Yokohama Publishers
ISSN 1349-8169 ONLINE JOURNAL

Pacific Journal of Optimization
Vol. 11, No. 4, pp. 583–592, October 2015

THE MINIMUM SPANNING SUBGRAPH PROBLEM WITH GIVEN CYCLOMATIC NUMBER*

Qin Wang† and Jinjiang Yuan

Abstract: This paper discusses the minimum spanning subgraph problem with given cyclomatic number k, where the cyclomatic number of a graph G, denoted by $\beta(G)$, is the dimension of its cycle space. For a given weighted graph $G = (V, E, w)$, the problem asks to find a spanning subgraph F of G such that $\beta(F) = k$ and $w(F)$ is as small as possible. We show that this problem is strongly NP-hard. When both G and F are required to be connected, we present a strongly polynomial-time algorithm to solve this problem. In this case, we also consider its reverse problem, which asks how to modify the weight function w under some given bounds in graph G such that the total modification cost plus the total weight of F under the new weights is minimized. A strongly polynomial-time algorithm is also proposed to solve this reverse problem.

Key words: *minimum spanning subgraph, reverse problem, cyclomatic number, polynomial time algorithm, strongly NP-hard*

Mathematics Subject Classification: *90C27*

1 Introduction

Graphs considered in this paper are finite and loopless. For a weighted graph $G = (V, E, w)$, its vertex set and edge set are denoted by $V = V(G)$ and $E = E(G)$, respectively. We use $w = (w(e) \in \mathbb{R} : e \in E(G))$ to denote the real weight vector defined on $E(G)$. For a subgraph T of G, the weight of T, denoted by $w(T)$, is defined by $w(T) = \sum_{e \in E(T)} w(e)$. The cyclomatic number [2] of graph G, denoted by $\beta(G)$, is the dimension of its cycle space and equals $|E(G)| - |V(G)| + c(G)$, where $c(G)$ is the number of the connected components of G. Throughout the paper, for a set F, we use $|F|$ to represent its cardinality. Readers can refer to [2] for other graph theory terms not defined here.

The cyclomatic number is an important software measurement to indicate the complexity of a program and it has been intensively studied (see, for example, [1, 3, 4, 6]). In this paper, we consider the minimum spanning subgraph problem with cyclomatic number k, which includes the following two kinds of problems.

MSS-CN-k problem (Minimum spanning subgraph problem with cyclomatic number k): Find a spanning subgraph F of a weighted graph $G = (V, E, w)$ such that $\beta(F) = k \leq$

*The first author was supported by NSFC (11171316) and Zhejiang Provincial Natural Science Foundation of China under Grant No. LQ14A010014, and the second author was supported by NSFC (11271338).

†Corresponding Author.

$\beta(G)$ and $w(F)$ is as small as possible.

When both G and F are required to be connected, we obtain the following problem.

MCSS-CN-k problem (Minimum connected spanning subgraph problem with cyclomatic number k): Find a connected spanning subgraph F of a connected weighted graph $G = (V, E, w)$ such that $\beta(F) = k \le \beta(G)$ and $w(F)$ is as small as possible.

When $k = 0$, the MCSS-CN-k problem is equivalent to the minimum spanning tree problem which is well solved.

In this paper, we show that the MSS-CN-k problem is strongly NP-hard even when the weight of each edge is unit. We give two optimality conditions for the MCSS-CN-k problem and present a strongly polynomial-time algorithm for solving it. We also consider a model of reverse MCSS-CN-k problem as follows.

Let $\mathcal{F}$ be the family of all connected spanning subgraphs with cyclomatic number k of graph G and let $c = (c(e) \in \mathbb{R}_+ : e \in E(G))$ be the given cost vector. For each $e \in E$, $c(e)$ stands for the cost of modifying (increasing or decreasing) $w(e)$ by one unit. Let $b \in \mathbb{R}_+^E$ be a bounding vector of maximum allowable modifications (increases or reductions). Let $x \in \mathbb{R}^E$ be such that $w(e) - b(e) \le x(e) \le w(e) + b(e)$ for all $e \in E$. We call x an adjusted weight vector. The reverse MCSS-CN-k problem considered in this paper can be described as follows.

R-MCSS-CN-k problem (Reverse MCSS-CN-k problem): Find an adjusted weight vector x such that
(a) $0 \le |w(e) - x(e)| \le b(e)$ for each $e \in E$, and
(b) the total cost $f(x) = \sum_{e \in E}(c(e)|w(e) - x(e)|) + \min_{F \in \mathcal{F}} \sum_{e \in F} x(e)$ is minimum.

Reverse problems play an important role in practice and have been intensively investigated in the literature. Here the term "reverse" was initially suggested by Zhang et al. in [8, 9, 10]. In this paper, we will show that the R-MCSS-CN-k problem is equivalent to that of weight reduction, that is, there will be no weight increasing when modifying the edge weights. As a result, we develop a strongly polynomial-time algorithm for solving this problem.

The paper is organized as follows. In Section 2, we show that the MSS-CN-k problem is strongly NP-hard by transforming the clique problem into this model. We present strongly polynomial-time algorithms for the MCSS-CN-k problem and for the R-MCSS-CN-k problem in Section 3 and Section 4, respectively. Some concluding remarks are given in Section 5.

2 NP-Hardness of the MSS-CN-k Problem

In this section we will show that the MSS-CN-k problem is strongly NP-hard even when the weight of each edge is unit. We will use the strongly NP-hard clique problem [5] for the reduction.

For a simple graph G, a vertex subset $X \subseteq V(G)$ is called a clique of G if its induced subgraph $G[X]$ is complete. The decision version of the clique problem can be described as: for a given simple graph G and a positive integer r with $r \le |V(G)|$, is there a clique X of G such that $|X| = r$?

Theorem 2.1. *The MSS-CN-k problem is strongly NP-hard.*

Proof. Let (G, r) be an instance of the clique problem, where G is a simple graph and r is a positive integer with $r \le |V(G)|$. Without loss of generality, suppose $r \ge 3$. We construct an instance of the decision version of the MSS-CN-k problem as follows. The graph in consideration is still G. Define the weight $w(e) = 1$ for each edge $e \in E(G)$. The value of k is defined by $k = (r-1)(r-2)/2 \ge 1$. The threshold value is defined by $Y = r(r-1)/2$. The decision version of the MSS-CN-k problem asks whether there is a spanning subgraph F of G such that $\beta(F) = k$ and $w(F) \le Y$. Note that this is equivalently to ask whether there is a subgraph F of G such that $\beta(F) = k$ and $|E(F)| \le Y$.

First, it is trivial to see that this decision problem is in NP. It can also be observed that the above construction can be done in polynomial time. In the following, we show that the instance of the clique problem has a solution if and only if there is a subgraph F of G such that $\beta(F) = k$ and $|E(F)| \le Y$.

Suppose first that the instance of the clique problem has a solution. Then there is a clique X of G such that $|X| = r$. Define $F = G[X]$. Then $|E(F)| = r(r-1)/2 = Y$ and $\beta(F) = |E(F)| - |V(F)| + 1 = r(r-1)/2 - r + 1 = k$, as required.

Conversely, suppose that there is a subgraph F of G such that $\beta(F) = k$ and $|E(F)| \le Y$. Then

$$|E(F)| - |V(F)| + c(F) = k \text{ and } |E(F)| \le r(r-1)/2.$$

This implies that

$$|V(F)| \le r - 1 + c(F). \tag{2.1}$$

If F is disconnected, let $F_1, F_2, \ldots, F_c$ be the connected components of F, where $c = c(F)$. Write $n_i = |V(F_i)|$, $1 \le i \le c$. Since a tree has cyclomatic number zero, we can assume that $n_i \ge 3$ for $1 \le i \le c$. Since $|E(F_i)| \le n_i(n_i-1)/2$, we have $\beta(F_i) \le n_i(n_i-1)/2 - n_i + 1 = (n_i-1)(n_i-2)/2$. Consequently,

$$\begin{aligned} \beta(F) &= \textstyle\sum_{1 \le i \le c} \beta(F_i) \\ &\le \textstyle\sum_{1 \le i \le c} (n_i-1)(n_i-2)/2 \\ &< (r-1)(r-2)/2 \\ &= k, \end{aligned}$$

where the last inequality follows from (2.1). But this contradicts the assumption that $\beta(F) = k$. Hence, F is a connected graph with $|V(F)| \le r$. Note that $\beta(F) = k \ge 1$ means that $|V(F)| \ge 3$.

If $|V(F)| \le r-1$, then $|E(F)| \le |V(F)|(|V(F)|-1)/2 \le (r-1)(r-2)/2$. Thus

$$\beta(F) = |E(F)| - |V(F)| + 1 < (r-1)(r-2)/2 = k.$$

Again, this contradicts the assumption that $\beta(F) = k$. Hence, $|V(F)| = r$. Consequently, we have

$$|E(F)| = \beta(F) + |V(F)| - 1 = (r-1)(r-2)/2 + r - 1 = r(r-1)/2.$$

Since $|V(F)| = r$ and $|E(F)| = r(r-1)/2$, we conclude that $V(F)$ is a clique of G such that $|V(F)| = r$. The result follows. □

From the above NP-hardness proof of Theorem 2.1, we can observe that, for a graph $G = (V, E, w)$ with $w(e) = 1$ for each edge $e \in E(G)$ and for a positive integer r with $2 \le r \le |V(G)|$, F is a spanning subgraph of G with $\beta(F) = (r-1)(r-2)/2$ and $w(F) \le r(r-1)/2$ if and only if the nontrivial component of F is a clique of r vertices of G. So, the clique problem is a special case of the MSS-CN-k problem.

3 The MCSS-CN-k Problem

In this section, we consider the case where both G and F are required to be connected. Clearly, a connected spanning subgraph F of graph G has cyclomatic number $\beta(F) = k$ if and only if $|E(F)| = |V(G)| + k - 1$.

We use $C(G)$ to denote the set of the cut edges of G. We further write $\bar{C}(G) = E(G) \setminus C(G)$. For two distinct vertices u and v of G, we define

$$C(G, u, v) = \{e \in E(G) : u \text{ and } v \text{ are disconnected in } G - e\}.$$

For a connected spanning subgraph F of graph G and $e \in C(F)$, we define

$$C^*(F, e) = \{xy \in E(G) : x \in X,\ y \in Y\},$$

where X and Y are the vertex sets of the two connected components of $F - e$.

First, we have the following equivalent conditions for the MCSS-CN-k problem.

Theorem 3.1. *Let $G = (V, E, w)$ be a connected weighted graph. Let F be a connected spanning subgraph of G with $\beta(F) = k$. Then the following statements are equivalent.*

(a) *F is a minimum connected spanning subgraph of G with $\beta(F) = k$;*

(b) *for each $e \in C(F)$ and each $f \in C^*(F, e)$, we have $w(e) \le w(f)$, and for each $g \in \bar{C}(F)$ and each $h \in E(G) \setminus E(F)$, we have $w(g) \le w(h)$;*

(c) *for each $f = uv \in E(G) \setminus E(F)$ and each $e \in C(F, u, v) \cup \bar{C}(F)$, we have $w(f) \ge w(e)$.*

Proof. (a) $\Rightarrow$ (b): Suppose that condition (a) holds. If statement (b) does not hold, then either there are $e \in C(F)$ and $f \in C^*(F, e)$ such that $w(e) > w(f)$ or there are $g \in \bar{C}(F)$ and $h \in E(G) \setminus E(F)$ such that $w(g) > w(h)$. If the former occurs, then $F - e + f$ is a connected spanning subgraph of G with $\beta(F) = k$ such that $w(F - e + f) = w(F) - w(e) + w(f) < w(F)$, contradicting the assumption of (a). If the latter occurs, then $F - g + h$ is a connected spanning subgraph of G with $\beta(F) = k$ such that $w(F - g + h) = w(F) - w(g) + w(h) < w(F)$, contradicting the assumption of (a) again. Hence, statement (b) holds.

(b) $\Rightarrow$ (c): Suppose that condition (b) holds. Suppose, to the contrary, that statement (c) does not hold. Then there are $f = uv \in E(G) \setminus E(F)$ and $e \in C(F, u, v) \cup \bar{C}(F)$ such that $w(f) < w(e)$. If $e \in C(F, u, v)$, then $e \in C(F)$ and $f \in C^*(F, e)$. But then $w(e) > w(f)$, contradicting the assumption of (b). If $e \in \bar{C}(F)$, then $f \in E(G) \setminus E(F)$ such that $w(e) > w(f)$. This contradicts the assumption of (b) again. Hence, statement (c) holds.

(c) $\Rightarrow$ (a): Suppose that condition (c) holds. Suppose, to the contrary, that statement (a) does not hold. Then F is not a minimum connected spanning subgraph of G with $\beta(F) = k$. Let T be a minimum connected spanning subgraph of G with $\beta(T) = k$ such that $|E(T) \cap E(F)|$ is maximum. Then $|E(T) \setminus E(F)| > 0$. Let $f = uv \in E(T) \setminus E(F)$ such that $w(f)$ is minimum. Then $f = uv \in E(G) \setminus E(F)$. By the assumption of (c), for each $e \in C(F, u, v) \cup \bar{C}(F)$, we have $w(f) \ge w(e)$. We distinguish the following two cases.

Case 1 $f \in C(T)$. We claim that there is $e \in C(F, u, v) \cup \bar{C}(F)$ such that $e \in C^*(T, f)$. Otherwise, denote the union of (u, v)-paths in F be P_F. Then $E(P_F) \subseteq C(F, u, v) \cup \bar{C}(F)$. So, $E(P_F) \cap C^*(T, f) = \emptyset$, which means that u and v are still connected in $G - C^*(T, f)$, a contradiction. Hence, the claim holds. By picking such an edge e in the claim and setting $H = T - f + e$, we obtain a new minimum connected spanning subgraph H of G with $\beta(H) = k$. But then $|E(F) \cap E(H)| > |E(F) \cap E(T)|$. This contradicts the choice of T.

Case 2 $f \in \bar{C}(T)$. We claim that $\bar{C}(F) \setminus \bar{C}(T)$ is not empty. Otherwise, $\bar{C}(F) \subseteq \bar{C}(T)$. We use $\bar{C}(T)$ and $\bar{C}(F)$ to simply denote the edge induced subgraph of T and F induced by $\bar{C}(T)$ and $\bar{C}(F)$, respectively. Then

$$\beta(\bar{C}(T)) = \beta(T) = k \text{ and } \beta(\bar{C}(F)) = \beta(F) = k. \tag{3.1}$$

Since $f \notin \bar{C}(F)$, we have $\bar{C}(F) \subseteq \bar{C}(T) - f$. Hence,

$$\beta(\bar{C}(F)) \le \beta(\bar{C}(T) - f). \tag{3.2}$$

Since f is not a cut edge of T, it is also not a cut edge of $\bar{C}(T)$. This implies that

$$\beta(\bar{C}(T) - f) = \beta(\bar{C}(T)) - 1 = k - 1. \tag{3.3}$$

But then, (3.1), (3.2) and (3.3) imply that $k = \beta(\bar{C}(F)) \le \beta(\bar{C}(T) - f) = k - 1$, a contradiction. This completes the proof. □

Now we present an algorithm for the MCSS-CN-k problem. Suppose that $G = (V, E, w)$ is a connected weighted graph.

Algorithm 1. Find a minimum spanning tree T of $G = (V, E, w)$. Sort the edges in $E(G) \setminus E(T)$ such that $w(e_1) \le w(e_2) \le \cdots \le w(e_\varepsilon)$, where $\varepsilon = |E(G) \setminus E(T)|$. Set $F = T + \{e_1, e_2, \ldots, e_k\}$ and end the algorithm.

Lemma 3.2 ([7])**.** *Let $G = (V, E, w)$ be a connected weighted graph and let T be a spanning tree of G. Then T is a minimum spanning tree of G if and only if for each $e \in E(T)$ and each $f \in C^*(T, e)$, $w(e) \le w(f)$.*

Theorem 3.3. *Algorithm 1 correctly find a minimum connected spanning subgraph F of $G = (V, E, w)$ with $\beta(F) = k$.*

Proof. Let T, F, and $\{e_1, e_2, \ldots, e_k\}$ be the same as in Algorithm 1. Let $F_j = T + \{e_1, e_2, \ldots, e_j\}$, $0 \le j \le k$. We need only show that, for each j with $0 \le j \le k$, F_j is a minimum connected spanning subgraph of G with $\beta(F_j) = j$. The assertion holds trivially for $j = 0$ since $F_0 = T$ is a minimum spanning tree of G.

Inductively, suppose that $1 \le j \le k$ and F_{j-1} is a minimum connected spanning subgraph of G with $\beta(F_{j-1}) = j - 1$. From Theorem 3.1(b), for each $e \in C(F_{j-1})$ and each $f \in C^*(F_{j-1}, e)$, we have $w(e) \le w(f)$, and for each $g \in \bar{C}(F_{j-1})$ and each $h \in E(G) \setminus E(F_{j-1})$, we have $w(g) \le w(h)$, where the conclusion for $F_{j-1} = F_0 = T$ with $j = 1$ follows from Lemma 3.2. Since e_j has the minimum weight among the edges in $E(G) \setminus E(F_{j-1})$, the condition in Theorem 3.1(b) still holds for $F_j = F_{j-1} + e_j$. It follows from Theorem 3.1 that F_j is a minimum connected spanning subgraph of G with $\beta(F_j) = j$.

The above discussion implies that a minimum connected spanning subgraph F with cyclomatic number k can be constructed by a minimum spanning tree T and k shortest edges of $E(G) \setminus E(T)$. It follows that Algorithm 1 works correctly. This completes the proof. □

In Algorithm 1, sorting the weights of all the edges in $E(G) \setminus E(T)$ costs $O(\varepsilon \log \varepsilon)$ time which is dominated by the running time for finding the minimum spanning tree. Hence, Algorithm 1 can be performed with the same time complexity $O(|E| \log |E|)$ as the minimum spanning tree problem.

Remark. It can be verified that, when G is a connected graph, $(E, \mathcal{F})$ forms a matroid, where $\mathcal{F}$ consists of the subset $F \subseteq E$ so that $|F| \leq |V| + k - 1$ and either $G[F]$ is acyclic or $G[F]$ is spanning and connected. Then $F \subseteq E$ is a basis of $(E, \mathcal{F})$ if and only if $|F| = |V| + k - 1$ and either $G[F]$ is acyclic or $G[F]$ is spanning and connected. Thus, the greedy algorithm for finding the minimum weighted basis of $(E, \mathcal{F})$ does work for the MCSS-CN-k problem. It seems that Algorithm 1 is a concentration of the greedy algorithm in this special matroid.

4 The R-MCSS-CN-k Problem

First, we will show that the R-MCSS-CN-k problem is equivalent to that of weight reduction, that is, there will be no weight increasing when modifying the edge weights. We have

Theorem 4.1. *Suppose that $\bar{x}$ is an optimal solution of the R-MCSS-CN-k problem, then $\bar{x}(e) \leq w(e)$ for each $e \in E$.*

Proof. Suppose, to the contrary, that there is an edge $g \in E$ such that $\bar{x}(g) > w(g)$. Then clearly

$$x'(e) = \begin{cases} w(e), & \text{if } e = g \ , \\ \bar{x}(e), & \text{otherwise} \end{cases}$$

is a feasible solution of the R-MCSS-CN-k problem. Note that

$$\sum_{e \in E} (c(e)|w(e) - x'(e)|) = \sum_{e \in E} (c(e)|w(e) - \bar{x}(e)|) - c(g)(\bar{x}(g) - w(g)).$$

Since $c(g) > 0$ and $\bar{x}(g) > w(g)$, we have

$$\sum_{e \in E} (c(e)|w(e) - x'(e)|) < \sum_{e \in E} (c(e)|w(e) - \bar{x}(e)|).$$

Moreover, it is obvious that $\min_{F \in \mathcal{F}} \sum_{e \in F} x'(e) \leq \min_{F \in \mathcal{F}} \sum_{e \in F} \bar{x}(e)$. So, $f(x') < f(\bar{x})$, contradicting the assumption that $\bar{x}$ is an optimal solution of the R-MCSS-CN-k problem. Hence, we have $\bar{x}(e) \leq w(e)$ for each $e \in E$. This completes the proof. □

Based on Theorem 4.1, in the following, we need only deal with the weight reduction case of the R-MCSS-CN-k problem, and we call it the R′-MCSS-CN-k problem, which can be described formally as follows.

R′-MCSS-CN-k problem: Find an adjusted weight vector x such that
(a) $0 \leq w(e) - x(e) \leq b(e)$ for each $e \in E$, and
(b) the total cost $f(x) = \sum_{e \in E} c(e)(w(e) - x(e)) + \min_{F \in \mathcal{F}} \sum_{e \in F} x(e)$ is minimum.

For convenience, we call a weight vector x satisfying the condition (a) a feasible weight solution. The following theorem can be observed.

Theorem 4.2. *Suppose that $\bar{x}$ is an optimal solution of the R′-MCSS-CN-k problem and $\bar{F}$ is an optimal solution of the MCSS-CN-k problem* $\min_{F \in \mathcal{F}} \sum_{e \in F} \bar{x}(e)$. *Then*

$$x'(e) = \begin{cases} \bar{x}(e), & \text{if } e \in \bar{F} \ , \\ w(e), & \text{otherwise} \end{cases}$$

is also an optimal solution of the R′-MCSS-CN-k problem.

For each $e \in E$, we define

$$w^*(e) = \begin{cases} w(e) - b(e), & \text{if } c(e) < 1 \ , \\ w(e), & \text{if } c(e) \geq 1. \end{cases}$$

Let $w_1(e) = c(e)(w(e) - w^*(e)) + w^*(e)$ for each $e \in E$. We use F^1 to denote an optimal solution of the MCSS-CN-k problem $\min_{F \in \mathcal{F}} \sum_{e \in F} w_1(e)$ under the weight w_1. Moreover, we define x_1 by the following way:

$$x_1(e) = \begin{cases} w^*(e), & \text{if } e \in F^1 \ , \\ w(e), & \text{otherwise} \ . \end{cases}$$

By the definition of x_1 and w^*, it is easy to see that $0 \leq w(e) - x_1(e) \leq b(e)$ for each $e \in E$. Hence, x_1 is a feasible weight solution of the R′-MCSS-CN-k problem.

In the following we will show that x_1 is also an optimal solution of the R′-MCSS-CN-k problem. First, we have,

Theorem 4.3. *Every feasible weight solution x has the total cost*

$$f(x) \geq C = \sum_{e \in E} c(e)(w(e) - x_1(e)) + \sum_{e \in F^1} x_1(e).$$

Proof. Suppose $\bar{x}$ is an optimal weight solution of the R′-MCSS-CN-k problem and $F^2 \in \mathcal{F}$ is an optimal solution of the MCSS-CN-k problem $\min_{F \in \mathcal{F}} \sum_{e \in F} \bar{x}(e)$. Define a new weight solution x' by setting

$$x'(e) = \begin{cases} \bar{x}(e), & \text{if } e \in F^2 \ , \\ w(e), & \text{otherwise} \ . \end{cases}$$

By Theorem 4.2, x' is also an optimal weight solution. Note that $f(x') = \sum_{e \in E} c(e)(w(e) - x'(e)) + \sum_{e \in F^2} x'(e)$. Moreover, we have

$$\begin{aligned} C &= \sum_{e \in E} c(e)(w(e) - x_1(e)) + \sum_{e \in F^1} x_1(e) \\ &= \sum_{e \in F^1} c(e)(w(e) - w^*(e)) + \sum_{e \in F^1} w^*(e) \\ &= \sum_{e \in F^1} w_1(e). \end{aligned} \tag{4.1}$$

From the definition of w_1 and F^1, we have

$$\sum_{e \in F^1} w_1(e) \leq \sum_{e \in F^2} w_1(e) \leq \sum_{e \in F^2} c(e)(w(e) - x'(e)) + \sum_{e \in F^2} x'(e) = f(x'). \tag{4.2}$$

Since $f(x')$ is the optimal value of the R′-MCSS-CN-k problem, from (4.1) and (4.2), we have

$$C \leq f(x') \leq f(x).$$

The result follows. □

In the following, we need only show that F^1 is the optimal solution of the MCSS-CN-k problem under weight vector x_1.

Theorem 4.4. *F^1 is the optimal solution of the MCSS-CN-k problem under weight vector x_1.*

Proof. By contradiction. Suppose that F^1 is not optimal and let $\bar{F} \in \mathcal{F}$ be an optimal solution of the MCSS-CN-k problem under weight vector x_1. Then $\sum_{e\in F^1} x_1(e) > \sum_{e\in \bar{F}} x_1(e)$. Define weight solution x_2 by:

$$x_2(e) = \begin{cases} x_1(e), & \text{if } e \in \bar{F}, \\ w(e), & \text{otherwise.} \end{cases}$$

Then the total cost of G under weight vector x_2 is given by

$$\begin{aligned} f(x_2) &= \textstyle\sum_{e\in F^1\cap\bar{F}} c(e)(w(e) - x_1(e)) + \sum_{e\in\bar{F}} x_1(e) \\ &< \textstyle\sum_{e\in F^1} c(e)(w(e) - x_1(e)) + \sum_{e\in F^1} x_1(e) \\ &= C, \end{aligned}$$

contradicting Theorem 4.3. The proof is completed. □

Combining Theorem 4.4 with Theorem 4.3 we conclude that weight vector x_1 is an optimal solution of the R′-MCSS-CN-k problem. Now we outline the strongly polynomial-time solution procedures for the R′-MCSS-CN-k problem in the following.

Algorithm 2. Step 1 Define the revised weight vector w_1 by

$$w_1(e) = \begin{cases} c(e)b(e) + w(e) - b(e), & \text{if } c(e) < 1, \\ w(e), & \text{if } c(e) \geq 1, \end{cases}$$

for each $e \in E$.
Step 2 Use Algorithm 1 to solve the MCSS-CN-k problem $\min_{F\in\mathcal{F}} \sum_{e\in F} w_1(e)$ and denote the optimal solution by F^1.
Step 3 Construct the adjusted weight vector x by

$$x(e) = \begin{cases} w(e) - b(e), & \text{if } c(e) < 1 \text{ and } e \in F^1, \\ w(e), & \text{otherwise.} \end{cases}$$

for each $e \in E$. Then x is the optimal solution and the optimal value is $f(x)$.

The running time used in Step 1 for computing the new weights $w_1(e)$, $e \in E$, is $O(|E|)$. Solving the MCSS-CN-k problem in Step 2 by Algorithm 1 takes $O(|E|\log|E|)$ time. The running time used in Step 3 for computing the adjusted weights $x(e)$,$e \in E$, is also $O(|E|)$. So the time complexity of Algorithm 2 is given by $O(|E|\log|E|)$.

5 Concluding Remarks

We have proved that the MSS-CN-k problem is strongly NP-hard and both the MCSS-CN-k problem and the R-MCSS-CN-k problem can be solved in strongly polynomial times. In fact, Algorithm 2 can be applied to solve many reverse optimization problems when $\mathcal{F}$ is the family satisfying the following Assumption 5.1.

Assumption 5.1. We assume that the problem

$$\text{(A1):} \quad \min_{F\in\mathcal{F}} w(F) = \min_{F\in\mathcal{F}} \sum_{e\in F} w(e)$$

can be solved in polynomial time.

There are many cases in which the assumption is satisfied. For example, when $\mathcal{F}$ is the family of all spanning trees, or all matchings of a graph, or all perfect matchings of a complete bipartite graph $G = (X, Y; E)$ with $|X| = |Y|$, then problem (A1) becomes the well-known minimum spanning tree problem, maximum weighted matching problem, or assignment problem, and so, there are strongly polynomial-time algorithms to solve these problems. So, under Assumption 5.1, Algorithm 2 can solve a large class of such kind of reverse optimization problems in polynomial time.

We can also consider other kinds of reverse problems, for example, in the model of the R′-MCSS-CN-k problem, the modification bounds are on each edge of G and they are not affecting each other. If we give an overall budget $B > 0$ on the total modification, we have the following R″-MCSS-CN-k problem.

R″-MCSS-CN-k problem Find an adjusted weight vector x such that
(a) $0 \le w(e) - x(e) \le b(e)$ for each $e \in E$,
(b) $\sum_{e\in E}(w(e) - x(e)) \le B$, and
(c) the total cost $f(x) = \sum_{e\in E} c(e)(w(e) - x(e)) + \min_{F\in\mathcal{F}} \sum_{e\in F} x(e)$ is minimum.

In further research, we can study the complexity of the R″-MCSS-CN-k problem. It seems that this problem is also polynomially solvable. Moreover, it may be meaningful to consider whether other relevant models under other norms are polynomially solvable.

Acknowledgments

The authors would like to thank the associate editor and the anonymous reviewers for their constructive comments and helpful suggestions.

References

[1] G.L. Chia, W. Hemakul and S. Singhun, Graphs with cyclomatic number two having panconnected square, *Discrete Math.* 311 (2011) 850–855.

[2] R. Diestel, *Graph Theory*, third edition, Springer-Verlag, Berlin Heidelberg, 2005.

[3] A.A. Dobrynin and L.S. Mel'nikov, Wiener index for graphs and their line graphs with arbitrary large cyclomatic numbers, *Appl. Math. Lett.* 18 (2005) 307–312.

[4] A. Dress, S. Grunewald and D. Stevanovic, Semiharmonic graphs with fixed cyclomatic number, *Appl. Math. Lett.* 17 (2004) 623–629.

[5] M.R. Garey and D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freeman and Company, New York, 1979.

[6] J. Harant, D. Rautenbach, P. Recht and F. Regen, Packing edge-disjoint cycles in graphs and the cyclomatic number, *Discrete Math.* 310 (2010) 1456–1462.

[7] B. Korte and J. Vygen, *Combinatorial Optimization, Theory and Algorithms*, fourth edition, Springer-Verlag, Berlin, Heidelberg, 2007.

[8] J.Z. Zhang and Y.X. Lin, Computation of the reverse shortest-path problem, *J. Global Optim.* 25 (2003) 243–261.

[9] J.Z. Zhang, Z.H. Liu and Z.F. Ma, Some reverse location problem, *European J. Oper. Res.* 124 (2000) 77–88.

[10] J.Z. Zhang, X.G. Yang and M.C. Cai, Reverse center location problem, *Lecture Notes in Comput. Sci.* 1741 (1999) 279–294.

Manuscript received 13 August 2014
revised 14 February 2015
accepted for publication 14 February 2015

Qin Wang
College of Sciences, China Jiliang University
Hangzhou, Zhejiang 310018, People's Republic of China
E-mail address: wq@cjlu.edu.cn

Jinjiang Yuan
School of Mathematics and Statistics, Zhengzhou University
Zhengzhou, Henan 450001, People's Republic of China
E-mail address: yuanjj@zzu.edu.cn