J Comb Optim (2007) 13:123–135
DOI 10.1007/s10878-006-9016-6

# Some inverse min-max network problems under weighted $l_1$ and $l_\infty$ norms with bound constraints on changes

**Xiaoguang Yang · Jianzhong Zhang**

Published online: 20 October 2006


**Abstract** We consider some inverse min-max (or max-min) network problems. Such an inverse problem is to modify the weights with bound constraints so that a given feasible solution becomes an optimal solution of a min-max (or max-min) network problem, and the deviation of the weights, measured by the weighted $l_1$ norm or weighted $l_\infty$ norm, is minimum. In this paper, we present strongly polynomial time algorithms to solve the inverse min-max spanning tree problem and the inverse maximum capacity path problem.

**Keywords** Inverse min-max network problem · Weighted $l_1$ norm · Weighted $l_\infty$ norm · Bound constraints · Polynomial time algorithms

## 1 Introduction

The inverse optimization problems have attracted increasing interest in recent years. The research was motivated by its background in traffic planning, and people have found more and more applications, such as high speed communication, computerized tomography, conjoint analysis, behavioral decision making, geophysical science, performance evaluation, etc. (For example, see (Ahuja and Orlin, 2001; Heuburger, 2004; Orlin, 2003).) We believe that the research on inverse optimization problems has a great potential both in theory and in real applications.

Generally speaking, an inverse optimization problem is to find a minimal modification cost of changing parameter values of an optimization problem such that some given solutions become optimum under the new parameter values. The most commonly considered parameters are cost coefficients in the original optimization problems. Among the published literatures

X. Yang (✉)
Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100080, China
e-mail: xgyang@iss.ac.cn

J. Zhang
Department of System Engineering and Engineering Management,
The Chinese University of Hong Kong, Hong Kong, China

on inverse optimization problems, most of original problems have min-sum (or max-sum) format, i.e., the objective functions are of the format $\min \sum_{i\in I} c_i x_i$ (or $\max \sum_{i\in I} c_i x_i$), see for example (Ahmed and Guan, 2005; Ahuja and Orlin, 2000, 2001; Burkard et al., 2004; Burton and Toint, 1992, 1994; Cai and Yang, 1995; Hochbaum, 2003; Iyengar and Kang, 2005; Sakkalingam et al., 1999; Xu and Zhang, 1995; Yang and Zhang, 1999; Zhang and Liu, 1999; Zhang and Ma, 1996; Zhang et al., 1997) ; or of the format $\min \sum_{i\in I} c_i H_i(x_i)$ where $H_i(x)$ is a discrete function, see for example (He et al., 2005; Liu and He, 2006). However, when the optimization problems have min-max (or max-min) format, i.e., the objective functions are of the format $\min \max\{c_i x_i \mid i \in I\}$ (or $\max \min\{c_i x_i \mid i \in I\}$), there are only very few results on the corresponding inverse optimization problems. Among these results, Cai et al. (1999) proved that the inverse center location problem is strongly NP-hard, noting that the original problem, the center location problem, is of min-max format and polynomially solvable. Yang and Zhang (1998) discussed some inverse max-min optimization problems and showed that they are polynomially solvable, under the conditions that $l_1$ norm is used to measure the deviations of parameters and there is no restriction on the change of weights.

In this paper, we consider two inverse min-max (or max-min) network problems: inverse min-max spanning tree problem and inverse maximum capacity path problem. In fact we have also studied other inverse min-max problems such as inverse problem of min-max base of matroid and the inverse min-max arborescence problem, but due to space limitation, we would concentrate on only the two types of inverse problems. The paper has two main contributions. First we consider inverse min-max (max-min) network problems *with bound constraints*, which make the problems more practical but harder than their unbounded counterparts. Second we extend the research to the $l_\infty$ norm case. As we can see, the $l_1$ and $l_\infty$ cases have different features. Moreover, we consider *weighted norms*, i.e. weighted $l_1$ and $l_\infty$ norms which are more general.

Like inverse min-sum problems, inverse min-max problems also have a big application potential. For example, in some computer networks, the real capacity of each link in a network differs from its designed one because of wearing-down of the link after years of operation. As the maximum capacity path between a pair of nodes can be observed by sending a large file through the network, if we reasonably assume that the real capacities of links are not far from their designed ones, then we may use the inverse maximum capacity path problem to estimate the real capacities of the links.

The paper is organized as follows. In Section 2, we consider inverse min-max spanning tree problem and inverse maximum capacity path problem under weighted $l_1$ norm with bound constraints. In Section 3, we consider inverse min-max spanning tree problem and inverse maximum capacity path problem under weighted $l_\infty$ norm with bound constraints. We present some strongly polynomial combinatorial algorithms to solve the problems in these situations. In Section 4, we give some concluding remarks.

## 2 Inverse min-max spanning tree problem and inverse capacity path problem under weighted $l_1$ norm with bound constraints

In this section, we consider inverse min-max spanning tree problem and inverse maximum capacity path problem under weighted $l_1$ norm with bound constraints. In order to make our analysis clear, we first consider the inverse min-max spanning tree with a fixed objective value. Then we relax the restriction of fixed objective value to a free objective value. In the last subsection, we extend the technique developed in the first two subsections to inverse maximum capacity path problem.

## 2.1 Inverse min-max spanning tree under weighted $l_1$ norm with a fixed objective value

Let $G = (V, E)$ be a connected graph and $w$ be a weight vector defined on $E$. Let $\mathcal{T}$ denote the collection of all spanning trees of $G$. For a spanning tree $T \in \mathcal{T}$, write $w(T) = \max\{w(e) \mid e \in T\}$ and call it the weight of $T$. The min-max spanning tree problem is to find a $T^* \in \mathcal{T}$ such that $w(T^*) = \min\{w(T) \mid T \in \mathcal{T}\}$. Camerini (1978) gave an $O(|E|)$ algorithm to solve the min-max spanning tree problem in 1978.

It is straightforward to see that

**Lemma 1.** *A spanning tree $T$ of $G$ is a min-max spanning tree under a weight vector $w$ if and only if $G$ becomes disconnected after deleting the edges whose weights are not less than $w(T)$.*

Now we consider the inverse min-max spanning tree problem. For a given connected graph $G = (V, E)$ and a given spanning tree $T^* \in \mathcal{T}$, we want to modify the weights of the edges such that $T^*$ is a min-max spanning tree under the new weight vector $w^*$. Moreover, let $b^+, b^- \geq 0$ be two bound vectors defined on $E$, and let $c^+, c^- > 0$ be two cost functions defined on $E$. We require that $w - b^- \leq w^* \leq w + b^+$, and $\sum_{e\in E}[c^-(e)\max\{w(e) - w^*(e), 0\} + c^+(e)\max\{w^*(e) - w(e), 0\}]$ is minimum. Note that the objective function of the inverse problem is an (asymmetric) weighted $l_1$ norm of deviations of the modified weights from the original weights. Thus we call it *the inverse problem under weighted $l_1$ norm*.

Let $T^+$ be a min-max spanning tree under $w$, and assume $w(T^*) > w(T^+)$ for otherwise we need nothing to do. We give a range for the value $w^*(T^*)$. First, we claim that

**Lemma 2.** $w^*(T^*) \leq w(T^*)$.

**Proof:** In fact, if $w^*(T^*) > w(T^*)$, we can construct a new weight vector $\bar{w}$ such that $\bar{w}(e) = w(T^*)$ for $e \in T^*$ and $w^*(e) > w(T^*)$, and $\bar{w}(e) = w^*(e)$ otherwise. Clearly, we have $\bar{w}(T^*) = w(T^*)$. Moreover, for any spanning tree $T$, we have $w^*(T) \geq w^*(T^*)$. Let $w^*(e_T) = w^*(T)$ for some $e_T \in T$. Then if $e_T \notin T^*$, we have $\bar{w}(e_T) = w^*(e_T) = w^*(T) > w(T^*)$; if $e_T \in T^*$, we have $\bar{w}(e_T) = w(T^*)$. Therefore $\bar{w}(T) \geq \bar{w}(T^*)$, and hence $\bar{w}$ is an optimal solution too. Obviously, the cost incurred by $\bar{w}$ is less than that by $w^*$. This is a contradiction. □

On the other hand, we have that

**Lemma 3.** $w(T^+) \leq w^*(T^*)$.

**Proof:** Let $w(T^+) = \rho$ and $w^*(T^*) = \tau$, and we prove the lemma by contradiction. Suppose

$$\rho > \tau, \tag{1}$$

and we show that $w^*$ would not be an optimal solution. Let

$$\Omega_\rho = \{e \in T^* \mid w(e) \geq \rho\},$$
$$\Omega_\tau = \{e \in T^* \mid w(e) \geq \tau\}.$$

Under (1), obviously $\Omega_\rho \subseteq \Omega_\tau$. Let the cost of adjusting $w$ to $w^*$ be $C^*$, which includes at least the cost of reducing the weights $w(e)$ for $e \in \Omega_\tau$ to $\tau$. Thus

$$C^* \geq \sum_{e\in\Omega_\tau}\{c^-(e)(w(e)-\tau)\} > \sum_{e\in\Omega_\rho}\{c^-(e)(w(e)-\rho)\}.$$

Now we define $\tilde{w}$ as follows:

$$\tilde{w}(e) = \begin{cases} \rho, & \text{if } e \in \Omega_\rho; \\ w(e), & \text{otherwise.} \end{cases}$$

For each spanning tree $T$, as $w(T) \geq \rho$, we see that $\tilde{w}(T) \geq \rho$. On the other hand, the definition of $\tilde{w}$ shows that $\tilde{w}(T^*) = \rho$, i.e., under $\tilde{w}$, $T^*$ becomes the minimum weight tree. The cost of adjusting $w$ to $\tilde{w}$, denoted by $\tilde{C}$, is

$$\tilde{C} = \sum_{e\in\Omega_\rho}\{c^-(e)(w(e)-\rho)\} < C^*,$$

which shows that $w^*$ is not an optimal (minimum cost) solution, a contradiction. □

Moreover, since there are lower bounds on the reduction of weights, the smallest possible value of $w^*(T^*)$ is $\max\{w(e) - b^-(e) \mid e \in T^*\}$. Write $\underline{w} = \max\{w(T^+), \max\{w(e) - b^-(e) \mid e \in T^*\}\}$, and from the above analysis we know that $\underline{w} \leq w^*(T^*) \leq w(T^*)$.

Before we consider how to solve the inverse min-max spanning tree problem directly, let us consider a *restricted version* of the inverse min-max spanning tree problem. That is, for a given value $p$, we first consider how to make $T^*$ a min-max spanning tree under a weight vector $w_p$ such that $w_p(T^*) = p$, and $w_p$ satisfies the bound restrictions and makes the modification cost minimum. We may call this restricted version of the inverse min-max spanning tree problem *the inverse min-max spanning tree problem with value $p$*. Due to the range obtained for the value of $w^*(T^*)$, we need only consider $p$ in the interval $[\underline{w}, w(T^*)]$.

Let $T^*(p) = \{e \in T^* \mid w(e) \geq p\}$. Clearly, for each edge $e \in T^*(p)$, we need reduce the weight of $e$ to $p$ in order to let the maximum weight on $T^*$ be equal to $p$. Define $C(T^*(p)) = \sum\{c^-(e)(w(e) - p) \mid e \in T^*(p)\}$, i.e. $C(T^*(p))$ is the cost to let the largest weight of $T^*(p)$ be $p$.

Moreover, for each edge $e \in E \setminus T^*$ such that $w(e) \geq p$, we need not reduce its weight, but for $e \in E(p) := \{e \in E \mid w(e) < p\}$, we may increase its weight to $p$. Clearly, if $p' < p$, then $E(p') \subseteq E(p)$.

Consider the graph $G(p) = (V, E(p))$. If $G(p)$ is not connected, we know that $T^*$ is already a min-max spanning tree with respect to the modified weight vector $w_p$ and $w_p(T^*) = p$, where $w_p(e) = p$ for $e \in T^*(p)$ and $w_p(e) = w(e)$ for $e \in E \setminus T^*(p)$.

Thus we need only consider the case that $G(p)$ is a connected graph. Denote by $\mathcal{T}(p)$ the collection of spanning trees of $G(p)$. In this situation, we need increase weights of some edges in $E(p)$ to $p$ such that $w_p(T) = p$ for every spanning tree $T \in \mathcal{T}(p)$.

A cut $K$ of $G(p)$ is called feasible with respect to $p$ if $w(e) + b^+(e) \geq p$ for every $e \in K$. For each value of $p$, we denote by $\mathcal{K}(p)$ the collection of feasible cuts of $G(p)$.

*Definition 1.* We call $p$ feasible if every spanning tree $T \in \mathcal{T}(p)$ contains an edge $e_T$ with $w(e_T) + b^+(e_T) \geq p$.

We claim that $p$ is feasible if and only if $\mathcal{K}(p) \neq \emptyset$.

We know that a set of edges is a cut if and only if it intersects every spanning tree. Hence if $p$ is feasible, then by definition, $D(p) := \bigcup_{T\in\mathcal{T}(p)}\{e \in T \mid w(e) + b^+(e) \geq p\}$ is a feasible cut, which means that $\mathcal{K}(p) \neq \emptyset$.

Conversely if $\mathcal{K}(p) \neq \emptyset$, then there is a feasible cut $K \in \mathcal{K}(p)$ such that $w(e) + b^+(e) \geq p, \forall e \in K$. Thus for each spanning tree $T \in \mathcal{T}(p)$, there is an edge $e \in T \bigcap K$, thus $w(e) + b^+(e) \geq p$. So, $p$ is feasible by definition, and the claim holds.

Let us define a modification cost $c_p$ for each edge $e$ in $E(p)$ as follows:

$$c_p(e) = \begin{cases} c^+(e)(p - w(e)), & e \in E(p) \quad \text{and} \quad p - w(e) \leq b^+(e), \\ +\infty, & e \in E(p) \quad \text{and} \quad p - w(e) > b^+(e). \end{cases}$$

Taking $c_p(e)$ as the capacities of edges in $E(p)$, we have $C_p(K) = \sum\{c_p(e) \mid e \in K\}$ for the capacity of each cut $K$. Clearly for each $K \in \mathcal{K}(p)$, as $p - w(e) \leq b^+(e)$ for each $e \in K$, we have $C_p(K) < +\infty$. If we increase the weight of each edge in a cut $K$ to $p$ in $G(p)$, then it is guaranteed that the weight of every spanning tree $T$ in $G(p)$ is increased to $p$. Denote by $K(p)$ the minimum cut of $G(p)$ with respect to $c_p(e)$. The value $C_p(K(p))$ is in fact the minimum cost to make every spanning tree of $G(p)$ have a weight $p$.

Note that if $p$ is feasible, then $\mathcal{K}(p) \neq \emptyset$, hence we have $C_p(K(p)) < +\infty$. On the other hand if $C_p(K(p)) < +\infty$, then for each $e \in K(p)$, $w(e) + b^+(e) \geq p$, which means that $K(p)$ is a feasible cut, i.e., $\mathcal{K}(p) \neq \emptyset$. Thus $p$ is feasible. So, the existence of feasible cut can be judged by the finiteness of the value $C_p(K(p))$.

Based on the above analysis, we can see that under the weighted $l_1$ norm, it is optimal to make $T^*$ a min-max spanning tree of $G$ with $w_p(T^*) = p$ by increasing weights of the edges in $K(p)$ to $p$ and decreasing weights of the edges in $T^*(p)$ to $p$, and the optimal modification cost is

$$\phi(p) := C(T^*(p)) + C_p(K(p)). \tag{2}$$

Notice that computing $T^*(p)$ and $C(T^*(p))$ is straightforward, and the main computation is to find $K(p)$. Therefore we can conclude that

**Lemma 4.** *Solving a restricted version of the inverse min-max spanning tree problem with value $p$ under weighted $l_1$ norm can be reduced to finding a minimum cut in $G(p)$.*

### 2.2 General inverse min-max spanning tree problem under weighted $l_1$ norm with bound constraints

Now we turn to the method of solving the original version of the inverse min-max spanning tree problem.

We know that the optimal solution $w^*$ corresponds to a value $p^*$ such that $\underline{w} \leq w^*(T^*) = p^* \leq w(T^*)$, and the cost $\phi(p^*)$ is the minimum one among $\phi(p)$ for all feasible $p$. We now aim to search for this $p^*$.

Denote by $C^+(p)$ the minimum cost to make every spanning tree $T$ of $G(p)$ have $w_p(T) = p$. Based on the analysis above, we have that: if $G(p)$ is not connected, $C^+(p) = 0$; if $G(p)$ is connected, then $C^+(p) = C_p(K(p))$ which includes a special case that if there is no feasible cut, then $C^+(p) = C_p(K(p)) = +\infty$.

Clearly, if $C^+(p) = +\infty$, then $C^+(q) = +\infty$ for all $q \geq p$. Thus if we meet a value $p$ such that $C^+(p) = +\infty$, we need not search the interval $[p, w(T^*)]$.

Consider the collection of distinct values of the original weights and the upper bounds of weights within the interval $[\underline{w}, w(T^*)]$, i.e. the set $(\{w(e) \mid e \in E\} \cup \{w(e) + b^+(e) | e \in E\} \cup \{\underline{w}\}) \cap [\underline{w}, w(T^*)]$. We sort these values in a strictly increasing order, say $\underline{w} = q_1 < q_2 < \cdots < q_m = w(T^*)$. Here each $q_k$ $(k \geq 2)$ corresponds to edges with $w(e) = q_k$ or $w(e) + b^+(e) = q_k$.

We claim that

**Lemma 5.** *$C^+(p)$ is a non-decreasing function.*

**Proof:** Consider two values $p' < p$. Without loss of generality, we assume that $C^+(p') > 0$ and $C^+(p) < +\infty$.

Consider the feasible cut $K(p)$. Let $B = \{e \in K(p) \mid w(e) < p'\}$. Then $B$ is a cut of $G(p')$ with the same partition of $V$ as what $K(p)$ makes in $G(p)$. Also, it is a feasible cut of $G(p')$ because each edge $e \in B$ satisfies $w(e) + b^+(e) \geq p > p'$.

Hence we obtain $C_p(K(p)) \geq C_p(B) \geq C_{p'}(B) \geq C_{p'}(K(p'))$, namely $C^+(p) \geq C^+(p')$. This shows that $C^+(p)$ is a non-decreasing function. □

**Lemma 6.** *If there exists $p' \in (q_k, q_{k+1})$ such that $C^+(p') < +\infty$, then $C^+(p) < +\infty$ for all $p \in [q_k, q_{k+1}]$. Moreover $C^+(p)$ is a concave function in $(q_k, q_{k+1}]$, and $\lim_{p \to q_k^+} C^+(p) \geq C^+(q_k)$.*

**Proof:** Suppose there exists $p' \in (q_k, q_{k+1})$ such that $0 < C^+(p') < +\infty$. Since $C^+(p)$ is a non-decreasing function, we have $C^+(q_k) < +\infty$.

For any $p \in (q_k, q_{k+1})$, by the definition of $\{q_k\}$, it is easy to see that $G(q_{k+1}) = G(p)$. Moreover, for any edge $e$ in these two subgraphs, $w(e) + b^+(e) \geq p$ if and only if $w(e) + b^+(e) \geq q_{k+1}$. $\mathcal{K}(p) = \mathcal{K}(q_{k+1})$ for all $p \in (q_k, q_{k+1}]$. By the definition of $C_p(K(p))$, we obtain that $C^+(p) = \min\{\sum_{e \in K} c^+(e)(p - w(e)) \mid K \in \mathcal{K}(q_{k+1})\}$. Hence in the interval $(q_k, q_{k+1}]$, $C^+(p)$ is a concave piecewise linear function, and $C^+(p) < +\infty$.

Finally, as $C^+(p)$ is a non-decreasing function, which is continuous in the interval $(q_k, q_{k+1}]$, we have $\lim_{p \to q_k^+} C^+(p) \geq C^+(q_k)$. □

**Lemma 7.** *The minimizer of $\phi(p)$ in each interval $[q_k, q_{k+1}]$ is reached at one of the end-points of the interval, i.e., $q_k$ or $q_{k+1}$ is the minimizer of $\phi(p)$ in the interval $[q_k, q_{k+1}]$.*

**Proof:** Recall that $\phi(p) = C(T^*(p)) + C^+(p)$. We already know that $C^+(p)$ is a concave function on $(q_k, q_{k+1}]$. Now consider $C(T^*(p))$. It is easy to see that $T^*(p) = T^*(q_{k+1})$ for $p \in (q_k, q_{k+1}]$. Moreover,

$$
\begin{aligned}
c(T^*(q_k)) &= \sum_{e \in T^*(q_{k+1})} c^-(e)(w(e) - q_k) + \sum_{e \in T^*(q_k), w(e) = q_k} c^-(e)(w(e) - q_k) \\
&= \sum_{e \in T^*(q_{k+1})} c^-(e)(w(e) - q_k).
\end{aligned}
$$

Therefore, for $p \in [q_k, q_{k+1}]$, $C(T^*(p)) = \sum_{e \in T^*(q_{k+1})} c^-(e)(w(e) - p)$ which is a linear function in the interval $[q_k, q_{k+1}]$.

Hence we know that $\phi(p)$ is a continuous and concave function on $(q_k, q_{k+1}]$, and $\lim_{p \to q_k^+} \phi(p) \geq \phi(q_k)$.

By the concavity of function $\phi(p)$, if $\lim_{p\to q_k^+}\phi(p)\geq\phi(q_{k+1})$, then $\phi(p)\geq\phi(q_{k+1})$ for every $p\in(q_k,q_{k+1}]$, and the minimizer of $\phi(p)$ in $[q_k,q_{k+1}]$ is attained at $q_k$ or $q_{k+1}$. If $\lim_{p\to q_k^+}\phi(p)<\phi(q_{k+1})$, then $\phi(p)\geq\lim_{p\to q_k^+}\phi(p)\geq\phi(q_k)$ for every $p\in(q_k,q_{k+1}]$, and the minimizer of $\phi(p)$ in $[q_k,q_{k+1}]$ is attained at $q_k$. Hence no matter which case happens, the minimizer of $\phi(p)$ is one of $q_k$ and $q_{k+1}$. □

From Lemma 6, we need only check $\phi(p)$ values at $\{q_i\}_1^m$, then we can find the optimal solution of the inverse min-max spanning tree problem. In fact, if $\phi(q_k)=\infty$ for $1\leq k\leq m$, then the inverse min-max spanning tree problem is infeasible, otherwise the restricted version of inverse min-max spanning tree problem with value $q_k$ gives the optimal solution, where $q_k=\arg\min\{\phi(q_i)\mid i=1,2,\ldots,m\}$.

Notice that $m$, the number of restricted versions of the inverse min-max spanning tree problems to be concerned, is at most $2|E|+1$. In each restricted version of the inverse min-max spanning tree problem, we need find a minimum cut, which can be solved in strongly polynomial time (Hao and Orlin, 1994; Nagamochi and Ibaraki, 1992). Therefore we can obtain the following result.

**Theorem 1.** *The inverse min-max spanning tree problem under weighted $l_1$ norm can be solved in strongly polynomial time.*

*Remark 1.* Notice that if $C_{q_k}(K(q_k))=+\infty$, then $C_{q_i}(K(q_i))=+\infty$ for all $i\geq k$. Hence in the real implementation, we need not check each $q_k$ one by one, but use the binary-search strategy to determine the smallest $h$ such that $C_{q_h}(T^*(q_h))<+\infty$. Then we check the values $q_k$ within the interval $[q_1,q_h]$.

### 2.3 Inverse maximum capacity path problem under weighted $l_1$ norm with bound constraints

The max-min form path problem is usually called maximum capacity path problem with applications in reliability of network (see Schrijver, 2003).

In this section, we will employ the technique developed in the previous two sections to handle the inverse maximum capacity path problem under weighted $l_1$ norm with bound restriction on the change of capacities.

Let $G=(V,E)$ be a connected graph and $w$ be a capacity vector defined on $E$. Let $\mathcal{P}(s,t)$ denote the collection of all paths of $G$ between two specific nodes $s$ and $t$. For a path $P\in\mathcal{P}(s,t)$, let $w(P)=\min\{w(e)\mid e\in P\}$, which is called the capacity of the path. The maximum capacity path problem is to find a $P'\in\mathcal{P}(s,t)$ such that $w(P')=\max\{w(P)\mid P\in\mathcal{P}(s,t)\}$. The problem is also called the maximum reliability problem which can be solved in $O(|E|+|V|\log(|V|))$ time (see Schrijver, 2003, p. 117).

The inverse maximum capacity path problem under weighted $l_1$ norm is to modify the capacities of the edges such that a given $P^*\in\mathcal{P}(s,t)$ becomes a maximum capacity path under a new capacity vector $w^*$ which meets the requirements that $\sum_{e\in E}[c^-(e)\max\{w(e)-w^*(e),0\}+c^+(e)\max\{w^*(e)-w(e),0\}]$ is minimum, and $w(e)-b^-(e)\leq w^*(e)\leq w(e)+b^+(e)$ for all $e\in E$.

Let $P^+\in\mathcal{P}(s,t)$ be the max-min $s$-$t$ path under $w$. $\overline{w}:=\min\{w(P^+),\min\{w(e)+b^+(e)\mid e\in P^*\}\}$. It is easy to see that $w(P^*)\leq w^*(P^*)\leq\overline{w}$.

For any value $p \in [w(P^*), \overline{w}]$, let $\hat{E}(p) = \{e \in E \mid w(e) > p\}$ and define a modification cost $\overline{c}_p$ for every edge in $\hat{E}(p)$ as follows:

$$\overline{c}_p(e) = \begin{cases} c^-(e)(w(e) - p), & e \in \hat{E}(p) \quad \text{and} \quad w(e) - p \le b^-(e); \\ +\infty, & e \in \hat{E}(p) \quad \text{and} \quad w(e) - p > b^-(e). \end{cases}$$

Taking $\overline{c}_p(e)$ as the capacity of edge $e$ and let $\hat{K}(p)$ be a minimum capacity $s$-$t$ cut on $\hat{E}$. It is easy to show that the cheapest way to make $P^*$ a maximum capacity path with $w_p(P^*) = p$ is to increase capacities to $p$ for those edges on $P^*$ whose original capacities are less than $p$, and to decrease capacities of the edges in $\hat{K}(p)$ to $p$.

Similar to Section 2.2, we need only search the points in $(\{w(e) \mid e \in E\} \cup \{w(e) - b^-(e) \mid e \in E\} \cup \{\overline{w}\}) \cap [w(P^*), \overline{w}]$. The number of such target points is at most $2|E| + 1$, and at each point the main computation is to find a minimum $s$-$t$ cut. Therefore we conclude that

**Theorem 2.** *The inverse maximum capacity path problem under weighted $l_1$ norm can be solved in strongly polynomial time.*

## 3 Inverse min-max spanning tree problem and inverse maximum capacity path problem under weighted $l_\infty$ norm with bound constraints

In this section, we consider inverse min-max spanning tree problem and inverse maximum capacity path problem under weighted $l_\infty$ norm with bound constraints. Similar to the last section, we consider how to solve inverse min-max spanning tree problem first, and then extend the technique to solve inverse maximum capacity path problem.

### 3.1 Inverse min-max spanning tree problem under weighted $l_\infty$ norm with bound constraints

Now we consider the inverse min-max spanning tree problem under the weighted $l_\infty$ norm. That is, we want to minimize

$$\max_{e \in E} \max\{c^-(e) \max\{w(e) - w^*(e), 0\}, c^+(e) \max\{w^*(e) - w(e), 0\}\}.$$

Again, we first consider, for a given value $p$, the restricted version of the inverse min-max spanning tree problem which is how to make $T^*$ a min-max spanning tree under the new weight vector $w_p$ such that $w_p(T^*) = p$, $w_p$ satisfies the bound restrictions and the modification cost under the weighted $l_\infty$ norm is minimum.

Using the similar arguments as in the last section, we only need to consider $p \in [\underline{w}, w(T^*)]$; and for each $p$, if $G(p)$ is not connected, then only the cost

$$\hat{c}(T^*(p)) = \max\{c^-(e)(w(e) - p) \mid e \in T^*(p)\} \tag{3}$$

is involved and the solution can be obtained with great ease. So, we now assume that $G(p)$ is connected. In this case we should find a cut $K$ such that $w(e) + b^+(e) \ge p$ for all $e \in K$, and

$$\hat{c}_p(K) := \max\{c^+(e)(p - w(e)) \mid e \in K\} \tag{4}$$

is minimum among all feasible cuts.

Let $c_p(e)$ be defined the same as in the last section. So, a cut is feasible if and only if every edge $e$ of the cut has a finite value $c_p(e)$. Let $T'$ be the max-sum spanning tree of $G(p)$ with respect to $c_p(e)$. Let $e_{T'} \in T'$ be an edge with the smallest value $c_p(e)$ in the tree. Clearly if $c_p(e_{T'}) = \infty$, then $\hat{c}_p(K) = \infty$ for every cut $K$, as $K$ must contain an edge of $T'$. In this case it is impossible to make $T^*$ a min-max spanning tree with $w_p(T^*) = p$.

So we assume that $c_p(e_{T'}) < \infty$. Let $\hat{K}(e_{T'})$ be the fundamental cut of $e_{T'}$ with respect to $T'$ in $G(p)$. Then $c_p(e) \le c_p(e_{T'})$ for every $e \in \hat{K}(e_{T'})$ as $T'$ is the max-sum spanning tree. Thus $\hat{K}(e_{T'})$ is feasible and $\hat{c}_p(\hat{K}(e_{T'})) = c_p(e_{T'})$.

We claim that $\hat{K}(e_{T'})$ is the minimum cost cut with respect to the value $\hat{c}_p(K)$ among all feasible cuts $K$. Indeed, as each $K$ contains at least one edge $e' \in T'$, $\hat{c}_p(K) \ge c_p(e') \ge c_p(e_{T'}) = \hat{c}_p(\hat{K}(e_{T'}))$.

Therefore we can see that under the weighted $l_\infty$ norm, the minimum cost adjustment to make $T^*$ a min-max spanning tree of $G(p)$ with $w_p(T^*) = p$ is to increase weights of the edges in $\hat{K}(e_{T'})$ to $p$ and to decrease weights of the edges in $T^*(p)$ to $p$, and the optimal modification cost is

$$\psi(p) := \max\{\hat{c}(T^*(p)), \hat{c}_p(\hat{K}(e_{T'}))\} = \max\{\hat{c}(T^*(p)), c_p(e_{T'})\},$$

where $\hat{c}(T^*(p))$ is given by (3).

Since the main computation is to find the max-sum spanning tree $T'$, we obtain that

**Lemma 8.** *Solving the restricted version of the inverse min-max spanning tree problem with value $p$ under the weighted $l_\infty$ norm can be reduced to finding a max-sum spanning tree under the edge weight $c_p(e)$ on $G(p)$.*

Let us turn to the original version of the inverse min-max spanning tree problem under the weighted $l_\infty$ norm. Once again, we consider the sequence of values $\underline{w} = q_1 < q_2 < \cdots < q_m = w(T^*)$, and denote by $\hat{C}(p)$ the minimum cost to make every spanning tree $T$ of $G(p)$ satisfy $w_p(T) \ge p$. If $G(p)$ is not connected, then $\hat{C}(p) = 0$; otherwise $\hat{C}(p) = \hat{c}_p(\hat{K}(p))$, where $\hat{K}(p)$ is the minimum cost cut of $G(p)$. Using the same arguments as in Section 2, we have that

(a) $\hat{C}(p)$ is a non-decreasing function;
(b) if $\hat{C}(p') < +\infty$ for a $p' \in (q_k, q_{k+1})$, then $\hat{C}(p) < +\infty$ for all $p \in [q_k, q_{k+1}]$;
(c) For all $p \in (q_k, q_{k+1}]$, $\mathcal{K}(p) = \mathcal{K}(q_{k+1})$, and

$$\hat{C}(p) = \min_{K \in \mathcal{K}(q_{k+1})} \max_{e \in K}\{c^+(e)(p - w(e))\}; \tag{5}$$

(d) $\lim_{p \to q_k^+} \hat{C}(p) \ge \hat{C}(q_k)$.

From (5) above we know that over the interval $(q_k, q_{k+1}]$, $\hat{C}(p)$ is a piecewise linear function; by (3), we see that $\hat{c}(T^*(p))$ is also a piecewise linear function. Therefore $\psi(p)$ is a piecewise linear function in $(q_k, q_{k+1}]$. More exactly, over this subinterval,

$$\psi(p) = \max\left\{\max_{e\in T^*(q_{k+1})}\{c^-(e)(w(e)-p)\},\ \min_{K\in\mathcal{K}(q_{k+1})}\max_{e\in K}\{c^+(e)(p-w(e))\}\right\}.$$

Note that not like $\phi(p)$, function $\psi(p)$ is in general not a concave one in each interval, and its minimizer on each $[q_k, q_{k+1}]$ is not necessary an endpoint of the interval. But as $\psi$ is piecewise linear, its minimizer over $(q_k, q_{k+1}]$ must be located at either some points where two linear functions of the following group intersect:

$$\{c^-(e)(w(e)-p) \mid e \in T^*(q_{k+1})\} \cup \{c^+(e)(p-w(e)) \mid e \in \mathcal{K}(q_{k+1})\},$$

or an endpoint of the interval.

Therefore, we need check only sequence $\{q_i\}_1^m$ and the intersection points of linear functions $\{c^-(e)(w(e)-p) \mid e \in T^*\} \cup \{c^+(e)(p-w(e)) \mid e \in E\}$ within interval $[q_1, q_m]$. As the number of linear functions under consideration is $|E|+|V|-1$, the number of such intersection points is at most $\frac{1}{2}(|E|+|V|-1)(|E|+|V|-2)$. The total number of values $p$ to be checked in order to find $\min\psi(p)$ is at most $2|E|+1+\frac{1}{2}(|E|+|V|-1)(|E|+|V|-2)$, and for each such $p$, the main computation is to find a max-sum spanning tree which can be completed in $O(|E|+|V|\log(|V|))$ time (see Schrijver, 2003). Hence,

**Theorem 3.** *The inverse min-max spanning tree problem under weighted $l_\infty$ norm can be solved in strongly polynomial time.*

*Remark 2.* Let us consider the case that all modification costs are equal but the modification of weights has bound restrictions. It is straightforward to see that over the subinterval $(q_k, q_{k+1}]$, the cost function $\psi(p)$ becomes

$$\psi(p) = \max\left\{w(T^*)-p,\ \min_{K\in\mathcal{K}(q_{k+1})}\max_{e\in K}\{(p-w(e))\}\right\}.$$

Hence the minimizer of $\psi(p)$ over $[q_k, q_{k+1}]$ must be located at either some intersect points of the linear function $w(T^*)-p$ with a linear function in the group $\{p-w(e) \mid e \in \mathcal{K}(q_{k+1})\}$, or an endpoint of the interval. That is, the minimizer of $\psi(p)$ over $[q_k, q_{k+1}]$ must be contained in the set

$$\{q_k, q_{k+1}\} \cup \left\{\frac{1}{2}(w(T^*)+w(e)) \mid e \in \mathcal{K}(q_{k+1})\right\}.$$

So, the overall minimizer of $\psi(p)$ must be contained in the following set

$$\Gamma = \{q_i\}_1^m \cup \left\{\frac{1}{2}(w(T^*)+w(e)) \mid e \in E\right\}.$$

Therefore, in order to find the minimizer of $\psi(p)$, we need check only the points in the set $\Gamma$, which contains at most $3|E|+1$ points. So we can solve the case more efficiently than the general one.

*Remark 3.* If there are no bounds on the restriction of modifications and all modification costs are equal, say $c^+(e) = c^-(e) = 1$ for all $e \in E$, then the inverse min-max spanning tree problem under $l_\infty$ norm becomes quite easy. First, it is easy to see that the minimum cost $\psi(p^*) \geq \frac{1}{2}(w(T^*) - w(T^+))$, where $T^+$ is any min-max spanning tree with respect to the original weights, and $p^*$ represents the value of $w^*(T^*)$ under the optimal solution $w^*$. In fact, we have $\psi(p^*) \geq \max\{w(T^*) - w^*(T^*), w^*(T^+) - w(T^+)\}$, and $w^*(T^*) \leq w^*(T^+)$. Hence $\psi(p^*) \geq \frac{1}{2}(w(T^*) - w(T^+))$.

Second, the lower bound $\tau := \frac{1}{2}(w(T^*) - w(T^+))$ can be reached, and hence the minimum cost is $\psi(p^*) = \tau$. In fact, let us set $p^* = \frac{1}{2}(w(T^*) + w(T^+))$ and define a new weight vector

$$w^*(e) = \begin{cases} \min\{w(e), p^*\}, & \forall e \in T^*, \\ p^*, & \forall e \in \hat{K}(e_\rho), \\ w(e), & \text{otherwise}, \end{cases}$$

where $\hat{K}(e_\rho)$ is the fundamental cut of $e_\rho$ with respect to $T_\rho$ in $G(p^*)$, $T_\rho$ is the min-sum spanning tree of $G(p^*)$ with respect to $w$, and $e_\rho$ is the largest edge of $T_\rho$ with respect to $w$.

Note that $w(e_\rho) \leq w(e)$ for all $e \in \hat{K}(e_\rho)$ as $T_\rho$ is the min-sum spanning tree of $G(p^*)$ under $w$. Hence $T_\rho$ is also a min-max spanning tree of $G$ under $w$ and $w(e_\rho) = w(T_\rho) = w(T^+)$. It is easy to verify that $w^*(T^*) = p^*$ and $T^*$ is a min-max spanning tree under $w^*$.

Now let us show that the $l_\infty$ cost for changing $w$ to $w^*$ is indeed $\tau$. The cost for changing weights of some edges in $T^*$ is $\max\{w(e) - p^* \mid e \in T^*, w(e) \geq p^*\} = w(T^*) - p^* = \tau$, and the cost for changing weights of the edges in $\hat{K}(e_\rho)$ is $\max\{p^* - w(e) \mid e \in \hat{K}(e_\rho)\} = p^* - \min\{w(e) \mid e \in \hat{K}(e_\rho)\} = p^* - w(e_\rho) = \tau$. Hence the minimum cost $\psi(p^*) = \tau$.

### 3.2 Inverse maximum capacity path problem under weighted $l_\infty$ norm with bound constraints

We now turn to the inverse maximum capacity path problem under the weighted $l_\infty$ norm.

Let $\overline{c}_p(e)$ be defined the same as in Section 2.3 and let $\hat{T}(p)$ be the max-sum spanning tree of $\hat{G}(p) = (V, \hat{E}(p))$ with respect to $\overline{c}_p(e)$. There is a unique path between $s$ and $t$ on $\hat{T}(p)$, say $\hat{P}(p)$. Let $\hat{e}$ be an edge in $\hat{P}(p)$ with the smallest value $\overline{c}_p(e)$. Clearly if $\overline{c}_p(\hat{e}) = \infty$, then $\overline{c}_p(e) = \infty$ for all $e \in \hat{P}(p)$. In this case for each $s$-$t$ cut $K$, as $\hat{P}(p) \cap K \neq \emptyset$,

$$\overline{c}_p(K) = \max\{\overline{c}_p(e) \mid e \in K\} = \infty,$$

i.e., it is impossible to make $P^*$ a maximum capacity path with $w_p(P^*) = p$.

So we assume that $\overline{c}_p(\hat{e}) < \infty$. Let $K(\hat{e})$ be the fundamental cut of $\hat{e}$ with respect to $\hat{T}(p)$ in $\hat{G}(p)$. Clearly $K(\hat{e})$ is an $s$-$t$ cut, and $\overline{c}_p(e) \leq \overline{c}_p(\hat{e})$ for all $e \in K(\hat{e})$ as $\hat{T}(p)$ is the max-sum spanning tree. Hence $K(\hat{e})$ is a feasible $s$-$t$ cut and $\overline{c}_p(K(\hat{e})) = \overline{c}_p(\hat{e})$.

We claim that $K(\hat{e})$ is the minimum cost $s$-$t$ cut with respect to the weighted $l_\infty$ cost $\overline{c}_p(K)$ among all $s$-$t$ cuts $K$. Indeed, as every $K$ contains an edge $e'$ of $\hat{P}(p)$, $\overline{c}_p(K) \geq \overline{c}_p(e') \geq \overline{c}_p(\hat{e}) = \overline{c}_p(K(\hat{e}))$.

Therefore, in order to make $P^*$ a maximum capacity path and $w_p(P^*) = p$ under weighted $l_\infty$ norm, it is optimal to increase capacities to $p$ for those edges on $P^*$ whose original capacities are less than $p$, and to decrease capacities of the edges in $K(\hat{e})$ to $p$.

Similar to Section 3.1, we need only check the intersection points of every two lines among $\{c^+(e)(p - w(e)) \mid e \in P^*\} \cup \{c^-(e)(w(e) - p) \mid e \in E\}$ and the points in the set

$\{w(e) \mid e \in E\} \cup \{w(e) - b^-(e) \mid e \in E\} \cup \{\overline{w}\}$ which lie in the interval $[w(P^*), \overline{w}]$. At each of these points, the main computation is to find a max-sum spanning tree. Thus we have the following result.

**Theorem 4.** *The inverse maximum capacity path problem under weighted $l_\infty$ norm can be solved in strongly polynomial time.*

*Remark 4.* Similar to the inverse min-max spanning tree problem, if there are no restrictions on the modification of edge capacities, and the modification costs on all edges are equal, say $c^+(e) = c^-(e) = 1$ for all $e \in E$, then the inverse maximum path problem under $l_\infty$ norm is easy to solve. The optimal solution can be written directly as follows.

$$w^*(e) = \begin{cases} \max\{w(e), w(P^*) + \delta\}, & \forall e \in P^*, \\ \min\{w(e), w(P^*) - \delta\}, & \forall e \in K(e_{T^+}), \\ w(e), & \text{otherwise}, \end{cases}$$

where $K(e_{T^+})$ is the fundamental cut of $e_{T^+}$ with respect to a max-sum spanning tree $T^+$ in $G$ under the original capacity vector $w$, and $e_{T^+}$ is the minimum capacity edge on the unique $s$-$t$ path in $T^+$. In this case the minimum cost is

$$\delta = \frac{w(e_{T^+}) - w(P^*)}{2}.$$

## 4 Concluding remarks

In this paper, we present some technique to solve the inverse min-max spanning tree problems under weighted $l_1$ norm and weighted $l_\infty$ norm with bound constraints on changes of weights. The technique can also be extended to handle the inverse problem of min-max base of matroid and the inverse min-max arborescence problem, under both weighted $l_1$ norm and weighted $l_\infty$ norm.

**Acknowledgments** The research work is supported by the Hong Kong University Grand Council under the grant CERG CUHK103105 (formerly CityU 103105). X.G. Yang's work is also supported by the National Key Research and Development Program of China (2002CB312004) and the National Natural Science Foundation of China (No. 70425004).

## References

Ahmed S, Guan YP (2005) The inverse optimal value problem. Math Program 102:91–110

Ahuja RK, Orlin JB (2001) Inverse optimization, part i: linear programming and general problem. Oper Res 35:771–783

Ahuja RK, Orlin JB (2000) A faster algorithm for the inverse spanning tree problem. J Algorithms 34:177–193

Burkard R, Pleschiutschnig C, Zhang JZ (2004) Inverse median problems. Discrete Optim 1:23–39

Burton D, Toint PhL (1992) On an instance of the inverse shortest paths problem. Math Program 53:45–61

Burton D, Toint PhL (1994) On the use of an inverse shortest paths algorithm for recovering linearly correlated costs. Math Program 63:1–22

Cai MC, Yang XG (1995) Inverse shortest path problems. Operations research and its applications. In: First international symposium ISORA'95 Beijing PR China

Cai MC, Yang XG, Zhang J (1999) The complexity analysis of the inverse center location problem. J Global Optim 15:213–218
Camerini PM (1978) The min-max spanning tree problem and some extensions. Inf Proc Lett 7:10–14
Hao J, Orlin JB (1994) A faster algorithm for finding the minimum cut in a directed graph. J Algorithms 17:424–446
Heuburger C (2004) Inverse optimization, a survey on problems, methods, and results. J Comb Optim 8:329–361
Hochbaum DS (2003) Efficient algorithms for the inverse spanning tree problem. Oper Res 51:785–797
He Y, Zhang B, Yao E (2005) Weighted inverse minimum spanning tree problems under Hamming distance. J Comb Optim 9:91–100
Iyengar G, Kang WM (2005) Inverse conic programming with applications. Oper Res Lett 33:319–330
Liu LC, He Y (2006) Inverse minimum spanning tree problem and reverse shortest-path problem with discrete values. Prog Nat Sci 16:649–655
Nagamochi H, Ibaraki T (1992) Computing edge-connectivity in multigraphs and capacitated graphs. SIAM J Discrete Math 5:54–66
Orlin JB (2003) Inverse optimization and partial inverse optimization. PPT presentation on Optimization Day Columbia University November 3
Schrijver A (2003) Combinatorial optimization: polyhedra and efficiency. Springer-Verlag Berlin and Heidelberg
Sokkalingam PT, Ahuja RK, Orlin JB (1999) Solving inverse spanning tree problems through network flow techniques. Oper Res 47:291–298
Xu S, Zhang J (1995) An inverse problem of the weighted shortest path problem. Japan J Ind Appl Math 12:47–59
Yang C, Zhang J (1998) Inverse maximum capacity problems. OR Spektrum 20:97–100
Yang C, Zhang J (1999) Two general methods for inverse optimization problems. Appl Math Lett 12:69–72
Zhang J, Liu Z (1999) A further study on inverse linear programming problems. J Comput Appl Math 106:345–359
Zhang J, Ma Z (1996) A network flow method for solving some inverse combinatorial optimization problems. Optimization 37:59–72
Zhang J, Xu S, Ma Z (1997) An algorithm for inverse minimum spanning tree problem. Optim Meth Soft 8:69–84