Qin WANG
China Jiliang University

Partial Inverse Most Unbalanced Spanning Tree Problem

***Abstract**. In this paper, we consider the partial inverse most unbalanced spanning tree problem, which is how to modify the weights of the edges in a simple undirected weighted graph with minimum cost such that the partially given forest is contained in a new most unbalanced spanning tree. Two models are studied: the problem under the weighted Hamming distance and the problem under the weighted l_1 norm. We present their respective algorithms that all run in strongly polynomial times.*

***Streszczenie.** Rozważano częściowo odwrotny najbardziej niezrównoważony problem drzewa rozpinającego, czyli jak modyfikować wagi brzegów niebezpośrednio ważonego grafu. Rozpatrzono dwa modele: ważonego dystansu Hamminga i ważonej normy l1. (**Częściowo odwrotny najbardziej niezrównoważony problem drzewa rozpinającego**)*

Keywords: Most unbalanced spanning tree, Forest, Computational complexity, Strongly polynomial time algorithm.
Słowa kluczowe: grafy, drzewo rozpinające.

Introduction

The inverse optimization problem is to modify some parameters of the original problem, with minimum modification cost, such that some given feasible solutions become the optimal ones under the new parameters. Inverse optimization problem has attracted an increasing interest during the last several years and many applications have been found in the real world. (For example, see[1, 2].) Readers may refer to the survey paper [3] and the papers cited therein for a comprehensive review on the development in inverse optimization.

In reality, obtaining the complete information about all variables of an optimization problem is difficult, and we may have only a partial solution instead of a full solution. Therefore, the inverse optimization problems can be further extended to more general models in which the purpose to revise the parameters with minimum cost is to let the resulting optimal solution contain a given partial solution. This type of problem is called a partial inverse optimization problem. Partial inverse optimization problems are more difficult and more realistic than normal inverse optimization problems. Some publications on partial inverse optimization problems can be found in the literature [2, 4, 5, 6, 7, 8, 9]. Only the partial inverse minimum assignment problem under l_1 norm without bound constraints, the partial inverse maximum weight closure problem under the l_1 norm, the partial inverse sorting problems under l_1, l_2 and l_∞ norms and the partial inverse minimum spanning tree problem under the constraint that edge weights can not be increased are known to be polynomially solvable. In this paper, we show that the partial inverse most unbalanced spanning tree problem under weighted Hamming distance and weighted l_1 norm can be solved efficiently by strongly polynomial time algorithms.

Given a finite set E, a family $\mathbb{F} \subseteq 2^E$ of subsets of E and a real weight vector $w(e)$ associated with every $e \in E$, we consider the problem of finding a subset $F \in \mathbb{F}$ for which the amount $\max_{e\in F} w(e) - \min_{e\in F} w(e)$ is minimized or maximized. We call this problem the most balanced or the most unbalanced optimization problem, respectively. As a special case, the most balanced and the most unbalanced spanning tree problems have been investigated in [10]. [11] presented improved strongly polynomial time algorithms for a particular problem. [12] investigated some most balanced combinatorial optimization problems with a linear side constraint. [13] presented an inverse model for the most balanced problem and showed that this model can be solved in polynomial time whenever an associated min-sum problem can be solved in polynomial time. Some applications of the most balanced and the most unbalanced optimization problems have been presented in [10].

The paper is organized as follows. In section 2, we introduce the partial inverse most unbalanced spanning tree problem under weighted Hamming distance and we present a strongly polynomial time algorithm to solve this problem. In section 3, we give some properties of the partial inverse most unbalanced spanning tree problem under weighted l_1 norm and show that it is also strongly polynomially solvable. Some conclusions are made in section 4.

The problem under weighted Hamming distance

Given an undirected graph $G=(V, E, w)$ where V is the vertex set and E is the edge set of G with $|V| = n$ and $|E| = m$, and $w(e_i)$ denotes the weight of the edge $e_i \in E$. For any spanning tree T of G we define the degree of balance of T as $B(T) = \max_{e\in T} w(e) - \min_{e\in T} w(e)$. The most unbalanced spanning tree problem is to find a spanning tree T^* such that $D_{T^*} = \max\{D_T \mid T \in G\}$. We call T^* a most unbalanced spanning tree of G. Let F be a forest in G (see Fig. 1) and $c(e_i)$ be the weight modification cost on $e_i \in E$. Suppose that the the new weight of $e_i \in E$ is $w^*(e_i)$. Then for the partial inverse most unbalanced spanning tree problem under weighted Hamming distance, we look for a new edge weight vector w^* such that

(a) under w^* a most unbalanced spanning tree T^* exists such that it contains F;

(b) the total modification cost $\sum_{e_i\in E} c(e_i) H(w^*(e_i), w(e_i))$ is minimized, where $H(w^*(e_i), w(e_i))$ is the Hamming distance between $w^*(e_i)$ and $w(e_i)$, i.e., $H(w^*(e_i), w(e_i)) = 0$ if $w^*(e_i) = w(e_i)$ and 1 otherwise.

Denote (T^*, w^*) be an optimal solution of this problem.

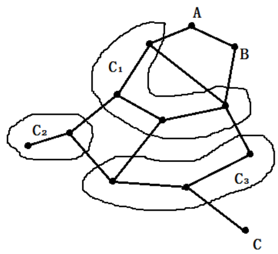

Fig.1. Graph G and the forest $F = C_1 \cup C_2 \cup C_3$

Suppose that there are *k* components of *F* denoted by C_i, $i=1,...,k$ (see Fig. 1). Contract each component C_i of *F* into a single vertex, denote the resulting graph by G' (see Fig. 2). Obviously, G' is connected since *G* is connected.

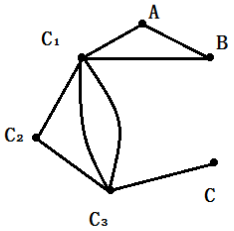

Fig.2. Graph G'

We have the following Theorem:

Theorem 1 There exists an optimal solution w^* such that for any edge $e_i \in E$, $w^*(e_i) \in \{w(e_j) \mid e_j \in E\}$.

Proof. It is easy to see that there exists an optimal solution (T^*, w^*) such that $\min\{w(e_j) \mid e_j \in E\} \le w^*(e_i) \le \max\{w(e_j) \mid e_j \in E\}$ for each $e_i \in E$, for otherwise, we can modify $w^*(e_i)$ to $\min\{w(e_j) \mid e_j \in E\}$ or $\max\{w(e_j) \mid e_j \in E\}$, then T^* is still a most unbalanced spanning tree containing *F* and the total modification cost is minimum. Suppose that there exists $w^*(e_i)$ such that $w(e_l) < w^*(e_i) < w(e_k)$, where $w(e_l) = \max\{w(e_j) \mid w(e_j) < w^*(e_i)\}$ and $w(e_k) = \min\{w(e_j) \mid w(e_j) > w^*(e_i)\}$. Then we can set $w^*(e_i) = w(e_l)$ when $w(e_i) > w^*(e_i)$ and set $w^*(e_i) = w(e_k)$ when $w(e_i) < w^*(e_i)$. The total modification cost is still minimum. This completes the proof of Theorem 1. ■

Denote the maximum edge weight and the minimum edge weight of *G* under *w* be $w_{G_L} = \max\{w(e_j) \mid e_j \in E\}$ and $w_{G_S} = \min\{w(e_j) \mid e_j \in E\}$, respectively. And denote $I = \{(v_i, v_j) \mid (v_i, v_j) \in G \setminus E(F), v_i, v_j \in C_l, l = 1,...,k\}$. By Theorem 1, for each pair (w_i, w_j), $w_i \le w_j$, let $w^*_{G_L} = w_j$, $w^*_{G_S} = w_i$. Then we modify the edge weights according to each pair and choose the one with the minimum modification cost. Note that there are at most $|E(G)|(|E(G)|+1)/2$ such pairs. For the pair (w_i, w_j), we have the following algorithm:

Algorithm 1

Step 1. First we increase the weight of each edge in $\{e_i \in I \mid w(e_i) < w_i\}$ to w_i, and decrease the weight of each edge in $\{e_i \in I \mid w(e_i) > w_j\}$ to w_j. Denote $E_i = \{e_t \in G \setminus I \mid w(e_t) = w_i\}$ and $E_j = \{e_t \in G \setminus I \mid w(e_t) = w_j\}$. Go to step 2.

Step 2. If there exists $e \in E_i$, $f \in E_j$ such that $F \bigcup \{e, f\}$ contains no cycle, then increase the weight of each edge in $\{e_i \in G \setminus I \mid w(e_i) < w_i\}$ to w_i, and decrease the weight of each edge in $\{e_i \in G \setminus I \mid w(e_i) > w_j\}$ to w_j. Otherwise, go to step 3.

Step 3. If for any $e \in E_i$, $f \in E_j$, $F \bigcup \{e, f\}$ contains a cycle, then $F \bigcap (E_i \bigcup E_j) = \varnothing$. Denote $E_{ij} = \{e_i \in G \setminus I \mid w(e_i) < w_i\} \bigcup \{e_i \in G \setminus I \mid w(e_i) > w_j\}$.

If there exists two edges $u, v \in E_{ij}$ such that $F \bigcup \{u, v\}$ contains no cycle, then let $w^*(u) = w_i$, $w^*(v) = w_j$; If there are two edges $u \in E_{ij}$ and $v \in E_i \bigcup E_j$ such that $F \bigcup \{u, v\}$ contains no cycle, without loss of generality, suppose $v \in E_i$, then let $w^*(u) = w_j$; If for any $u, v \in E_{ij} \bigcup E_i \bigcup E_j$, $F \bigcup \{u, v\}$ contains a cycle, then sort the cost of each edge of $G \setminus (I \bigcup E_{ij} \bigcup E_i \bigcup E_j)$ in increasing order. Denote *s* by the first edge satisfying that there exists $t \in E_{ij} \bigcup E_i \bigcup E_j$ such that $F \bigcup \{s, t\}$ contains no cycle. Without loss of generality, suppose $t \in E_{ij}$, then let $w^*(s) = w_j$, $w^*(t) = w_i$, go to step 6; If $E_{ij} = \varnothing$, then sort the cost of each edge of $G \setminus (I \bigcup E_i \bigcup E_j)$ in increasing order. Denote *u* by the first edge satisfying that there exists $v \in E_i \bigcup E_j$ such that $F \bigcup \{u, v\}$ contains no cycle. Then $c(u)$ is the optimal value in this phase, go to step 6. Finally, set the weights of the edges in E_{ij} to w_i or w_j. Otherwise, go to step 4.

Step 4. $E_i = \varnothing$ or $E_j = \varnothing$. Without loss of generality, suppose $E_i = \varnothing$. For each edge $f \in E_j$, sort the cost of each edge of E_{ij} in increasing order, and sort the cost of each edge of $G \setminus (I \bigcup E_{ij} \bigcup \{f\})$ in increasing order and put them at the end of the sorted edges in E_{ij}. Denote by *u* the first edge such that $F \bigcup \{u, f\}$ contains no cycle. Let $w^*(u) = w_i$, and change the weights of the edges in E_{ij} to w_i or w_j. Go to step 6. Otherwise, go to step 5.

Step 5. $E_i = E_j = \varnothing$. If there exists two edges $u, v \in E_{ij}$ such that $F \bigcup \{u, v\}$ contains no cycle, then let $w^*(u) = w_i$, $w^*(v) = w_j$; If $|E_{ij}| = 1$ or for any $u, v \in E_{ij}$, $F \bigcup \{u, v\}$ contains a cycle, then for each edge $f \in E_{ij}$, sort the cost of each edge of $G \setminus (I \bigcup E_{ij})$ in increasing order. Denote *u* by the first edge such that $F \bigcup \{u, f\}$ contains no cycle. Let $w^*(u) = w_i$, $w^*(f) = w_j$, go to step 6; If $E_{ij} = \varnothing$, go to step 6.

Step 6. Set the weights of the edges in E_{ij} to w_i or w_j. Denote *u* and *v* by the two edges of $G \setminus (I \bigcup E_{ij} \bigcup E_i \bigcup E_j)$ with minimum modification cost and such that $F \bigcup \{u, v\}$ contains no cycle. Let $w^*(u) = w_i$, $w^*(v) = w_j$. Compare the costs and find the minimum one.

Theorem 2 Algorithm 1 solves the problem under weighted Hamming distance in strongly polynomially times.

Proof. The correctness of Algorithm 1 is indicated in the above analysis. The computation in step 1 is $O(|E(G)|)$. The computation in step 2 to step 6 is $O(|E(G)|^2)$. Because there are at most $|E(G)|(|E(G)|+1)/2$ pairs to be compared, we obtain that the total complexity of Algorithm 1 is $O(|E(G)|^4)$. The proof is completed. ■

The problem under weighted l_1 norm

Let *G*=(*V*, *E*, *w*) be a simple connected undirected weighted graph with $|V| = n$ and $|E| = m$, where *w* denotes the weight vector defined on *E*. Let *F* be a forest in *G* and

c_1 and $c_2 : E \to R^+$ be two cost vectors. For each $e \in E$, $c_1(e)$ stands for the cost of reducing $w(e)$ by one unit, and $c_2(e)$ is the cost of increasing $w(e)$ by one unit.

Let $x=x(e)$ be a real function satisfying $\min_{e\in E} w(e) \le x(e) \le \max_{e\in E} w(e)$ for all $e \in E$, and for any spanning tree T of G we define the degree of balance of T under x as $B_x(T) = \max_{e\in T} x(e) - \min_{e\in T} x(e)$. The problem under weighted l_1 norm is to change w to w^* such that

(a) under w^* a most unbalanced spanning tree T^* exists such that it contains F;

(b) The cost $\sum_{e\in E} \max\{c_1(e)(w(e)-w^*(e)), c_2(e)(w^*(e)-w(e))\}$ is minimized.

We say that (T^*, w^*) is an optimal solution. Set

(1) $U(T,x) = \max_{e\in T} x(e), L(T,x) = \min_{e\in T} x(e).$

(2) $c(x) = \sum_{e\in E} \max\{c_1(e)(w(e)-x(e)), c_2(e)(x(e)-w(e))\}.$

We call $U(T,x)$ the max-weight of T under x, $L(T,x)$ the min-weight of T under x and $c(x)$ the total modification cost of x. It is clear that $B_x(T) = U(T,x) - L(T,x)$.

Theorem 3 Let (T,x) be an optimal solution. Let $E' = \{e \in E : L(T,x) \le w(e) \le U(T,x)\}$. Then $|E(T \setminus E')| \le 1$.

Proof. Suppose that there are two edges f_1 and f_2 in F such that $f_1 \notin E'$ and $f_2 \notin E'$. if $w(f_1) < L(T,x)$ and $w(f_2) > U(T,x)$, Let y be defined by

(3) $$y(e) = \begin{cases} w(f_1), & if\ w(e) < w(f_1), \\ w(f_2), & if\ w(e) > w(f_2), \\ w(e), & otherwise. \end{cases}$$

Then we have $B_y(T) > B_x(T)$ and $c(y) < c(x)$. Otherwise, without loss of generality, suppose that $w(f_1) > U(T,x)$, $w(f_2) > U(T,x)$ and suppose that $x(f_1) \ne L(T,x)$ or $x(f_1) = x(f_2) = L(T,x)$. The case $w(f_1) < L(T,x)$ and $w(f_2) < L(T,x)$ can be discussed similarly. From a similar discussion, we can obtain y with $U(T,y) = w(f_1)$ and $L(T,y) = L(T,x)$, such that $B_y(T) > B_x(T)$ and $c(y) < c(x)$. Either of the above analysis contradicts the fact that (T,x) is an optimal solution. If there is only one edge $f_1 \in F$ such that $f_1 \notin E'$. With a similar discussion, we know that $T \setminus \{E' \cup F\} = \varnothing$.

When $F \subseteq E'$ and $T \setminus E' \ne \varnothing$. If $|E(T \setminus E')| \ge 2$, also from a similar discussion we can get the contradiction. If $T \setminus E' = \{f_1\}$, without loss of generality, suppose that $w(f_1) > U(T,x)$. In this case, we know that $x(f_1) = L(T,x)$ for otherwise we can also get the contradiction by constructing y with $U(T,y) = w(f_1)$ and $L(T,y) = L(T,x)$. This completes the proof of Theorem 3. ■

Theorem 4 For every optimal spanning tree T, there is an optimal function x such that (T,x) is an optimal solution, and both $U(T,x)$ and $L(T,x)$ are in $\{w(e) : e \in E\}$.

Proof. Suppose that (T,y) is an optimal solution and at least one of $U(T,y)$ and $L(T,y)$ is not in $\{w(e) : e \in E\}$. Without loss of generality, suppose that $U(T,y)$ is not in $\{w(e) : e \in E\}$. Note that $U(T,y) < \max_{e\in E} w(e)$. From Theorem 3, we know that there is only one edge $f \in T$ with $w(f) < U(T,y)$ and $y(f) = U(T,y)$. Set

(4) $$C_1 = \{e \notin T : w(e) > U(T,y)\},$$

(5) $$C_2 = \{e \in E : w(e) < U(T,y)\}.$$

If $C_1 = \varnothing$, from Theorem 3, we know that there is only one edge $m \in T$ with $w(m) = \max_{e\in E} w(e)$ and $x(m) = L(T,y)$. Also by Theorem 3, we have $\max_{e\in\{E\setminus\{m\}\}} w(e) < U(T,y) < w(m)$. We obtain a new function x such that $U(T,x) = \max_{e\in\{E\setminus\{m\}\}} w(e)$ and $L(T,x) = L(T,y)$ as follows:

(6) $$x(e) = \begin{cases} U(T,x), & if\ e = f, \\ y(e), & otherwise. \end{cases}$$

Since $U(T,x) < U(T,y)$, we have $c(x) < c(y)$ and T is a most unbalanced spanning tree containing F under x. If $C_1 \ne \varnothing$, then the total reduction and expansion cost $c(y)$ of the weight function y is

(7) $$c(y) = \sum_{e\in C_1} c_1(e)(w(e) - U(T,y)) + c_2(f)(U(T,y) - w(f)) + c_L(y),$$

where $c_L(y)$ denotes the cost for modifying the lower bound of the weights of E to $L(T,y)$. Without loss of generality, we suppose that $\sum_{e\in C_1} c_1(e) \le c_2(f)$. The case $\sum_{e\in C_1} c_1(e) > c_2(f)$ can be discussed similarly. We obtain a new function x such that $U(T,x) = \max_{e\in C_2} w(e)$ and $L(T,x) = L(T,y)$ as follows:

(8) $$x(e) = \begin{cases} U(T,x), & if\ e \in C_1\ or\ \ e = f, \\ y(e), & otherwise. \end{cases}$$

Note that $B_x(T) = \max_{e\in E} x(e) - \min_{e\in E} x(e)$ and the reduction and expansion cost of x is

(9) $$c(x) = \sum_{e\in C_1} c_1(e)(w(e) - U(T,x)) + c_2(f)(U(T,x) - w(f)) + c_L(y) \le c(y).$$

Hence (T,x) is also an optimal solution. The proof is completed. ■

Sort the edges of G to non-decreasing order with $l = w(e_1) \le w(e_2) \le \cdots \le w(e_m) = u$. We have the degree of balance of the most unbalanced spanning tree of G under w is $w(e_m) - w(e_1)$. Write $q_1 = \max_{e\in F} w(e)$ and $q_2 = \min_{e\in F} w(e)$. If $l = u$ or $q_1 = u$ and $q_2 = l$, then any spanning tree containing F is an optimal solution, and we need not modify any weight of the edges of G. So suppose $l < u$ or $q_1 < u$ and $q_2 > l$ in the sequel.

First, we contract each component of F into a single vertex and denote the resulting graph by G'. Suppose that G' is a simple graph. If there are multiple edges of G', we can deal with the problems by a similar but more complicated method. Here we omit the discussions for the multi-graph case. Write $u_1 = \max_{e\in G'\cup F} w(e) = w(e_s)$ and $l_1 = \min_{e\in G'\cup F} w(e) = w(e_t)$. If $u_1 = u$ and $l_1 = l$, we are done with any spanning tree containing $F \cup \{e_s, e_t\}$. This can be done in $O(|E|\log|E|)$ time. Otherwise, by Theorem 3, for each pair of $(w(e_i), w(e_j))$ with $i, j = 1, \cdots, m$ and $w(e_i) \le w(e_j)$ we find the most unbalanced spanning tree *T* containing *F* with minimum total modification cost under the constraint that $L(T, x) = w(e_i), U(T, x) = w(e_j)$. Note that there are $|E|(|E|+1)/2$ such pairs. We describe the main steps of this procedure as follows:

Algorithm 2

Step 1. Set $E' = \{e \in E : w(e_i) \le w(e) \le w(e_j)\}$. Contract $F \cup E'$ and denote the resulting graph by G'. If one of the following three cases exists

(a) $|F \setminus E'| > 1$;

(b) $|F \setminus E'| = 1$ and $|V(G')| > 1$;

(c) $|F \setminus E'| = 0$ and $|V(G')| > 2$,

we stop and turn to check the next pair of $(w(e_i), w(e_j))$. Otherwise go to Step 2.

Step 2. We distinguish the following three cases:

(a) $|F \setminus E'| = 0$ and $|V(G')| = 1$.

(b) $|F \setminus E'| = 0$ and $|V(G')| = 2$.

(c) $|F \setminus E'| = 1$ and $|V(G')| = 1$.

If case (a) exists, for each edge $e \in F \cup E'$, we compute the modifying costs for decreasing the weights to $w(e_i)$ and for increasing the weights to $w(e_i)$ by $d_1, d_2, \cdots, d_m$ and $i_1, i_2, \cdots, i_m$, respectively. Find two edges m_1, n_1 in $F \cup E'$ such that $m_1 \ne n_1$ and $d_{m_1} + i_{n_1}$ is minimum in $\{d_s + i_t : s, t = 1, \cdots, m, s \ne t\}$. Find a spanning tree T_1 containing $F \cup \{m_1, n_1\}$ with the edges in $F \cup E'$ and denote the total modification cost by C_1.

For each edge $e \in F \cup E'$, we compute the modifying costs for increasing the weights to $w(e_j)$ and for each edge *e* with $w(e) > w(e_j)$, we compute the modifying costs for decreasing the weights to $w(e_i)$. Similar to the above procedure we find two edges m_2, n_2 in $F \cup E' \cup \{e \in E : w(e) > w(e_j)\}$ such that $d_{m_2} + i_{n_2}$ is minimum. Find an spanning tree T_2 containing $F \cup \{m_2, n_2\}$ in $F \cup E' \cup \{m_2\}$ and denote the total modification cost by C_2.

For each edge $e \in F \cup E'$, we compute the modifying costs for decreasing the weights to $w(e_i)$ and for each edge *e* with $w(e) < w(e_i)$, we compute the modifying costs for increasing the weights to $w(e_i)$. Similar to the above procedure we find two edges m_3, n_3 in $F \cup E' \cup \{e \in E : w(e) < w(e_i)\}$ such that $d_{m_3} + i_{n_3}$ is minimum. Find an spanning tree T_3 containing $F \cup \{m_3, n_3\}$ in $F \cup E' \cup \{n_3\}$ and denote the total modification cost by C_3.

Finally we can get the solution by choosing the minimum one of C_1, C_2 and C_3.

Case (b) and case (c) can be applied similarly to case (a). By comparing all these minimum modification costs, we obtain the optimal solution (T^*, x^*). The computation in step 1 and step 2 is $O(|E|)$. From the obove discussion, we know that the total complexity of Algorithm 2 is $O(|E|^3)$. We now have shown:

Theorem 5 The problem under weighted l_1 norm can be strongly polynomially solvable by Algorithm 2.

Conclusions

In this paper, we showed that the partial inverse most unbalanced spanning tree problem under weighted Hamming distance and weighted l_1 norm can be solved efficiently by strongly polynomial algorithm. Since the great practical potential, it is meaningful to consider whether the problem with bound constraints or under other norms are polynomially solvable.

Acknowledgment

This research is supported by the National Natural Science Foundation of China under Grant No. 11171316, Zhejiang Provincial Natural Science Foundation of China (No. Y6090472) and the Scientific Research Fund of Zhejiang Provincial Education Department (No. Y201018835).

REFERENCES

[1] Ahuja R.K., Orlin J.B., Inverse optimization, part I: Linear programming and general problem, *Operations Research Letters*, 35 (2001), 771-783.

[2] Orlin J.B., Inverse optimization and partial inverse optimization, *PPT Presentation on Optimization Day Columbia University*, (2003), November 3.

[3] Heuberger C., Inverse combinatorial optimization: a survey on problems, methods, and results, *Journal of Combinatorial Optimization*, 8 (2004), 329-361.

[4] Gentry S., Partial inverse linear programming, *MIT Lab for Information and Decision Systems Report*, LIDS-P-2532 (2001), December.

[5] Yang X.G., Complexity of partial inverse assignment problem and partial inverse cut problem, *RAIRO Oper. Res.* 35 (2001), 117-126.

[6] Lai T.C., Orlin J.B., The Complexity of Preprocessing, *Research Report of Sloan School of Management*, MIT, (2003).

[7] Yang X.G., Zhang J.Z., Inverse sorting problem by minimizing the total weighted number of changes and partial inverse sorting problems, *Computational Optimization and Applications*, 36 (2007), 55-66.

[8] Yang X.G., Zhang J.Z., Partial inverse assignment problems under l_1 norm, *Operations Research Letters*, 35 (2007), 23-28.

[9] Cai M.C., Duin C.W., Yang X.G., Zhang J.Z., The partial inverse minimum spanning tree problem when weight increase is forbidden, *European Journal of Operational Research*, 188 (2008), 348-353.

[10] Camerini P.M., Maffioli F., Martello S., Toth P., Most and least uniform spanning trees, *Discrete Applied Mathematics*, 15 (1986), 181-197.

[11] Galil Z., Schieber B., On finding most uniform spanning trees, *Discrete Applied Mathematics*, 20 (1988), 173-175.

[12] Punnen A.P., Nair K.P.K., Constrained balanced optimization problems, *Computers and Mathematics with Applications*, 37 (1999), 157-163.

[13] Wang Q., Yuan J.J., Zhang J.Z., An inverse model for the most uniform problem, *Operations Research Letters*, 36 (2008), 26-30.

***Authors**: associate prof. dr Qin Wang, Department of Mathematics, China Jiliang University, Hangzhou 310018, China, E-mail: mathqin@126.com*