

Inverse Minimum Cut Problem with Lower and Upper Bounds

Adrian Deaconu * and Laura Ciupala

Department of Mathematics and Computer Science, Faculty of Mathematics and Computer Science,
Transilvania University of Brasov, 50003 Braşov, Romania; laura.ciupala@unitbv.ro

* Correspondence: a.deaconu@unitbv.ro; Tel.: +40-776378881

Received: 23 July 2020; Accepted: 31 August 2020; Published: 3 September 2020



Abstract: The inverse minimum cut problem is one of the classical inverse optimization researches. In this paper, the inverse minimum cut with a lower and upper bounds problem is considered. The problem is to change both, the lower and upper bounds on arcs so that a given feasible cut becomes a minimum cut in the modified network and the distance between the initial vector of bounds and the modified one is minimized. A strongly polynomial algorithm to solve the problem under l_1 norm is developed.

Keywords: minimum cut; maximum flow; inverse optimization

1. Introduction

Inverse optimization is a relatively new research domain (around 20 years old) and it has been intensively studied. Many papers have recently been published in this domain and are still being published nowadays. Inverse problems have lately found a lot of applications in modern areas, such as mathematical biology, materials science, remote sensing, medical imaging, seismology, geophysics, oceanography, mathematical finance, etc. An inverse combinatorial optimization problem consists of modifying some parameters of a network, such as capacities or costs, so that a given feasible solution of the direct optimization problem becomes an optimum solution and the distance between the initial vector and the modified vector of parameters is minimized. Different norms, such as l_1 , l_∞ , and even l_2 , or Hamming distances are considered to measure this distance. In the last years, many papers were published in the field of inverse combinatorial optimization [1–4]. Among these problems, inverse maximum flow (IMF) and inverse minimum cut (IMC) problems were studied since their direct counterparts are well known related problems. Yang et al. [5] presented the first strong polynomial-time algorithms to solve these two problems under the l_1 norm. IMF is to change as little as possible the capacities of arcs so that a given feasible flow becomes optimum (maximum). IMF under l_∞ was studied in [6]. Other IMF problems were studied in [7–9]. IMF considered for modification of upper and lower bounds on arcs was studied in [10] and it is the first time when both bounds are taken into account. The inverse minimum flow problem (ImF) was studied in [11]. In the case of IMC, the capacities (upper bounds) of arcs are changed, so that a cut becomes a minimum cut in the modified network. IMC is less studied than its related problem, IMF. Some inverse minimum cuts problems are considered in [12].

This paper studies the inverse minimum cut problem, where both lower and upper bounds on arcs are modified, so that the distance between initial vector of bounds and the vector of changed bounds measured with the l_1 norm is minimized and a given cut in the initial network becomes a minimum cut in the modified one. This problem is denoted as IMCUL. Although the direct problem of minimum cut with lower and upper bounds is a generalization of the problem of minimum cut with only upper bounds (where lower bounds can be considered equal to 0), the corresponding inverse

problem IMCUL is a different problem and is not a generalization of IMC, since in the case of IMC lower bounds cannot be modified. Moreover, as we shall see in this paper, in comparison with [5], the methods used to solve these two inverse problems are completely different.

The rest of the paper is organized as follows. Section 2 presents the minimum cut and maximum flow with lower and upper bounds problems. Section 3 introduces the inverse minimum cut problem with lower and upper bounds (IMCUL) and a strongly polynomial algorithm to solve the IMCUL under the l_1 norm is also presented. In Section 4, an example is given to illustrate the idea of the proposed algorithm to solve IMCUL under the l_1 norm. Some conclusions are made and some open problems are presented in the last section.

2. Minimum Cut and Maximum Flow

Let $G = (V, A, c, l, s, t)$ be a network, where V is the set of nodes, A is the set of directed arcs (each arc from A connects two nodes i and j from V), c is the upper bound (capacity) application $c : A \rightarrow R_+$, $l : A \rightarrow R_+$ is the lower bound application, s and t are two special nodes from V . These two nodes are called source, respectively, sink. For an arc $a = (i, j) \in A$, $l(a)$ and $c(a)$ are the minimum and, respectively, the maximum amount of flow that can pass through arc a from node i to node j . Of course, $l(a) \leq c(a)$.

A feasible flow in the network G is a function $f : A \rightarrow R_+$ that satisfies simultaneously the conditions (1–4):

$$l(i, j) \leq f(i, j) \leq c(i, j), \forall (i, j) \in A \quad (1)$$

$$\sum_{j \in V, (i, j) \in A} f(i, j) - \sum_{j \in V, (j, i) \in A} f(j, i) = v(i), \forall i \in V \quad (2)$$

$$v(i) = 0, \forall i \in V - \{s, t\} \quad (3)$$

$$v(s) = -v(t) > 0 \quad (4)$$

The value of flow f is denoted by $v(f)$ and defined as follows:

$$v(f) = v(s) = -v(t).$$

A maximum flow is a feasible flow f in G and among all feasible flows in G it has the maximum value (see (5)).

$$v(f) = \max\{v(f') \mid f' \text{ feasible flow in } G\} \quad (5)$$

Before presenting the definitions of the cut and of the capacity of a cut we introduce the following notations:

- for two non-empty sets of nodes V_1 and V_2 from V , (V_1, V_2) denotes the set of arcs that connects nodes from V_1 with the nodes from V_2 , i.e., $(V_1, V_2) = \{a = (x, y) \in A \mid x \in V_1, y \in V_2\}$;
- for a function $d : A \rightarrow R_+$, we define $d(V_1, V_2) = \sum_{(i, j) \in (V_1, V_2)} d(i, j)$.

For a non-empty set $X \subset V$, we denote $\bar{X} = V - X$. The set of arcs $[X, \bar{X}] = (X, \bar{X}) \cup (\bar{X}, X)$ is called *cut* in network G . If $s \in X$ and $t \in \bar{X}$ then $[X, \bar{X}]$ is called $s - t$ cut;

The *capacity of the cut* $[X, \bar{X}]$ is denoted by $c[X, \bar{X}]$ and is defined in (6):

$$c[X, \bar{X}] = c(X, \bar{X}) - l(\bar{X}, X) \quad (6)$$

From now on through the paper we will refer to a $s - t$ cut as a cut.

We recall two theorems that connect (minimum) cuts with (maximum) flows (see [13]):

Theorem 1. The value of a feasible flow f in G equals the value of the flow on a $s - t$ cut $[X, \bar{X}]$ and does not exceed the value of this cut (see (7)).

$$v(f) = f[X, \bar{X}] \leq c[X, \bar{X}]. \quad (7)$$

Theorem 2. The value of a minimum $s - t$ cut $[X, \bar{X}]$ in G equals the value of a maximum flow f in G (see (8)).

$$v(f) = f[X, \bar{X}] = c[X, \bar{X}]. \quad (8)$$

3. The Inverse Minimum Cut Problem with Lower and Upper Bounds

Let $G = (V, A, c, l, s, t)$ be a network.

We denote by $w = (c, l)$ the concatenation of the vectors c and l (the components of l are placed after the components of c). We call w the bound vector of the network G .

Let $[X, \bar{X}]$ be a $s - t$ cut in network G .

The set of w for which $[X, \bar{X}]$ is a $s - t$ cut in the corresponding network is denoted by $Q[X, \bar{X}]$ and defined in (9):

$$v(f) = f[X, \bar{X}] = c[X, \bar{X}]. \quad (9)$$

The inverse minimum cut problem (IMCUL) is to change the bound vector w so that the given $s - t$ cut $[X, \bar{X}]$ becomes a minimum cut in G and the distance between the initial bound vector w and the modified vector of bounds denoted $\bar{w} \in Q[X, \bar{X}]$ is minimized (see (10)):

$$\min \|w - \bar{w}\| \quad (10)$$

We shall concentrate now on the inverse minimum cut problem with lower and upper bounds under the l_1 norm (denoted IMCUL1). More exactly, for IMCUL the distance between w and \bar{w} is measured using the l_1 norm. This means that the sum of absolute modifications of the bounds on arcs is minimized (see (11a)):

$$\min \|w - \bar{w}\|_1 = \min \{ |c - \bar{c}| + |l - \bar{l}| \} \quad (11a)$$

$$\bar{w} \in Q[X, \bar{X}] \quad (11b)$$

We recall the definition of l_1 norm for the n -dimensional vector x :

$$l_1(x) = \|x\|_1 = |x| = \sum_{i=1}^n |x_i|$$

Lemma 1. If $w^* = (c^*, l^*)$ is the optimum solution of IMCUL1 then we have:

$$c^*(i, j) \leq c(i, j), \forall (i, j) \in (X, \bar{X}) \quad (12a)$$

$$c^*(i, j) \geq c(i, j), \forall (i, j) \in A - (X, \bar{X}) \quad (12b)$$

$$l^*(i, j) \geq l(i, j), \forall (i, j) \in (\bar{X}, X) \quad (12c)$$

$$l^*(i, j) \leq l(i, j), \forall (i, j) \in A - (\bar{X}, X) \quad (12d)$$

Proof. Let us suppose that one of the relations (12a) or (12b) is false. This means that there exists an arc $(i_0, j_0) \in (X, \bar{X})$ so that $c^*(i_0, j_0) > c(i_0, j_0)$ or $(i_0, j_0) \in A - (X, \bar{X})$ so that $c^*(i_0, j_0) < c(i_0, j_0)$.

We define a capacity vector c^{**} as follows:

$$c^{**}(i, j) = \begin{cases} c(i, j), & \text{if } (i, j) = (i_0, j_0) \\ c^*(i, j), & \text{if } (i, j) \neq (i_0, j_0). \end{cases} \quad (13)$$

It is easy to observe that $(c^{**}, l^*) \in Q[X, \bar{X}]$ (see (13)) and

$$|l^* - l| + |c^{**} - c| < |l^* - l| + |c^* - c|. \quad (14)$$

Relation (14) contradicts the optimality of (c^*, l^*) .

Let us suppose now that one of the relations (12c) or (12d) is false. This means that there exists an arc $(i_0, j_0) \in (\bar{X}, X)$ so that $l^*(i_0, j_0) < l(i_0, j_0)$ or $(i_0, j_0) \in A - (\bar{X}, X)$ so that $l^*(i_0, j_0) > l(i_0, j_0)$.

We define the following lower bound vector l^{**} :

$$l^{**}(i, j) = \begin{cases} l(i, j), & \text{if } (i, j) = (i_0, j_0) \\ l^*(i, j), & \text{if } (i, j) \neq (i_0, j_0). \end{cases} \quad (15)$$

It is easy to observe that $(c^*, l^{**}) \in Q[X, \bar{X}]$ (see (15)) and

$$|l^{**} - l| + |c^* - c| < |l^* - l| + |c^* - c|. \quad (16)$$

Relation (14) contradicts the optimality of (c^*, l^*) . \square

Lemma 2.

- (a) If the capacity of an arc $(i_0, j_0) \in A$ is increased with the value $u \geq 0$ then the difference between the value of the maximum flow in the modified network and the value of the maximum flow in the initial network is not greater than u .
- (b) If the lower bound of an arc $(i_1, j_1) \in A$ is decreased with the value $v \geq 0$ then the difference between the value of the maximum flow in the modified network and the value of the maximum flow in the initial network is not greater than v .

Proof. Let us consider a maximum flow f in the network G and $[X, \bar{X}]$ a minimum cut in G .

(a) After the capacity of the arc $(i_0, j_0) \in A$ is increased with the value $u \geq 0$, the capacity vector c' is obtained in (17):

$$c'(i, j) = \begin{cases} c(i, j) + u, & \text{if } (i, j) = (i_0, j_0) \\ c(i, j), & \text{if } (i, j) \neq (i_0, j_0). \end{cases} \quad (17)$$

Let f' be a maximum flow in the modified network $G' = (V, A, c', l' = l)$. It is easy to observe that the existence of a feasible flow (and a maximum flow) in G' is assured by the fact that $c \leq c'$ and by the existence of a maximum flow in G .

Of course, we have two possible situations: The arc (i_0, j_0) is in the direct set of arcs of a minimum cut $[X, \bar{X}]$ in G or not.

Case a.1. There is a minimum cut $[X, \bar{X}]$ in G so that the arc $(i_0, j_0) \in (X, \bar{X})$.

Let $[X', \bar{X}']$ be a minimum cut in G' . It is easy to see that:

$$\begin{aligned} v(f') &= c'[X', \bar{X}'] \leq c'[X, \bar{X}] = c'(X, \bar{X}) - l'(\bar{X}, X) = \\ &= c(X, \bar{X}) + u - l(\bar{X}, X) = c[X, \bar{X}] + u = v(f) + u \end{aligned} \quad (18)$$

Case a.2. There is no minimum cut $[X, \bar{X}]$ in G so that the arc $(i_0, j_0) \in (X, \bar{X})$.

Let $[X, \bar{X}]$ be a minimum cut in G . Of course, $(i_0, j_0) \notin (X, \bar{X})$. We shall prove next that $[X, \bar{X}]$ is a minimum cut in G' .

Let $[X', \bar{X}']$ be a cut in G' . It follows that $c[X', \bar{X}'] = c'[X', \bar{X}']$ (if $(x_0, y_0) \notin (X', \bar{X}')$) and $c[X', \bar{X}'] = c'[X', \bar{X}'] - u \leq c'[X', \bar{X}']$ (if $(x_0, y_0) \in (X', \bar{X}')$). Therefore, $c[X', \bar{X}'] \leq c'[X', \bar{X}']$ and, since $c'[X, \bar{X}] = c[X, \bar{X}] \leq c[X', \bar{X}']$ ($(i_0, j_0) \notin (X, \bar{X})$ and $[X, \bar{X}]$ is a minimum cut in G), it results that $c'[X, \bar{X}] \leq c'[X', \bar{X}']$. It follows that $[X, \bar{X}]$ is the minimum cut in G' because $[X', \bar{X}']$ is the arbitrarily chosen cut in G' .

Since $[X, \bar{X}]$ is a minimum cut in both networks, G and G' , and $(i_0, j_0) \notin (X, \bar{X})$ it results in equality (19):

$$v(f') = c'[X, \bar{X}] = c[X, \bar{X}] = v(f) \leq v(f) + u. \quad (19)$$

Therefore, in both cases, relation (20) was obtained:

$$v(f') - v(f) \leq u. \quad (20)$$

(b) After the lower bound of the arc $(i_1, j_1) \in A$ is decreased with the value $v \geq 0$, the following lower bound vector l' is obtained in (21):

$$l'(i, j) = \begin{cases} l(i, j) - v, & \text{if } (i, j) = (i_1, j_1) \\ l(i, j), & \text{if } (i, j) \neq (i_1, j_1). \end{cases} \quad (21)$$

A modified network $G' = (V, A, c' = c, l')$ is considered now and f' is a maximum flow in G' . The existence of a feasible flow (and a maximum flow) in G' is assured by the fact that $l \geq l'$ and by the existence of a maximum flow in G .

We have two possible situations: The arc (i_1, j_1) is in the inverse set of arcs of a minimum cut $[X, \bar{X}]$ in G or not.

Case b.1. There is a minimum cut $[X, \bar{X}]$ in G so that the arc $(i_1, j_1) \in (\bar{X}, X)$.

Let $[X', \bar{X}']$ be a minimum cut in G' . It is easy to see that:

$$\begin{aligned} v(f') &= c'[X', \bar{X}'] \leq c'[X, \bar{X}] = c'(X, \bar{X}) - l'(\bar{X}, X) = \\ &= c(X, \bar{X}) - (l(\bar{X}, X) - v) = c[X, \bar{X}] + v = v(f) + v \end{aligned} \quad (22)$$

Case b.2. There is no minimum cut $[X, \bar{X}]$ in G so that the arc $(i_1, j_1) \in (\bar{X}, X)$.

Let $[X, \bar{X}]$ be a minimum cut in G . Of course, $(i_1, j_1) \notin (X, \bar{X})$. We shall prove next that $[X, \bar{X}]$ is a minimum cut in G' .

Let $[X', \bar{X}']$ be a cut in G' . It follows that $c[X', \bar{X}'] = c'[X', \bar{X}']$ (if $(x_1, y_1) \notin (X', \bar{X}')$) or $c[X', \bar{X}'] = c'[X', \bar{X}'] - v \leq c'[X', \bar{X}']$ (if $(x_1, y_1) \in (X', \bar{X}')$). Therefore, $c[X', \bar{X}'] \leq c'[X', \bar{X}']$ and, since $c'[X, \bar{X}] = c[X, \bar{X}] \leq c[X', \bar{X}']$ ($(i_1, j_1) \notin (X, \bar{X})$ and $[X, \bar{X}]$ is a minimum cut in G), it results that $c'[X, \bar{X}] \leq c'[X', \bar{X}']$. It follows that $[X, \bar{X}]$ is the minimum cut in G' because $[X', \bar{X}']$ is the arbitrarily chosen cut in G' .

Since $[X, \bar{X}]$ is a minimum cut in both networks, G and G' , and $(i_1, j_1) \notin (\bar{X}, X)$ it results in equality (23):

$$v(f') = c'[X, \bar{X}] = c[X, \bar{X}] = v(f) \leq v(f) + v. \quad (23)$$

Therefore, in both cases, b.1 and b.2, relation (24) was obtained:

$$v(f') - v(f) \leq v. \quad (24)$$

□

Theorem 3. The pair of vectors (c^*, l^*) given in (25) and (26) is the optimum solution of IMCUL1, where:

$$c^*(i, j) = \begin{cases} f(i, j), & \text{if } (i, j) \in (X, \bar{X}) \\ c(i, j), & \text{if } (i, j) \in A - (X, \bar{X}). \end{cases} \quad (25)$$

$$l^*(i, j) = \begin{cases} f(i, j), & \text{if } (i, j) \in (\bar{X}, X) \\ l(i, j), & \text{if } (i, j) \in A - (\bar{X}, X). \end{cases} \quad (26)$$

Proof. Let (c^{**}, l^{**}) be an optimum solution of IMCUL1. A maximum flow f^{**} is considered in the network $G^{**} = (V, A, c^{**}, l^{**})$.

Using Lemma 1 and from the fact that (c^{**}, l^{**}) is an optimum solution of IMCUL1 the relations from (27) are obtained:

$$\begin{cases} c^{**}(i, j) \geq c(i, j), & \text{if } (i, j) \in A - (X, \bar{X}) \\ l^{**}(i, j) \leq l(i, j), & \text{if } (i, j) \in A - (\bar{X}, X). \end{cases} \quad (27)$$

Therefore, if the upper bounds on arcs $(i, j) \in A - (X, \bar{X})$ or the lower bounds of $(i, j) \in A - (\bar{X}, X)$ are modified then the value of the maximum flow is increased. Using Lemma 2 it results that:

$$\sum_{(i,j) \in A - (\bar{X}, X)} |l^{**}(i, j) - l(i, j)| + \sum_{(i,j) \in A - (X, \bar{X})} |c^{**}(i, j) - c(i, j)| \geq v(f^{**}) - v(f). \quad (28)$$

Using Lemma 1 it follows that:

$$\begin{aligned} \|l^{**} - l\|_1 + \|c^{**} - c\|_1 &= \sum_{(i,j) \in A} |l^{**}(i, j) - l(i, j)| + \sum_{(i,j) \in A} |c^{**}(i, j) - c(i, j)| \geq \\ &\sum_{(i,j) \in (\bar{X}, X)} |l^{**}(i, j) - l(i, j)| + \sum_{(i,j) \in (X, \bar{X})} |c^{**}(i, j) - c(i, j)| + v(f^{**}) - v(f) = \\ &\sum_{(i,j) \in (\bar{X}, X)} (l^{**}(i, j) - l(i, j)) + \sum_{(i,j) \in (X, \bar{X})} (c(i, j) - c^{**}(i, j)) + v(f^{**}) - v(f) = \\ &c[X, \bar{X}] - c^{**}[X, \bar{X}] + v(f^{**}) - v(f) = c[X, \bar{X}] - v(f). \end{aligned} \quad (29)$$

Using the definition of c^* and l^* (see (25) and (26)) we have:

$$\begin{aligned} \|l^* - l\|_1 + \|c^* - c\|_1 &= \sum_{(i,j) \in (\bar{X}, X)} |f(i, j) - l(i, j)| + \sum_{(i,j) \in (X, \bar{X})} |f(i, j) - c(i, j)| = \\ &\sum_{(i,j) \in (\bar{X}, X)} (f(i, j) - l(i, j)) + \sum_{(i,j) \in (X, \bar{X})} (c(i, j) - f(i, j)) = \\ &c[X, \bar{X}] - l[\bar{X}, X] - (f[X, \bar{X}] - f[\bar{X}, X]) = c[X, \bar{X}] - v(f). \end{aligned} \quad (30)$$

From (29) and (30) it results that:

$$\|l^{**} - l\|_1 + \|c^{**} - c\|_1 \geq \|l^* - l\|_1 + \|c^* - c\|_1. \quad (31)$$

Using the definition of c^* and l^* (see (25) and (26)) we have:

$$\begin{aligned} c^*[X, \bar{X}] &= c^*(X, \bar{X}) - l^*(X, \bar{X}) = \sum_{(i,j) \in (X, \bar{X})} c^*(i, j) - \sum_{(i,j) \in (\bar{X}, X)} l^*(i, j) = \\ &\sum_{(i,j) \in (X, \bar{X})} f(i, j) - \sum_{(i,j) \in (\bar{X}, X)} f(i, j) = f[X, \bar{X}] = v(f). \end{aligned}$$

From this equality and from the fact that f is a feasible flow in $G^* = (V, A, c^*, l^*)$ it results that $[X, \bar{X}]$ is the minimum cut in G^* and the feasible solution of IMCUL1. It results that:

$$\|l^{**} - l\|_1 + \|c^{**} - c\|_1 = \|l^* - l\|_1 + \|c^* - c\|_1. \quad (32)$$

This means that (c^*, l^*) is an optimum solution of IMCUL1. \square

Corollary 1. *If (c^{**}, l^{**}) is an optimum solution of IMCUL1, then the distance between (c^{**}, l^{**}) and the initial vector (c, l) is given in (33):*

$$\|l^{**} - l\|_1 + \|c^{**} - c\|_1 = c[X, \bar{X}] - v(f). \quad (33)$$

Proof. It is directly from (30) and (32). \square

Corollary 2. *If (c^{**}, l^{**}) is an optimum solution of IMCUL1, it is given in (34):*

$$\sum_{(i,j) \in A - (\bar{X}, X)} |l^{**}(i, j) - l(i, j)| + \sum_{(i,j) \in A - (X, \bar{X})} |c^{**}(i, j) - c(i, j)| = v(f^{**}) - v(f). \quad (34)$$

Proof. Using Corollary 1, inequality (29) becomes equality and then inequality (28) becomes also an equality. \square

From Theorem 3 the following algorithm denoted AIMCUL1 is obtained to solve IMCUL1:

AIMCUL1

Calculate a maximum flow f^* in G ;

Using (25) and (26) construct (c^*, l^*) ;

(c^*, l^*) is an optimum solution of IMCUL1.

The maximum flow problem has been intensively studied over decades [13]. A lot of algorithms to solve it have been developed. The first algorithm was proposed by Ford and Fulkerson in 1956. As long as there is an augmenting path in the network, the flow is increased along this path. This algorithm is not polynomial since its time complexity linearly depends on the value of the maximum flow. However, it is the base idea for other algorithms such as those due to Edmonds-Karp or Dinic which are polynomial (the time complexity is polynomial in the number of nodes and the number of arcs). There are also other polynomial approaches based on maintaining a preflow (push-relabel maximum flow). The best current known algorithm was published in 2013 by James Orlin [14]. This algorithm solves the maximum flow problem as a sequence of improvement phases. A strongly polynomial time algorithm was obtained by replacing the residual network of the Δ -improvement phase by a more compact representation. The author proved that the maximum flow can be computed in $O(n \cdot m)$ time or even $O(n^2 / \log(n))$ if $m = O(n)$, where n is the number of nodes ($n = |V|$) and m is the number of arcs ($m = |A|$).

Theorem 4. *The time complexity of AIMCUL1 is $O(n \cdot m)$.*

Proof. The time complexity of AIMCUL1 is given by the time complexity of the algorithm used to calculate the maximum flow f . \square

4. Example

We shall give an example to illustrate how AIMCUL1 works. In Figure 1, a network G and a given $s - t$ cut $[X, \bar{X}]$ in G are presented. The maximum flow f^* is calculated in G (see Figure 2) using any known algorithm briefly presented before theorem 4. The optimum solution (c^*, l^*) is presented in

Figure 3. The upper bounds of the arcs $(3, 6)$ and $(7, 6)$ were modified from 6 to 5 and, respectively, from 3 to 1 according to formula (25). The lower bound of the arc $(2, 3)$ was modified from 1 to 3 using formula (26). The total amount of modifications brought to the boundaries is $(6 - 5) + (3 - 1) + (3 - 1) = 5$. Therefore, $[X, \bar{X}]$ becomes a minimum $s - t$ cut in $G^* = (V, A, c^*, l^*, s, t)$.

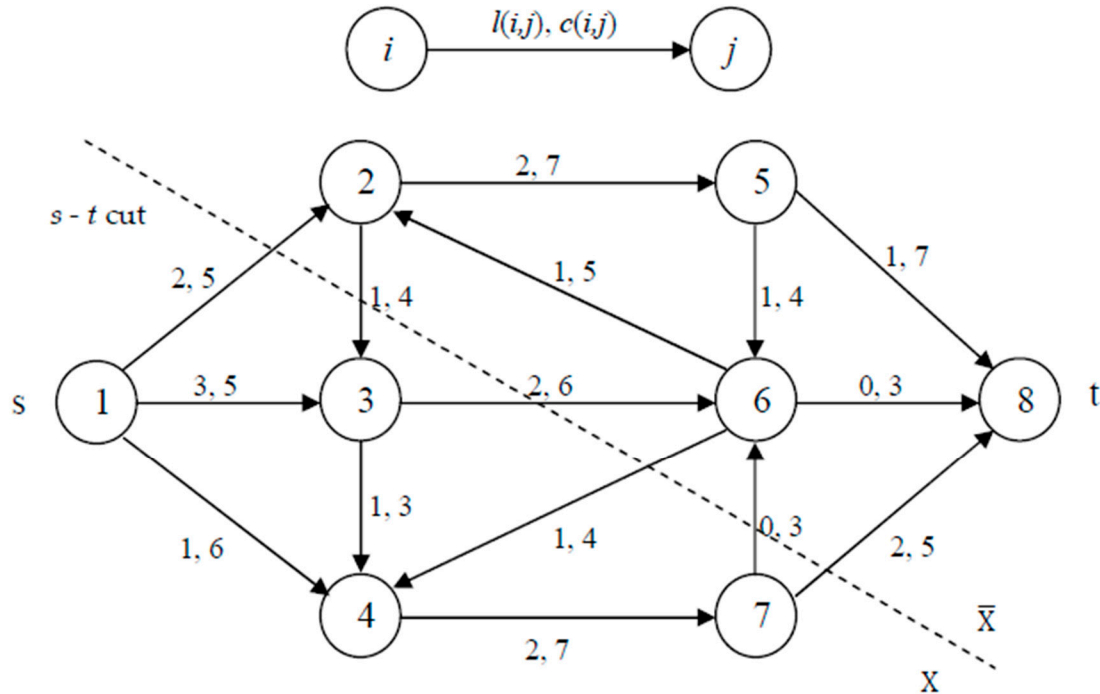


Figure 1. Initial network G and $s - t$ cut $[X, \bar{X}]$.

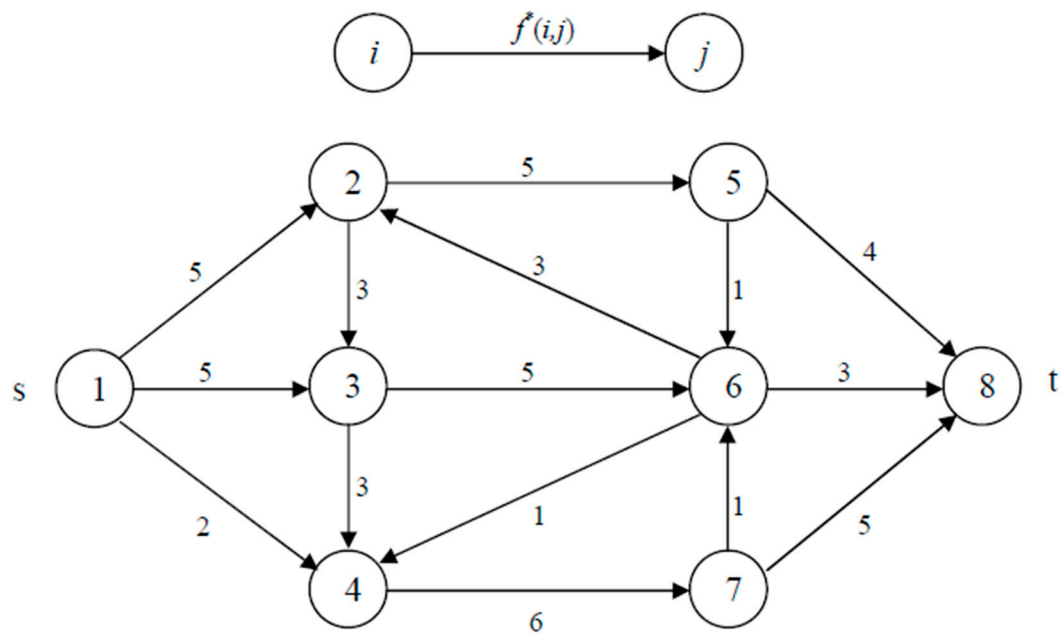


Figure 2. Maximum flow f^* in G .

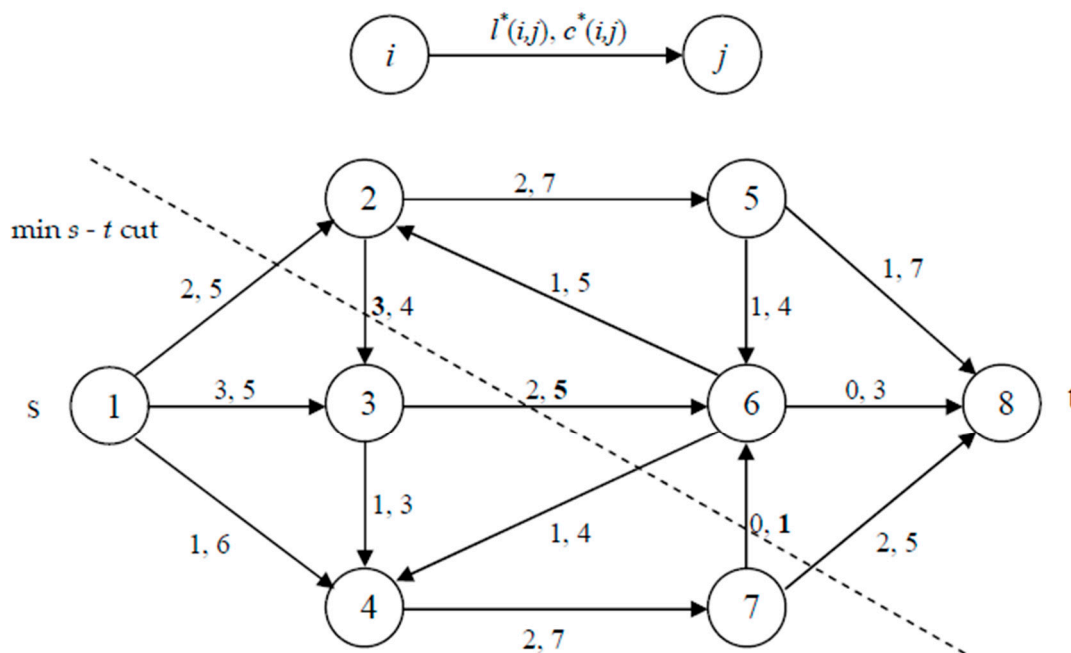


Figure 3. Optimum solution of inverse minimum cut problem with lower and upper bounds 1 (IMCUL1).

5. Conclusions

An efficient strongly polynomial algorithm was deduced to solve IMCUL1. Although the direct problem of minimum cut with lower and upper bounds is a generalization of the problem of minimum cut with only upper bounds, the corresponding inverse problem IMCUL1 is a different problem, which is not a generalization of IMC since in the case of IMC lower bounds cannot be modified. An example to illustrate the proposed algorithm for IMCUL1 has been presented.

As a future work, the inverse minimum cut with lower and upper bounds under other norms and distances (such as l_∞ , l_2 or Hamming distances) could be considered.

Author Contributions: Conceptualization, A.D.; methodology, L.C.; validation, L.C.; formal analysis, A.D.; writing—original draft preparation, A.D.; writing—review and editing, L.C. and A.D.; funding acquisition, A.D. and L.C. All authors have read and agreed to the published version of the manuscript.

Funding: The APC was funded by Transilvania University of Brasov.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ahuja, R.K.; Orlin, J.B. Combinatorial algorithms for inverse network flow problems. *Networks* **2002**, *40*, 181–187. [\[CrossRef\]](#)
2. Ahuja, R.K.; Orlin, J.B. Inverse Optimization. *Oper. Res.* **2001**, *49*, 771–783. [\[CrossRef\]](#)
3. Demange, M.; Monnot, J. An introduction to inverse combinatorial problems. In *Vangelis Th. Paschos, Paradigms of Combinatorial Optimization (Problems and New Approaches)*; Wiley: London, UK; Hoboken, NJ, USA, 2010.
4. Heuberger, C. Inverse Combinatorial Optimization: A Survey on Problems, Methods, and Results. *J. Comb. Optim.* **2004**, *8*, 329–361. [\[CrossRef\]](#)
5. Yang, C.; Zhang, J.; Ma, Z. Inverse maximum flow and minimum cut problems. *Optimization* **1997**, *40*, 147–170. [\[CrossRef\]](#)
6. Deaconu, A. The inverse maximum flow problem considering L_∞ norm. *RAIRO Oper. Res.* **2008**, *42*, 401–414. [\[CrossRef\]](#)
7. Tayyebi, J.; Deaconu, A. Inverse Generalized Maximum Flow Problems. *Mathematics* **2019**, *7*, 899. [\[CrossRef\]](#)
8. Tayyebi, J.; Mohammadi, A.; Kazemi, S.M.R. Reverse maximum flow problem under the weighted Chebyshev distance. *RAIRO Oper. Res.* **2018**, *52*, 1107–1121. [\[CrossRef\]](#)

9. Liu, L.; Zhang, J. Inverse maximum flow problems under the weighted Hamming distance. *J. Comb. Optim.* **2006**, *12*, 395–408. [[CrossRef](#)]
10. Deaconu, A. The inverse maximum flow problem with lower and upper bounds for the flow. *Yugosl. J. Oper. Res.* **2008**, *18*, 13–22. [[CrossRef](#)]
11. Ciurea, E.; Deaconu, A. Inverse minimum flow problem. *J. Appl. Math. Comput.* **2007**, *23*, 193–203. [[CrossRef](#)]
12. Zhang, J.; Cai, M.-C. Inverse problem of minimum cuts. *ZOR-Math. Methods Oper. Res.* **1998**, *47*, 51–58. [[CrossRef](#)]
13. Smith, D.K.; Ahuja, R.K.; Magnanti, T.L.; Orlin, J.B. *Network Flows: Theory, Algorithms, and Applications*; Prentice Hall: Englewood Cliffs, NJ, USA, 1993.
14. Orlin, J.B. Max flows in $O(nm)$ time, or better. In Proceedings of the Forty-fifth Annual ACM Symposium on Theory of Computing, Palo Alto, CA, USA, 2–4 June 2013; pp. 765–774.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).