On DC. optimization algorithms for solving minmax flow problems *

Le Dung Muu
Institute of Mathematics, Hanoi
Tel.: +84 4 37563474   Fax: +84 4 37564303
Email: ldmuu@math.ac.vn
Le Quang Thuy
Faculty of Applied Mathematics and Informatics,
Hanoi University of Science and Technology

June 17, 2011

Abstract

We formulate minmax flow problems as a DC. optimization problem. We then apply a DC. primal-dual algorithm to solve the resulting problem. The obtained computational results show that the proposed algorithm is efficient thanks to particular structures of the minmax flow problems. *Keywords:* Minmax flow problem smooth DC optimization regularization.

1 Introduction

In the minimax flow problem to be considered, we are given a directed network flow $N(V, E, s, t, p)$, where V is the set of $m+2$ nodes, E is the set of n arcs, s is the single source node, t is the single sink node, and $p \in \mathbb{R}^n$ is the vector of arc capacities. Let $\partial^+ : E \to V$ and $\partial^- : E \to V$ be incidence

*This paper is supported by the Vietnam National Foundation for Science Technology Development

functions. When $h = (u, v)$, i.e., arc h leaves node u and enters node v, we write $\partial^+ h = u$ and $\partial^- h = v$. A vector $x \in \mathbb{R}^n$ is said to be a *feasible flow* if it satisfies the capacity constraints:

$$0 \leq x_h \leq p_h \ \forall h \in E \tag{1.1}$$

and conservation equations

$$\sum_{\partial^+ h = v} x_h = \sum_{\partial^- h = v} x_h \ \forall \ v \in V \backslash \{s, t\}. \tag{1.2}$$

Note that conservation equations can be simply written as

$$Ax = 0,$$

where the $m \times n$ matrix $A = (a_{vh})$ is the well-known node-arc incidence matrix, whose, for each $(v, h) \in V \backslash \{s, t\} \times E$, the entry a_{vh} is defined as

$$a_{vh} = \begin{cases} 1 & \text{if arc } h \text{ leaves node } v, \text{ i.e., } \partial^+ h = v \\ -1 & \text{if arc } h \text{ enters node } v, \text{ i.e., } \partial^- h = v \\ 0 & \text{otherwise.} \end{cases}$$

Then, the constraint (1.1), (1.2) becomes

$$Ax = 0, \ 0 \leq x \leq p.$$

Let X denote the set of feasible flows, i.e.,

$$X = \{x \in \mathbb{R}^n : Ax = 0, \ 0 \leq x \leq p\}.$$

For each $x \in X$, the value of the flow x is given by

$$d^T x = \sum_{\partial^+ h = s} x_h - \sum_{\partial^- h = s} x_h,$$

where d is a n - dimensional row vector defined as

$$d_h = \begin{cases} 1 & \text{if arc } h \text{ leaves the source node } s, \text{ i.e., } \partial^+ h = s \\ -1 & \text{if arc } h \text{ enters the source node } s, \text{ i.e., } \partial^- h = s \\ 0 & \text{otherwise.} \end{cases}$$

A feasible flow $x^0 \in X$ is said to be a *maximal flow* if there is no feasible flow $x \in X$ such that $x \geq x^0$ and $x \neq x^0$. We use X_E to denote the set of maximal flows. The problem of finding a minimal value flow on the set of all maximal flows, shortly (*minmax flow problem*), to be solved in this paper can be given as

$$\begin{array}{l} \min \ d^T x \\ \text{subject to } x \in X_E. \end{array} \tag{1.3}$$

Minmax flow problem was considered (see e.g. [8], [17], [18], [21]) and is closely related to the uncontrollable flow raised by Iri (see e.g. [12], [13]). Note that since the set X_E, in general, is nonconvex, Problem (1.3), is a nonconvex optimization one.

Some global optimization algorithms are proposed for solving minmax flow problems (see e.g. [8], [17], [18], [21]). These algorithms can solve minmax flow problems where the number of the criteria is relatively small. However, in minmax flow problems, the number of the criteria is just equal to the number of decision variables that often is large in practical applications. To this case, local optimization approaches should be used.

Recently local optimization approaches to DC. mathematical programming problems have been well developed. A well known primal-dual DC. algorithm, called DCA, introduced by P.D. Tao (see e.g. [16]) and further developed by L.T.H.An and P.D.Tao, has been successfully applied to solve a lot of practical problems (see e.g. [4], [16] and the refences therein).

In this paper, first we formulate the mimax flow problem described above as a smoothly DC. optimization problem by using a regularization technique widely used in variational inequality. Then we apply the DCA algorithm to solve the resulting DC. optimization problem. The main advantage of this algorithm is that the subproblems that we need to solve at each iteration of DCA are strongly convex quadratic programs rather than general convex programs as in the general case.

The paper is organized as follows. In Section 2, first we show how to use the Yoshida regularization technique to obtain a smoothly DC. optimization formulation for the minmax flow problem. Then we describe the DCA algorithm for the resulting DC. program to the mimax flow problem. Computational results reported in the last section show that DCA is efficient for minmax flow problems.

2 A Smoothly DC. Optimization Formulation

Clearly, X_E is the set of all Pareto efficient solutions to the multiple objective linear program

$$\begin{array}{l} \text{Vmax } x \\ \text{subject to } x \in X. \end{array} \tag{2.4}$$

As we have seen in the introduction part, for the mimax flow problem the number of the decision variables is just equal to the number of the criteria.

From Philip [14] it is known that one can find a simplex $\Lambda \subset \mathbb{R}^n_{++}$ such that a point $x \in X_E$ if and only if there exists $\lambda \in \Lambda$ such that

$$x \in \text{argmax}\{\lambda^T y \ : y \in X\}.$$

Thus,

$$X_E = \{x \in X : \lambda^T x \geq \phi(\lambda) \text{ for all } \lambda \in \Lambda\},$$

where

$$\phi(\lambda) = \max\{\lambda^T y : y \in X\}.$$

In our case, the simplex Λ can be defined explicitly as in the following theorem.

Theorem 2.1. *([17]) Let Λ be one of the following simplices*

$$\Lambda := \{\lambda \in \mathbb{Z}^n : e \leq \lambda \leq ne\}$$

or

$$\Lambda = \{\lambda \in \mathbb{R}^n : \lambda \geq e; \sum_{k=1}^{n} \lambda_k = n^2\},$$

where e is the vector whose every entry is one. Then x is a maximal flow if and only if there is $\lambda \in \Lambda$ such that

$$x \in \text{argmin}\{\lambda^T y : y \in X\}.$$

Let $c > 0$, plays as a regularization parameter, and $K = \Lambda \times X$. For each $(\lambda, x) \in K$, we define

$$\gamma_c(\lambda, x) := \max_{y \in K}\{\langle -\lambda, x - y\rangle - \frac{c}{2}\|x - y\|^2\}. \tag{2.5}$$

Since the objective function $\langle -\lambda, x - y\rangle - \frac{c}{2}\|x - y\|^2$ of problem (2.5) is strongly concave on K with respect to y, the problem (2.5) defining $\gamma_c(\lambda, x)$ has a unique solution that we denote by $y_c(\lambda, x) \in K$.

Proposition 2.1. *For every $(\lambda, x) \in \mathbb{R}^{p+n}$ we have:*

(i) $y_c(\lambda, x) = P_K\left(x + \frac{1}{c}\lambda\right)$*, where* $P_K(x + \frac{1}{c}\lambda)$ *is the projection of* $(x + \frac{1}{c}\lambda)$ *on the set* K*;*

(ii) $\gamma_c(\lambda, x) \geq 0 \ \forall \ (\lambda, x) \in K$*,* $\gamma_c(\lambda, x) = 0, (\lambda, x) \in K$ *if and only if* $x \in X_E$*;*

(iii) $\gamma_c(.,.)$ *is continuously differentiable on* K *and its gradient*

$$\nabla\gamma_c(\lambda, x) = -[x - y_c(\lambda, x), \lambda + c(x - y_c(\lambda, x))]^T.$$

Proof. (i) Since

$$\begin{aligned}
&\frac{1}{2c}\|\lambda\|^2 - \frac{c}{2}\|x + \frac{1}{c}\lambda - y\|^2 \\
&= \frac{1}{2c}\|\lambda\|^2 - \frac{c}{2}(\|x - y\|^2 + \frac{2}{c}\langle\lambda, x - y\rangle + \frac{1}{c^2}\|\lambda\|^2) \\
&= \langle -\lambda, x - y\rangle - \frac{c}{2}\|x - y\|^2,
\end{aligned}$$

we have

$$\langle -\lambda, x - y\rangle - \frac{c}{2}\|x - y\|^2 = \frac{1}{2c}\|\lambda\|^2 - \frac{c}{2}\|x + \frac{1}{c}\lambda - y\|^2.$$

Then

$$\begin{aligned}
\gamma_c(\lambda, x) &= \max_{y\in K}\{\langle -\lambda, x - y\rangle - \frac{c}{2}\|x - y\|^2, \ (\lambda, x) \in \mathbb{R}^{2n}\} \\
&= \max_{y\in K}\{\frac{1}{2c}\|\lambda\|^2 - \frac{c}{2}\|x + \frac{1}{c}\lambda - y\|, \ (\lambda, x) \in \mathbb{R}^{2n}\} \\
&= \frac{1}{2c}\|\lambda\|^2 + \max_{y\in K}\{-\frac{c}{2}\|x + \frac{1}{c}\lambda - y\|, \ (\lambda, x) \in \mathbb{R}^{2n}\} \\
&= \frac{1}{2c}\|\lambda\|^2 - \frac{c}{2}\min_{y\in K}\{\|x + \frac{1}{c}\lambda - y\|, \ (\lambda, x) \in \mathbb{R}^{2n}\}
\end{aligned}$$

which implies that $y_c(\lambda, x) = P_K(x + \frac{1}{c}\lambda)$.

(ii) Clearly, by the definition of $\gamma_c(\lambda, x)$, we have $\gamma_c(\lambda, x) \geq 0$ for every $(\lambda, x) \in K$. Let $(\lambda^*, x^*) \in K$ such that $\gamma_c(\lambda^*, x^*) = 0$. Then

$$\langle -\lambda^*, x^* - y_c(\lambda^*, x^*) \rangle - \frac{c}{2}\|x^* - y_c(\lambda^*, x^*)\|^2 = 0,$$

which implies

$$\langle -\lambda^*, x^* - y_c(\lambda^*, x^*) \rangle = \frac{c}{2}\|x^* - y_c(\lambda^*, x^*)\|^2 \geq 0. \tag{2.6}$$

On the other hand, by the well known optimality condition for the convex program applying to the problem defining $\gamma_c(\lambda^*, x^*)$, we can write

$$\langle -\lambda^* + c[y_c(\lambda^*, x^*) - x^*], x - y_c(\lambda^*, x^*) \rangle \geq 0 \ \forall \ (\lambda, x) \in K.$$

With $x = x^*$ we have

$$\langle -\lambda^* + c[y_c(\lambda^*, x^*) - x^*], x^* - y_c(\lambda^*, x^*) \rangle \geq 0.$$

Thus

$$\langle -\lambda^*, x^* - y_c(\lambda^*, x^*) \rangle - c\|x^* - y_c(\lambda^*, x^*)\|^2 \geq 0.$$

Combining this inequality with (2.6) we can deduce that $\|x^* - y_c(\lambda^*, x^*)\| \leq 0$. Hence $x^* = y_c(\lambda^*, x^*)$. By *(i)*, we have

$$x^* = y_c(\lambda^*, x^*) = P_K(x^* + \frac{1}{c}\lambda^*).$$

On the other hand, according to properties of the projection $P_K(.)$ we know, that $x^* = P_K(x)$ if and only if $\langle y - x^*, x^* - x \rangle \geq 0$ for all $y \in K$. From the last inequality, by replacing $x = x^* + \frac{1}{c}\lambda^*$ we obtain $\langle -\lambda^*, y - x^* \rangle \geq 0$ for all $y \in K$, which implies $x^* \in X_E$.

Conversely, suppose that $x^* \in X_E$. Then $\langle -\lambda^*, y - x^* \rangle \geq 0$ for all $y \in K$.

Note that

$$\begin{aligned} \gamma_c(\lambda^*, x^*) &= \max_{y \in K}\{\langle -\lambda^*, x^* - y \rangle - \frac{c}{2}\|x^* - y\|^2\} \\ &= \max_{y \in K}\{-\langle -\lambda^*, y - x^* \rangle - \frac{c}{2}\|x^* - y\|^2\} \leq 0. \end{aligned}$$

Thus $\gamma_c(\lambda^*, x^*) = 0$.

(iii) Since $y_c(\lambda, x) = P_K(x + \frac{1}{c}\lambda)$, by continuity of the projection, $y_c(.,.)$ is continuous on K. Then, from $\gamma_c(\lambda, x) = \langle -\lambda, x - y_c(\lambda, x)\rangle - \frac{c}{2}\|x - y_c(\lambda, x)\|^2$, we can see that $\gamma_c(\lambda, x)$ is continuous on K.

Let $u = (\lambda, x)$ and $\varphi(u, y) = \langle \lambda, x - y\rangle + \frac{c}{2}\|x - y\|^2$. Since the function $\varphi(u, .)$ is strongly convex and continuously differentiable with respect to the variable y, there exists the unique solution $y_c(\lambda, x)$ of the problem

$$\min_{(0,y)\in K} \varphi(u, y)$$

and the function $\gamma_c(\lambda, x)$ is continuously differentiable. A simple computation yields

$$\begin{aligned}\nabla\gamma_c(\lambda, x) &= -\nabla_u\varphi(u, y) = -\nabla_u\varphi(u, y_c(\lambda, x)) = -\nabla_{(\lambda,x)}\varphi((\lambda, x), y_c(\lambda, x)) \\ &= -[x - y_c(\lambda, x), \lambda + c(x - y_c(\lambda, x))]^T.\end{aligned}$$

□

By virture of this proposition the problem (1.3) can be written equivalently as

$$\begin{aligned}&\min\ d^T x \\ &\text{subject to } \begin{cases} (\lambda, x) \in K \\ \gamma_c(\lambda, x) = 0. \end{cases}\end{aligned} \tag{2.7}$$

A simple computation shows that $\gamma_c(\lambda, x) = g(\lambda, x) - h(\lambda, x)$, where

$$g(\lambda, x) = \frac{1}{2}\|x\|^2 + \frac{1}{2}\|\lambda\|^2 + \max_{y\in X}\{\langle\lambda + cx, y\rangle - \frac{c}{2}\|y\|^2\},$$

$$h(\lambda, x) = \frac{1}{2}\|\lambda + x\|^2 + \frac{c}{2}\|x\|^2.$$

It is easy to see that both g and h are convex. Since the objective function of the maximization problem

$$\max_{y\in X}\{\langle\lambda + cx, y\rangle - \frac{c}{2}\|y\|^2, (\lambda, x) \in \mathbb{R}^{2n}\}$$

is strongly concave, the function g is differentiable. Clearly, h is differentiable. Thus the problem (2.7) can be converted into the DC constrained problem

$$\min\{d^T x : g(\lambda, x) - h(\lambda, x) \le 0, (\lambda, x) \in K\}. \tag{2.8}$$

For $t > 0$, we consider the penalized problem

$$\min\{f_t(\lambda, x) := d^T x + tg(\lambda, x) - th(\lambda, x) : (\lambda, x) \in K\}. \tag{2.9}$$

From [3] we know that there exists a exact penalty parameter $t_0 \geq 0$ such that for every $t > t_0$, the solution sets of problems (2.7) and (2.9) are identical.

In the minmax flow problem being consideration, we have $d = \sum_{h=1}^{n} \xi_h e^h$ where $\xi_h \in \{1, -1, 0\}$ and e^h denotes the h-th unit vector of $\mathbb{R}^n$. In this case, the exact penalty parameter t_0 can be calculated as follows ([3]:
Let $I := \{h : \xi_h > 0\}$. Define t_* by taking

$$0 < t_* \leq \max\{\xi_h : h \in I\} \text{ if } I \neq \emptyset \text{ and } t_* = 0 \text{ if } I = \emptyset.$$

Since $\xi_h \in \{-1, 0, 1\}$, we can take $0 < t_* \leq 1$ whenever $I \neq \emptyset$. So in this case by Proposition 2.4 in [3] we can take $0 < t_0 \leq 1$. Note that in the case when $I = \emptyset$, i.e., $t_* = 0$, it is easy to see that any optimal solution of the linear program $\min\{\sum_{h=1}^{n} \xi_h x_h : x \in X\}$ solves Problem (1.3). So in this case the minmax flow problem is reduced to the linear program $\min\{\sum_{h=1}^{n} \xi_h x_h : x \in X\}$.

We now use a DC. optimization algorithm, called DCA, developed in [2] to solve problem (2.9). The DCA can solve nonsmooth DC optimization problems of the form

$$\min\{F(u) := G(u) - H(u) : u \in \mathbb{R}^n\}$$

where G and H are lower semicontinuous proper convex functions. For this nonsmooth optimization problem, the sequence of iterates is constructed by taking

$$v^k \in \partial H(u^k), \ u^{k+1} = \operatorname{argmin}\{G(u) - \langle v^k, u\rangle\}.$$

Thus, convergence of this algorithm in this nonsmooth case crucially depends on the choice of $v^k \in \partial H(u^k)$. In our case, since H is differentiable, v^k is uniquely defined.

Clearly, problem (2.9) is a D.C. program. For this special problem, we can describe DCA as follows. For simple of notation we write $u = (\lambda, x)$.

Starting from a point $u^0 = (\lambda^0, x^0) \in K$. At each iteration $k = 0, 1, ...,$ having u^k we construct v^k and u^{k+1} by setting

$$v^k = \nabla th(u^k) = t[\lambda^k + x^k, \lambda^k + (1+c)x^k]^T,$$
$$u^{k+1} = \text{argmin}\{d^T x + tg(u) - \langle u - u^k, v^k\rangle : \; u \in K\}.$$

It is proved in [2], among others, that $f_t(u^{k+1}) \leq f_t(u^k)$ for every k, that if $u^{k+1} = u^k$, then u^k is a stationary point, and that any cluster point of the sequence $\{u^k\}$ is a stationary point to the problem (2.9). In the sequel we call u^k an ϵ-stationary point if $||u^{k+1} - u^k|| \leq \epsilon$.

Thanks to specific properties of the minmax flow problem under consideration, one can find a DC decomposition for the gap function γ_c such that the subproblems needed to solve are strongly convex quadratic. In fact we have the following result.

Theorem 2.2. *Let $\gamma(u)$ be defiend by (2.5), where $c > 0$ fixed. Then, for every real number ρ satisfying $c\rho \geq 1$, the function*

$$f_c(u) := \frac{1}{2}\rho||u||^2 - \gamma_c(u)$$

is convex on K.

Proof. By definition, the function $\gamma_c(u)$ can be written as

$$\gamma_c(u) = \frac{1}{2c}||\lambda||^2 - \frac{c}{2}\min_{y\in K}\{||x + \frac{1}{c}\lambda - y||^2\}.$$

Thus, we have

$$f_c(u) = \frac{1}{2}\rho||u||^2 - \frac{1}{2c}||\lambda||^2 + \frac{c}{2}\min_{y\in K}\{||x + \frac{1}{c}\lambda - y||^2\}.$$

Since $u = (\lambda, x)$,

$$f_c(u) = (\frac{1}{2}\rho - \frac{1}{2c})||\lambda||^2 + \frac{1}{2}\rho||x||^2 + \frac{c}{2}\min_{y\in K}\{||x + \frac{1}{c}\lambda - y||^2\}.$$

Clearly, if $(\frac{1}{2}\rho - \frac{1}{2c}) > 0$, then

$$q(u) = (\frac{1}{2}\rho - \frac{1}{2c})||\lambda||^2 + \frac{1}{2}\rho||x||^2$$

is a convex quadratic function.

Now we show that

$$p(u) := p(\lambda, x) = \frac{c}{2}\min_{y\in K}\{||x + \frac{1}{c}\lambda - y||^2\}$$

is convex on K.

Indeed, let $y^u, y^v \in K$ such that

$$p(u) = \frac{c}{2}||x^u + \frac{1}{c}\lambda^u - y^u||^2,$$

and

$$p(v) = \frac{c}{2}||x^v + \frac{1}{c}\lambda^v - y^v||^2.$$

Since K is convex, $\theta y^u + (1-\theta)y^v \in K$ whenever $0 \le \theta \le 1$.

On the other hand,

$$p(\theta u + (1-\theta)v) = \frac{c}{2}\min_{y\in K}\{||\theta x^u + (1-\theta)x^v + \frac{1}{c}(\theta\lambda^u + (1-\theta)\lambda^v) - y||^2\}.$$

Hence

$$p(\theta u+(1-\theta)v) \le \frac{c}{2}||\theta x^u+(1-\theta)x^v+\frac{1}{c}(\theta\lambda^u+(1-\theta)\lambda^v)-(\theta y^u+(1-\theta)y^v)||^2.$$

We observe that the inequality

$$\theta||x^u + \tfrac{1}{c}\lambda^u - y^u||^2 + (1-\theta)||x^v + \tfrac{1}{c}\lambda^v - y^v||^2 \ge$$
$$||\theta x^u + (1-\theta)x^v + \tfrac{1}{c}(\theta\lambda^u + (1-\theta)\lambda^v) - (\theta y^u + (1-\theta)y^v)||^2.$$

is equivalent to the one

$$\theta(1-\theta)\{||x^u+\frac{1}{c}\lambda^u-y^u||^2+||x^v+\frac{1}{c}\lambda^v-y^v||^2-2\langle x^u+\frac{1}{c}\lambda^u-y^u, x^v+\frac{1}{c}\lambda^v-y^v\rangle\} \ge 0.$$

It is easy to see that latter inequality is always holds true. Hence

$$\theta(1-\theta)||(x^u + \frac{1}{c}\lambda^u - y^u) - (x^v + \frac{1}{c}\lambda^v - y^v)||^2 \ge 0$$

which shows the convexity of f_c. □

From the result of Theorem 2.2, the problem (2.8) can be converted into the DC optimization problem

$$\alpha(t) := \min_{u\in K}\{G_t(u) = b^T u + \tfrac{1}{2}\rho||u||^2 - [\tfrac{1}{2}\rho||u||^2 - t\gamma_c(u)]\}, \tag{2.10}$$

where $u = (\lambda, x), b = (0, d)$.

For simplicity of notation, we write

$$G(u) = b^T u + \frac{1}{2}\rho||u||^2, \ H(u) = \frac{1}{2}\rho||u||^2 - t\gamma_c(u).$$

Then the problem has the form

$$\min\{G(u) - H(u) : u \in K\} \tag{2.11}$$

The sequences $\{u^k\}$ and $\{v^k\}$ constructed by DCA for the problem (2.11) now look as

$$v^k \in \ \nabla H(u^k) = \rho u^k - t\nabla\gamma_c(u^k),$$

$$u^{k+1} = \text{argmin}\{b^T u + \frac{1}{2}\rho||u||^2 - \langle u - u^k, v^k\rangle : \ u \in K\}.$$

Thus, at each iteration k we have to solve the two strongly convex quadratic programs

$$\min_{y\in K} \|y - (x^k + \frac{1}{c}\lambda^k)\|^2 \tag{2.12}$$

and

$$\min_{u\in K}\{b^T u + \frac{1}{2}\rho||u||^2 - \langle u - u^k, v^k\rangle\}. \tag{2.13}$$

Let $y_c(\lambda^k, x^k)$, be the unique solution of (2.12). Then

$$v^k = \rho u^k - t\nabla\gamma_c(u^k) = \rho(\lambda^k, x^k) + t[x^k - y_c(\lambda^k, x^k), \lambda + c(x^k - y_c(\lambda^k, x^k))]^T.$$

The DCA for this case can be described in details as follows.

Algorithm

- *Initialization*: Choose an exact penalty parameter $0 < t \leq 1$, a tolerance $\epsilon \geq 0$ and two positive numbers c and ρ such that $c\rho \geq 1$. Seek $u^0 \in K$.
  Let $k := 0$.

- *Iteration* k ($k = 0, 1, 2, ...$)
  ∗ Calculate $v^k = \rho u^k - t\nabla\gamma_c(u^k)$.
  ∗ Compute $u^{k+1} = \arg\min\{b^T u + \frac{1}{2}\rho||u||^2 - \langle u - u^k, v^k\rangle : \ u \in K\}$.
  If $||u^{k+1} - u^k|| \leq \epsilon$, then terminate: u^k is an ϵ-stationary point.
  ∗ Otherwise, go to iteration k with $k := k + 1$.

3 Illustrative Example and Computational Results

To illustrate the algorithm we take an example in [8]. In this example the network has $4+2$ nodes and 10 arcs as shown in Fig.1, where the number attached to each arc is the arc capacity given by
$p = (p_1, ..., p_{10})^T = (8, 3, 1, 4, 2, 1, 7, 1, 2, 8)^T$.

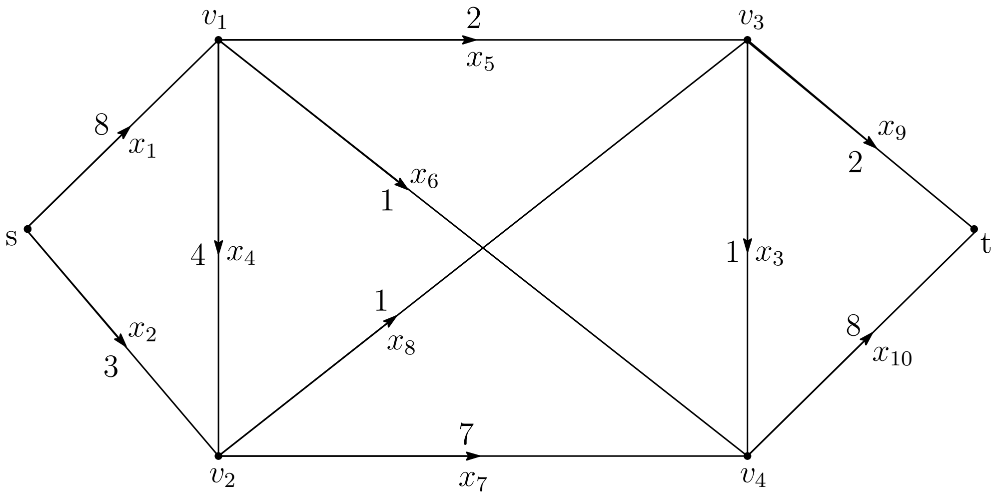

Fig.1. Network with 6 nodes and 10 arcs.

Initialization:
Compute an extreme maximal flow $u^0 = (\lambda^0, x^0)$, where
$\lambda^0 = (1.000, 1.000, 1.000, 1.000, 1.000, 1.000, 1.400, 1.000, 1.000, 90.600)$;
$x^0 = (7.000, 3.000, 0.067, 4.000, 2.000, 1.000, 6.933, 0.067, 2.000, 8.000)$.
Iteration k:
At $k = 1$, the iterate is $u^1 = (\lambda^1, x^1)$, where
$\lambda^1 = (1.000, 1.000, 1.000, 1.000, 1.000, 1.000, 1.720, 1.000, 1.000, 90.280)$,
$x^1 = (6.995, 3.000, 0.070, 4.000, 2.000, 0.995, 6.935, 0.065, 1.995, 8.000)$.
After 12 iterations, we obtain $u^* = (\lambda^*, x^*)$, where
$\lambda^* = (1.000, 1.000, 1.622, 1.000, 1.000, 1.000, 2.501, 1.000, 1.000, 88.877)$,
$x^* = (6.000, 3.000, 1.000, 4.000, 2.000, 0.000, 7.000, 0.000, 1.000, 8.000)$.
It is the same global optimal solution $x^* = (6, 3, 1, 4, 2, 0, 7, 0, 1, 8)$ that is obtained in [8]. It has been observed (see [4]) that for practical problems DCA often gives a global optimal solution. In Fig.2, the vector (x_i^*, p_i) is given on each arc.

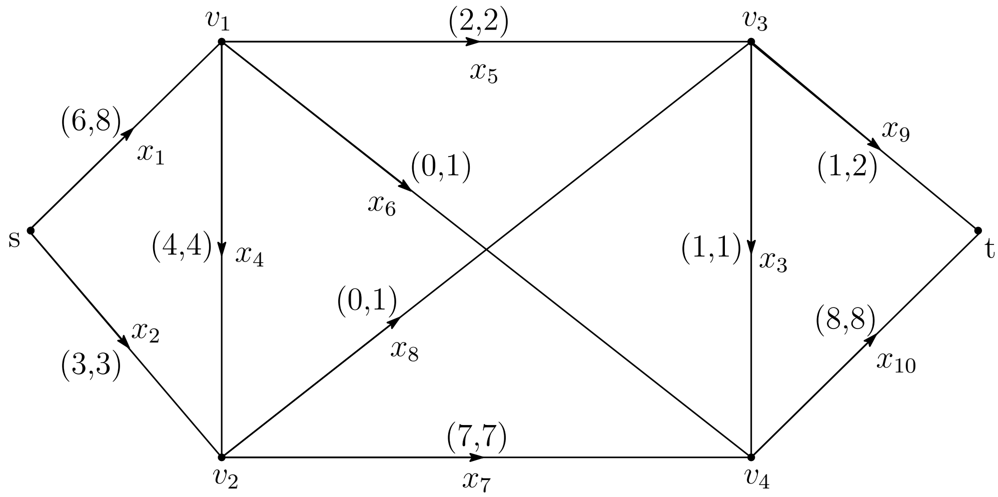

Fig.2. A minimum maximal flow (x_i^*, p_i).

To test the DCA algorithm, for each pair (m, n), we run it on 5 randomly different sets of data. These sets of data are chosen as in [21]. Test problems are executed on CPU, chip Intel Core(2) 2.53 GHz, RAM 2 GB, $C^{++}(VC^{++}2005)$ programming language. Numerical results are summarized in Table 1.

Table 1. Computational Results

m	n	Data	Op-value	Iter.	Value-γ_c	Time
6	10	$data_1$	7	7	3.02E-07	0.77
6	10	$data_2$	2	19	7.92E-07	0.86
6	10	$data_3$	5	12	1.95E-06	0.53
6	10	$data_4$	3	3	0.00E+00	0.19
6	10	$data_5$	8	4	0.00E+00	0.23
16	20	$data_6$	6	8	0.00E+00	0.55
16	20	$data_7$	4	16	4.46E-07	0.67
16	20	$data_8$	1	26	2.40E-07	1.05
16	20	$data_9$	0	21	4.41E-07	0.88
16	20	$data_{10}$	6	43	6.82E-07	1.76
30	70	$data_{11}$	3	116	1.06E-05	7.05
30	70	$data_{12}$	4	500	4.00E-05	30.58
30	70	$data_{13}$	7	165	8.00E-06	10.14
30	70	$data_{14}$	2	234	3.94E-06	14.31
30	70	$data_{15}$	6	242	1.13E-06	14.73
100	200	$data_{16}$	4	199	2.42E-07	41.73
100	200	$data_{17}$	1	191	1.52E-05	40.52
100	200	$data_{18}$	2	500	7.70E-06	105.74
100	200	$data_{19}$	4	390	2.37E-05	80.19
100	200	$data_{20}$	5	500	3.60E-05	103.09

where

Data: Sets of data;
Op-value: Optimal value;
Iter.: Number of iterations;
Value-γ_c: Value of $\gamma_c(.)$;
Time: Average CPU-Time in seconds.

4 Conclusions

We have used a regularization technique to formulate the minmax flow problem as a smoothly DC optimization problem. Then we have applied the DCA algorithm for solving the resulting DC problem. Computational results show that DCA is a good choice for the minimax flow problem due to specific properties of this problem.

References

[1] An L. T. H. and Tao P. D., The DC (Difference of convex functions) programming and DCA revisited with DC models of real world nonconvex optimization problems, Annals of Operations Research, 133, 23 - 46 (2005).

[2] An L. T. H., Tao P. D. and Muu L. D., Numerical solution for optimization over the efficient set by DC optimization algorithms, Operations Research Letters, 19, 117 - 128 (1996).

[3] An L. T. H., Tao P. D. and Muu L. D., Simplically-constrained DC optimization over the efficient and weakly sets. J. of Optimization Theory and Applications 117, 503 - 531 (2003).

[4] An L. T. H. and Tao P. D., Convex analysis approach to DC programming: Theory, algorithms and applications, ACTA Mathematica Vietnamica, 22, 289 - 355 (1997).

[5] An L. T. H. and Tao P. D., Solving a class of linearly constrained indefinite quadratic problems by DC algorithms, J.of Global Optimization, 11, 253 - 285 (1997).

[6] Benson H. P., Optimization over the efficient set, J. Mathematical Analysis and Applications, 98, 56 - 580 (1984).

[7] Benson H. P., An all-linear programming relaxation algorithm for optimization over the efficient set, J. Global Optimization, 1, 83 - 104 (1991).

[8] Gotoh J., Thoai N. V. and Yamamoto Y., Global optimization method for solving the minimum maximal flow problem, Optimization. Methods Software, 18, 395 - 415 (2003).

[9] Luc L. T. and Muu. L. D., A global optimization approach to optimization over the efficient set, Lecture Notes in Economics and Mathematical Systems, Springer, Berlin, Germany, 452, 183 - 196 (1997).

[10] Muu L .D., Computational aspects of optimization over the efficient set, Vietnam Journal of Mathematics, 23, 85 - 106 (1995).

[11] Kim N. T. B. and Muu L.D., On the projection of the efficient set and potential applications, Optimization 51, 401 - 421 (2002).

[12] Iri M., An essay in the theory of uncontrollable flows and congestion, Technical Report, Department of Information and System Engineering, Faculty of Science and Engineering, Chuo University, TRISE 94 - 03, (1994).

[13] Iri M., Network flow - theory and applications with practical impact, in: J. Dolezal and J. Fidler (eds.) System Modelling and Optimization, Chapman and Hall, London, 24 - 36 (1996).

[14] Philip J., Algorithms for the vector maximization problem, Mathematical Programming, 2, 207 - 229 (1972).

[15] Quy N. V. and Muu L. D., On penalty function method for class of nonconvex constrained optimization problems, Vietnam J. of Mathematics, 29, 235 - 256 (2001).

[16] Tao P. D., Algorithms for solving a class of non convex optimization. Methods of subgradients, Fecmat Days 85, Mathematics for Optimization, Elservier Sience Publishers, B.V.NorthHolland (1986).

[17] Shigeno M., Takahashi I. and Yamamoto Y., Minimum maximal flow problem - an optimization over the efficient set, J. Global Optimization, 25, 425 - 443 (2003).

[18] Shi J.M. and Yamamoto Y., A global optimization method for minimum maximal flow problem, Acta Mathematica Vietnamica, 22, 271 - 287 (1997).

[19] Yamamoto Y. , Optimization over the efficient set: overview, J. of Global Optimization, 22, 285 - 317 (2002).

[20] Zhu D. L. and Marcotte P., An extended descent framework for variational inequalities, J. of Optimization Theory and Applications: 80, 349 - 366 (1994).

[21] Zenke D., Algorithms for the minimum maximal flow problem - Optimization over the efficient set, Graduate School of Systems and Information Engineering, University of Tsukuba March, (2006).