorithm 9 achieves an approximation factor of $(2\beta + 1)$ for the *q*-coverage *k*-supplier problem with weights and costs. When $\beta = 1$, it ensures an approximation factor of 3 for the cost only case.

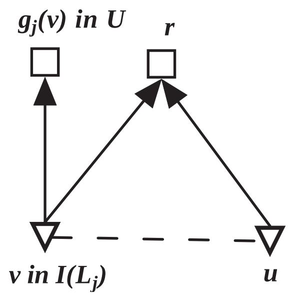

Fig. 6. Diagram for the proof of Theorem 10.

Compared with Algorithm 8, Algorithm 9 is changed as follows. After finding $I(L_i)$ for $1 \leqslant i \leqslant m$, we shift each demand $v \in I(L_i)$ to its cheapest supplier $s_i(v)$ with the lowest cost among all the suppliers having an edge from v in G_i. This forms $U_i = \{s_i(v) | v \in I(L_i)\}$. The threshold index j is the smallest index i, giving $deg(v) \geqslant q$ in L_i for $v \in D$ and the total cost of vertices in U_i, $c(U_i)$, is at most k. Since no two demands in $I(L_i)$ have edges from the same supplier in G_i for $1 \leqslant i \leqslant m$, we have $w(e_j) \leqslant OPT$.

Now, we prove that U_j have at most k cost centers within at most $(2\beta+1) \cdot OPT$ weighted distance to establish the following theorem.

Theorem 11. *Algorithm* 9 *gives an approximation factor of* $(2\beta + 1)$ *for the weighted and cost q-coverage k-center problem*.

Proof. By similar arguments in Theorem 10, the following two facts can be derived. On one hand, for each demander $u \in D$, there exists a vertex $v \in I(L_j)$ with $(u, v) \in L_j$. It is not hard to see that the weighted distance from u to $s_j(v) \in U_j$ is at most $(2\beta + 1)w(e_j)$. On the other hand, each center $s_j(v) \in U_j$, where $v \in I(L_j)$ and $deg(v) \geqslant q$, can cover at least q vertices, which are equivalent or adjacent to v in L_j, within $(2\beta+1)w(e_j)$ weighted distance. Recalling that the total cost of U_j is at most k and that $w(e_j) \leqslant OPT$, we find that the approximation factor is $2\beta + 1$. $\square$

Algorithm 9 Weighted q-coverage k-supplier

1: Sort edges so that $w(e_1) \leqslant w(e_2) \leqslant \cdots \leqslant w(e_m)$, and construct $L_1, L_2, \ldots, L_m$.
2: Compute a maximal independent set, $I(L_i)$, in each graph L_i, where $1 \leqslant i \leqslant m$.
3: For each demand $v \in I(L_i)$, let $s_i(v)$ denote the cheapest cost supplier among all vertices $u \in S$ with edge $(v, u) \in G_i$.
4: Let $U_i = \{s_i(v) | v \in I(L_i)\}$ for $1 \leqslant i \leqslant m$.
5: Find j, denoting the smallest index i, such that $c(U_i) \leqslant k$, and that for each vertex $v \in D$, $deg(v) \geqslant q$ in L_i.
6: Return U_j.

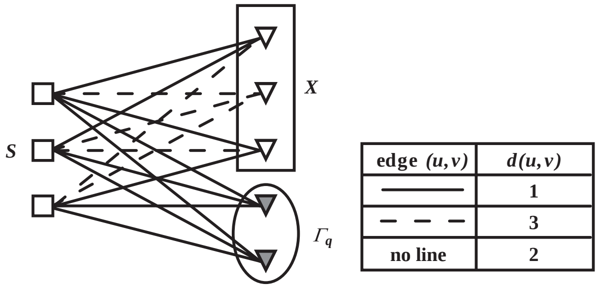

Fig. 7. Reduction from *Minimum Cover* to the q-coverage k-supplier problem.

6. Conclusion

We studied a new k-center problem which ensures minimum coverage of clients by centers. The problem is motivated by the need to balance services provided by centers while allowing centers to be utilized fully. We considered three variants of the problem. Besides in-approximation hardness, we provided approximation algorithms for the basic and generalized cases. The approximation factors found are close to or exactly at the best possible. Future work on this problem can include the consideration of the center capacities.

Appendix A. Proof of Theorem 9

Proof. Suppose there exists such an $(3-\varepsilon)$-approximation algorithm, denoted by $\mathcal{W}_q$ for a certain fixed non-negative integer q. We will show that $\mathcal{W}_q$ can solve the *Minimum Cover* [4], a well known $\mathcal{NP}$-complete problem, in polynomial time.

Minimum Cover

INSTANCE: a set $X = \{1, 2, \dots, n\}$, a collection of subsets of X: $P = \{P_1, P_2, \dots, P_m\}$, and a positive integer k.

QUESTION: Does P contain a cover for X of size k or less, i.e., a subset $P' \subseteq P$ with $|P'| \leqslant k$ such that every element of X belongs to at least one member of P'?

Given any instance of *Minimum Cover*, consider the following instance of the q-coverage k-supplier problem . Let $S = \{1, \dots, m\}$ be the supplier set. Define $\Gamma_q = \{n+1, \dots, n+q\}$ to be a set of q dummy demands. Let the demand set be $C = X \cup \Gamma_q$. For the graph $G = (V, E)$ where $V = S \cup C$, we define its edge distance as follows. For any two vertices u and v in V, if u equals to v then $d(u, v) = 0$, otherwise,

$$d(u,v) = \begin{cases} 1 & \text{if } v \in X,\ u \in S \quad \text{and } v \in P_u, \\ 3 & \text{if } v \in X,\ u \in S \quad \text{and } v \notin P_u, \\ 1 & \text{if } v \in \Gamma_q, \\ 2 & \text{otherwise.} \end{cases}$$

It is easy to verify that the distance d satisfies the triangle inequality. Fig. 7 gives an example of this reduction. Now we are going to prove that algorithm $\mathcal{W}_q$ can decide whether X has a cover with at most k subsets in P.

On one hand, if X has a cover P' with at most k subsets in P, then P' will give at most k centers within 1 distance, because the dummy demands in Γ_q is 1 distance from each supplier in S, which makes each center in P' to satisfy the q-coverage. So, applying the $(3-\varepsilon)$-approximation algorithm $\mathcal{W}_q$ on $G=(V,E)$ must provide a solution with 1 distance, since the distance between any supplier and any demand is either 1 or 3.

On the other hand, if $\mathcal{W}_q$ outputs a solution with 1 distance, then let P' be the set of subsets P_u, for at most k suppliers u selected as centers in the solution. Because any demand to a $v \in X$ is covered by a selected center u within $d(u,v)=1$, we know $v \in P_u$. By $P_u \in P'$, the set P' forms a cover of X with at most k size.

Hence, the algorithm $\mathcal{W}_q$ can solve the $\mathcal{NP}$-complete *Minimum Cover*, by verifying whether or not its output is one, leading to a contradiction. $\square$

References

[1] J. Bar-Ilan, G. Kortsarz, D. Peleg, How to allocate network centers, J. Algorithms 15 (3) (1993) 385–415.

[2] S. Chaudhuri, N. Garg, R. Ravie, The p-neighbor k-center problem, Inform. Process. Lett. 65 (3) (1998) 131–134.

[3] M. Dyer, A.M. Frieze, A simple heuristic for the p-center problem, Oper. Res. Lett. 3 (1985) 285–288.

[4] M.R. Garey, D.S. Johnson, Computers and Intractability—A Guide to the Theory of NP-Completeness, Freeman, San Francisco, 1979.

[5] D. Hochbaum, D.B. Shmoys, A best possible heuristic for the k-center problem, Math. Oper. Res. 10 (1985) 180–184.

[6] D. Hochbaum, D.B. Shmoys, A unified approach to approximation algorithms for bottleneck problems, J. ACM 33 (1986) 533–550.

[7] W. Hsu, G. Nemhauser, Easy and hard bottleneck location problems, Discrete Appl. Math. 1 (1979) 209–216.

[8] S. Khuller, R. Pless, Y.J. Sussmann, Fault tolerant k-center problems, Theoret. Comput. Sci. 242 (1–2) (2000) 237–245.

[9] S. Khuller, Y.J. Sussmann, The capacitated k-center problem, SIAM J. Discrete Math. 13 (3) (2000) 403–418.

[10] S.O. Krumke, On a generalization of the p-center problem, Inform. Process. Lett. 56 (2) (1995) 67–71.

[11] J. Plesnik, A heuristic for the p-center problem in graphs, Discrete Appl. Math. 17 (1987) 263–268.