Control and Cybernetics

vol. **43** (2014) No. 1

A new polynomial-time implementation of the out-of-kilter algorithm using Minty's lemma*

by

Mehdi Ghiyasvand

Department of Mathematics, Faculty of Science
Bu-Ali Sina University, Hamedan, Iran
mghiyasvand@basu.ac.ir

Abstract: It is less well known how to use the out-of-kilter idea to solve the min-cost flow problem because the generic version of the out-of-kilter algorithm runs in exponential time, although it is the sort of algorithm that computers can do easily. Ciupala (2005) presented a scaling out-of-kilter algorithm that runs in polynomial time using the shortest path computation in each phase. In this paper, we present a new polynomial time implementation of out-of-kilter idea. The algorithm uses a scaling method that is different from Ciupala's scaling method. Each phase of Ciupala's method needs a shortest path computation, while our algorithm uses Minty's lemma to transform all the out-of-kilter arcs into in-kilter arcs. When the given network is infeasible, Ciupala's algorithm does not work, but our algorithm presents some information that helps to repair the infeasible network.

Keywords: network flows, the minimum cost flow problem, out-of-kilter algorithm, Minty's lemma

1. Introduction

Classical algorithms for the minimum cost flow problem are the out-of-kilter algorithm (see Fulkerson, 1961, and Minty, 1960) and the cheapest path augmentation (see Busaker and Gowen, 1961). The out-of-kilter algorithm uses the complementary slackness optimality condition, it selects arcs that do not satisfy this condition and changes flow and potential to enforce the condition. Other well known ideas to solve the problem have been presented by Edmonds and Karp (1972), Goldberg and Tarjan (1990), Orlin (1993), and Ahuja, Goldberg, Orlin and Tarjan (1992).

The our-of-kilter algorithm runs in exponential time in the worst case (see Ahuja et al., 1993). This algorithm is the sort of algorithm that computers can do easily, but people must be careful in using. Also the out-of-kilter algorithm

*Submitted: November 2012; Accepted: January 2014.

gives a geometrical explanation to the optimality concept. Ciupala (2005) presented a polynomial-time implementation of the out-of-kilter algorithm. This algorithm performs several scaling phases for different values of a parameter Δ. Initially, $\Delta = 2^{\lfloor logU \rfloor}$, where U is the largest absolute arc bound. In each phase, the sum of kilter numbers decreases by at least Δ units. This reduces the value of Δ by the factor of 2 if the residual network contains no arc whose kilter number is at least Δ. In each phase, the shortest path computation is done. Eventually, $\Delta = 1$, and, at the end of this phase, the current flow is a minimum cost flow. Ciupala's (2005) algorithm runs in $O(m\ S(n,m)\ logU)$, where n, m, and $S(n,m)$ denote the number of nodes, number of arcs, and the time required to solve the shortest path problem, respectively.

Although Minty's lemma (see Gondran and Minoux, 1984, and Minty, 1966) looks more like an amusing fact than a deep result, it is actually a rather powerful lemma and plays an important role in the theory of flows in networks. In this paper, we describe a new polynomial-time implementation to the out-of-kilter idea using Minty's lemma. Our method is a scaling algorithm, which uses the scaling idea from Ervolina and McCormick (1993) to have a polynomial implementation. Our algorithm uses Minty's lemma to transform all the out-of-kilter arcs into in-kilter arcs. This algorithm gives a geometrical explanation to the optimality concept. The case when the network is infeasible is diagnosed by the algorithm. We call our algorithm Minty-out-of-kilter algorithm or *MOK algorithm*.

Ciupala's algorithm cannot work for infeasible networks, since an input for Ciupala's algorithm is a feasible flow (see Ciupala, 2005, page 1171, line -3). Thus, it does not present any information for these networks, but MOK algorithm computes a δ^*-min-cost flow for an infeasible network, which gives suitable information to repair the network and estimate the maximum cost for relaxing the lower and upper bounds.

This paper consists of four sections in addition to the Introduction section. Section 2 presents network notation and reviews some results used in the subsequent sections. Our algorithm is shown in Section 3. Section 4 presents two faster implementations of the algorithm. Finally, a comparison of MOK and Ciupala's algorithms is described in Section 5.

2. Preliminaries

2.1. Notation and definitions

A directed graph D is a pair $D = (N, A)$ where N is a set of nodes and A is a set of ordered pairs of nodes, called arcs. We denote an arc from node i to node j by $i \to j$ and define the cost on arc $i \to j$ by c_{ij}. A *simple cycle* C in a directed graph is a sequence $i_1, i_2, \ldots, i_k$ of distinct nodes of N such that either $(i_r, i_{r+1}) \in A$ (a *forward arc* in C) or $(i_{r+1}, i_r) \in A$ (a *backward arc* in C) for $r = 1, 2, \ldots, k$ (where we interpret i_{k+1} as i_1). A *directed cycle* is a simple cycle with all forward arcs. A *simple path* and *directed path* are the same as a simple cycle and directed cycle, respectively, without arc (i_k, i_1). If S is a non-trivial subset of N (i.e. $S \neq \emptyset, S \neq N$) and $\overline{S} = N - S$, then we define $(S, \overline{S}) = \{(i,j) \mid i \in S,\ j \notin S\}$ and $(\overline{S}, S) = \{(i,j) \mid i \notin S,\ j \in S\}$. The arc subset $(S, \overline{S})$ (and $(\overline{S}, S)$) is called a cut.

2.2. The minimum cost flow problem and the optimality of a flow

Let $D = (N, A)$ be a directed graph with $|N| = n$ and $|A| = m$. Let $c \in R^A$ be a cost function on A and l, u be lower and upper bounds on A, with $l_{ij} \leq u_{ij}$ for each arc $(i,j) \in A$. The primal linear programming formulation (*the primal problem*) of the minimum cost flow problem is

$$min \sum_{(i,j)\in A} c_{ij}x_{ij}$$

$$s.t. \sum_{j\in N} x_{ij} - \sum_{j\in N} x_{ji} = 0, \quad i \in N, \tag{1}$$

$$l_{ij} \leq x_{ij} \leq u_{ij}. \qquad (i,j) \in A. \tag{2}$$

The primal problem has flow x_{ij} on arc (i,j). We call $x \in R^A$ a *circulation* if only (1) is required and a *bounded circulation* if both (1) and (2) are required. The primal problem is feasible if and only if there is a flow x satisfying (1) and (2). The following theorem is a well-known result on the feasibility of the primal problem.

Theorem 1 (Hoffman, 1960). *The primal problem is feasible if and only if for every cut* $(S, \overline{S})$, $\sum_{(i,j)\in(S,\overline{S})} l_{ij} - \sum_{(i,j)\in(\overline{S},S)} u_{ij} \leq 0.$

We associate a node potential π_i to each node i and define the reduced cost of an arc (i,j) as

$$c^{\pi}_{ij} = c_{ij} - \pi_i + \pi_j. \tag{3}$$

The dual linear program of the minimum cost flow problem (*the dual problem*) is

$$\begin{array}{lll} \max & \sum_{i\to j\in A} (c^{\pi}_{ij})^{+} l_{ij} - \sum_{i\to j\in A} (c^{\pi}_{ij})^{-} u_{ij}. & \\ \text{s.t} & \pi_j + c_{ij} - \pi_i = c^{\pi}_{ij}, & (i,j) \in A, \\ & \pi_i\text{: free}, & i \in N \end{array}$$

Note that $(c^{\pi})^{+}$=max $(0, c^{\pi})$, $(c^{\pi})^{-}$=max $(0, -c^{\pi})$ and π is feasible for the dual problem if it satisfies the constrains of the dual problem. The complementary slackness conditions for general linear programming, when specialized for the minimum cost flow problem, result in the following characterization of optimal primal and dual solutions for the primal and dual problems.

Theorem 2 *The pair x, π is optimal for the primal and dual problems if and only if*

- *x is feasible for the primal problem,*
- *π is feasible for the dual problem, and*
- *for every $(i,j) \in A$,*

$$c^{\pi}_{ij} < 0 \Rightarrow x_{ij} = u_{ij}, \tag{4}$$

$$c^{\pi}_{ij} > 0 \Rightarrow x_{ij} = l_{ij}. \tag{5}$$

Supposing that we are given node potentials π, for each arc (i,j), define lower bound l^{π}_{ij} and upper bound u^{π}_{ij} by

$$\begin{array}{l} \text{if } c^{\pi}_{ij} > 0, \text{ then } l^{\pi}_{ij} = u^{\pi}_{ij} = l_{ij}, \\ \text{if } c^{\pi}_{ij} = 0, \text{ then } l^{\pi}_{ij} = l_{ij}\ ,\ u^{\pi}_{ij} = u_{ij}, \text{ and} \\ \text{if } c^{\pi}_{ij} < 0, \text{ then } l^{\pi}_{ij} = u^{\pi}_{ij} = u_{ij}. \end{array}$$

The following theorem is an equivalent formulation of (4) and (5).

Theorem 3 (Ervolina and McCormick, 1993; Hassin, 1983). *Feasible node potentials π are optimal to the dual problem if and only if the primal network with nodes N, arcs A, lower bounds l^{π} and upper bounds u^{π} has a bounded circulation flow.*

For $\beta \geq 0$, define that π is β-*optimal* if there is a circulation x such that for all arc (i,j),

$$c^{\pi}_{ij} > 0 \Rightarrow l_{ij} - \beta \leq x_{ij} \leq l_{ij} + \beta, \tag{6}$$

$$c^{\pi}_{ij} = 0 \Rightarrow l_{ij} - \beta \leq x_{ij} \leq u_{ij} + \beta, \text{ and} \tag{7}$$

$$c^{\pi}_{ij} < 0 \Rightarrow u_{ij} - \beta \leq x_{ij} \leq u_{ij} + \beta. \tag{8}$$

By Theorem 3, π is optimal if and only if it is 0-optimal. In our algorithm, we start out with β large and drive δ toward zero. The following lemma says that β need not start out too big, and need not end up too small. We define ψ as the largest absolute arc bound.

LEMMA 1 (Ervolina and McCormick, 1993). *Any node potentials π are ψ-optimal. Moreover, when $\beta < \frac{1}{m}$, all β-optimal node potentials are optimal to the dual problem.*

Consider node potential π, we call a network with nodes N, arcs A and constraints (1), (6), (7) and (8) *the β-network corresponding to π*. We say x is a feasible flow for the β-network corresponding to π if it satisfies (1), (6), (7) and (8). Thus, π is a β-optimal set of node potentials if and only if there is a feasible flow for β-network corresponding to π.

3. The MOK algorithm

Our algorithm treats β as a parameter and iteratively obtains β-optimal potential for successively smaller values of β. Initially, $\beta = \psi$, $x = 0$ and $\pi = 0$. The algorithm executes scaling phases, where each scaling phase cuts β in half and applies *Function-1* that transforms a 2β-optimal set of node potentials into a β-optimal set of node potentials. This continues until $\beta < \frac{1}{m}$, at which point Lemma 1 says that we are finished, having done $O(log(nU))$ phases.

Definition 1 *We are given a feasible flow x for 2β-network corresponding to π, then three sets α, θ, and λ are defined as follows:*
Let $\alpha = \alpha^- \cup \alpha^+$ such that

$$\alpha^+ = \begin{cases} \{(i,j) \in A \mid u_{ij} + \beta < x_{ij} \leq u_{ij} + 2\beta\}, & if \quad c^\pi_{ij} \leq 0, \\ \{(i,j) \in A \mid l_{ij} + \beta < x_{ij} \leq l_{ij} + 2\beta\}, & if \quad c^\pi_{ij} > 0, \end{cases}$$

and

$$\alpha^- = \begin{cases} \{(i,j) \in A \mid u_{ij} < x_{ij} \leq u_{ij} + \beta\}, & if \quad c^\pi_{ij} \leq 0, \\ \{(i,j) \in A \mid l_{ij} < x_{ij} \leq l_{ij} + \beta\}, & if \quad c^\pi_{ij} > 0. \end{cases}$$

Let $\theta = \theta^+ \cup \theta^-$ such that

$$\theta^+ = \begin{cases} \{ (i,j) \in A \mid u_{ij} - 2\beta \leq x_{ij} < u_{ij} - \beta\}, & if \quad c^\pi_{ij} < 0, \\ \{(i,j) \in A \mid l_{ij} - 2\beta \leq x_{ij} < l_{ij} - \beta\}, & if \quad c^\pi_{ij} \geq 0, \end{cases}$$

and

$$\theta^- = \begin{cases} \{(i,j) \in A \mid u_{ij} - \beta \leq x_{ij} < u_{ij}\}, & if \quad c^\pi_{ij} < 0, \\ \{(i,j) \in A \mid l_{ij} - \beta \leq x_{ij} < l_{ij}\}, & if \quad c^\pi_{ij} \geq 0. \end{cases}$$

Let

$$\lambda = \begin{cases} \{(i,j) \in A \mid x_{ij} = u_{ij}\}, & if \quad c^{\pi}_{ij} < 0, \\ \{(i,j) \in A \mid x_{ij} = l_{ij}\}, & if \quad c^{\pi}_{ij} > 0, \\ \{(i,j) \in A \mid l_{ij} \leq x_{ij} \leq u_{ij}\}, & if \quad c^{\pi}_{ij} = 0. \end{cases}$$

If $\theta^+ \cup \alpha^+ = \phi$, then x is also a feasible flow for the β-network corresponding to π. If $\theta^+ \cup \alpha^+ \neq \phi$, then Function-1 changes x_{ij}'s and π_i's using the idea of Minty's lemma. The following lemma is a modified version of Minty's Lemma (see Gondran and Minoux, 1984, and Minty, 1966).

LEMMA 2 *Let $D = (N, A)$ be a graph whose arcs are divided into three sets θ, α, and λ. Assume that $a_o \in \theta$ (respectively, $a_o \in \alpha$), then one of the following cases applies:*
(a) *A simple cycle C containing a_o exists such that all arcs of C that are in the set θ have the same (respectively, opposite) direction as a_o and all arcs of C that are in α have an opposite (respectively same) direction to a_o (call C a* Minty-cycle*).*
(b) *A cut $(S, \overline{S})$, which contains a_o and does not contain the arcs of λ, exists, such that all arcs of $(S, \overline{S})$ are in the set α and all arcs of $(\overline{S}, S)$ are in the set θ, i.e. if $(i,j) \in (S, \overline{S})$ then $(i,j) \in \alpha$ and if $(i,j) \in (\overline{S}, S)$ then $(i,j) \in \theta$ (call $(S, \overline{S})$ a* Minty-cut*).*

Function-1 is the essential part of each phase of the MOK algorithm. The input to Function-1(β, π, x) is a 2β-optimal set of node potentials π and a feasible flow x for the 2β-network corresponding to π and its output is a β-optimal set of node potentials π' and a feasible flow x' for the β-network corresponding to π'.

DEFINITION 2 *Given a directed network $D = (N, A)$, let x be a feasible flow for the 2β-network corresponding to π. We define β-residual network $D(x, \beta) = (N, A(x))$ as follows. For every arc $(i,j) \in A$, there is an arc $(i,j) \in A(x)$ with capacity $r_{ij} = (u_{ij} + \beta) - x_{ij}$ and cost c^{π}_{ij}, and there is an arc (j,i) with capacity $r_{ji} = x_{ij} - (l_{ij} - \beta)$ cost $c^{\pi}_{ji} = -c^{\pi}_{ij}$.*

For each arc $(i,j) \in A(x)$, the *β-kilter number* k_{ij} is defined as follows:

$$k_{ij} = \begin{cases} \lceil \frac{-r_{ji}}{\beta} \rceil, & if \;\; r_{ji} < 0, \\ \lceil \frac{r_{ij} - 2\beta}{\beta} \rceil, & if \;\; r_{ij} > \beta, \text{ and } c^{\pi}_{ij} < 0, \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

LEMMA 3 *Let x be a feasible flow for the 2β-network corresponding to π, then*
a) *$k_{ij} = 0$ or 1, for each $(i,j) \in A(x)$.*
b) *If, for each $(i,j) \in A(x)$, $k_{ij} = 0$, then $\theta^+ \cup \alpha^+ = \phi$.*

PROOF By Definition 2, we have $r_{ji} < 0$ if (i) there is an arc $(i,j) \in A$ such that $(i,j) \in \theta^+$ and $c^\pi_{ij} \geq 0$ (which means $l_{ij} - 2\beta \leq x_{ij} < l_{ij} - \beta$), or (ii) there is an arc $(j,i) \in A$ such that $(j,i) \in \alpha^+$ and $c^\pi_{ji} \leq 0$ (which means $u_{ji} + \beta < x_{ji} \leq u_{ji} + 2\beta$). In case (i), we have $r_{ji} = x_{ij} - (l_{ij} - \beta) < 0$ and $(l_{ij} - \beta) - x_{ij} < \beta$, so $0 < -r_{ji} < \beta$, which means $k_{ij} = \lceil \frac{-r_{ji}}{\beta} \rceil = 1$. In case (ii), we have $r_{ji} = (u_{ij} + \beta) - x_{ij} < 0$ and $0 < x_{ij} - (u_{ij} + \beta) < \beta$, thus $0 < -r_{ji} < \beta$, or $k_{ij} = \lceil \frac{-r_{ji}}{\beta} \rceil = 1$.

In a similar way, by Definition 2, we have $r_{ij} > \beta$, and $c^\pi_{ij} < 0$ if (1) there is an arc $(i,j) \in A$ such that $(i,j) \in \theta^+$ and $c^\pi_{ij} < 0$ (which means $u_{ij} - 2\beta \leq x_{ij} < u_{ij} - \beta$), or (2) there is an arc $(j,i) \in A$ such that $(j,i) \in \alpha^+$ and $c^\pi_{ji} > 0$ (which means $l_{ji} + \beta < x_{ji} \leq l_{ji} + 2\beta$). In case (1), we have $2\beta < r_{ij} = (u_{ij} + \beta) - x_{ij} \leq 3\beta$ or $0 < r_{ij} - 2\beta \leq \beta$, which means $k_{ij} = \lceil \frac{r_{ij} - 2\beta}{\beta} \rceil = 1$. Also, in case (2), we have $2\beta < r_{ij} = x_{ij} - (l_{ij} - \beta) \leq 3\beta$ or $0 < r_{ij} - 2\beta \leq \beta$, so, $k_{ij} = \lceil \frac{r_{ij} - 2\beta}{\beta} \rceil = 1$.
Thus, by (9), for each $(i,j) \in A(x)$, we have $k_{ij} = 0$ or 1. Also, if $(i,j) \in \theta^+ \cup \alpha^+$, then, in $A(x)$, we have $k_{ij} = 1$ or $k_{ji} = 1$ (and vice versa). Therefore, if $\theta^+ \cup \alpha^+ = \phi$, then, for each $(i,j) \in A(x)$, $k_{ij} = 0$ (and vice versa). □

Now, *the β-active network* is defined as follows. The β-active network consists of all arcs in the β-residual network, but
i) If $r_{ij} < \beta$, then delete (i,j).
ii) If $r_{ij} > \beta$, $r_{ji} > \beta$, and $c^\pi_{ij} \neq 0$, then, between (i,j) and (j,i), delete the one with a positive reduced cost.

LEMMA 4
a) *If $(i,j) \in \theta$, then (i,j) is in the β-active network, but (j,i) is not.*
b) *If $(i,j) \in \alpha$, then (j,i) is in the β-active network, but (i,j) is not.*
c) *If $(i,j) \in \lambda$, then both (i,j) and (j,i) are in the β-active network.*

PROOF Consider $(i,j) \in \theta$. If $c^\pi_{ij} \geq 0$, then $r_{ji} < \beta$ (note, if $(i,j) \in \theta^+$, then $r_{ji} < 0$), so (j,i) is not in the β-active network. If $c^\pi_{ij} < 0$, then $r_{ij} > \beta$ and $r_{ji} > \beta$. Thus, by the definition of the β-active network, (j,i) is deleted (since $c^\pi_{ji} > 0$).

Now, consider $(i,j) \in \alpha$. If $c^\pi_{ij} \leq 0$, then $r_{ij} < \beta$ (note, if $(i,j) \in G^+$, then $r_{ij} < 0$), so (i,j) is not in the β-active network. If $c^\pi_{ij} > 0$, then $r_{ij} > \beta$ and $r_{ji} > \beta$. Thus, by the definition of the β-active network, (i,j) is deleted (since $c^\pi_{ij} > 0$).

If $(i,j) \in \lambda$ then, in $A(x)$, $r_{ji} \geq \beta$ and $r_{ij} \geq \beta$. Also, if $(i,j) \in \lambda$ and $c^\pi_{ij} \neq 0$, then, in $A(x)$, we have $r_{ij} = \beta$ (if $c^\pi_{ij} < 0$) or $r_{ji} = \beta$ ($c^\pi_{ij} > 0$). Thus, both (i,j) and (j,i) are in the β-active network. □

Let x be a feasible flow for the 2β-network corresponding to π. Using Lemmas 3 and 4, we describe Function-1 (see Algorithm 1). Construct the β-active network. If $k_{ij} = 0$ (for each (i, j) in the β-active network), then, by Lemma 3, we have $\theta^+ \cup \alpha^+ = \phi$, which means x is a feasible flow for the β-network corresponding to π. Otherwise, an arc (w, v) with $k_{wv} = 1$ is chosen. Let S be the set of nodes that are reachable in the β-active network from node v. If $w \in S$, then find the cycle C containing (w, v) and send β units of flows around C (by Lemmas 2 and 4, C is a Minty-cycle). This will reduce k_{wv} to 0, and will not increase the β-kilter of any arc. By Lemma 4, we get the following lemma.

LEMMA 5 *Suppose that there is an arc (w, v) with $k_{wv} = 1$ and S is the set of nodes that are reachable in the β-active network from node v, then*
a) *In $D = (N, A)$, if $(i, j) \in (S, \overline{S})$, then $(i, j) \in \alpha$.*
b) *In $D = (N, A)$, if $(i, j) \in (\overline{S}, S)$, then $(i, j) \in \theta$.*

Note that, by Lemmas 2 and 5, $(S, \overline{S})$ is a Minty-cut. If $i \notin S$, Function-1 changes the node potentials π so that we can reach at least a node of the set of nodes $\overline{S}$. For it, the procedure sends at least one $(i, j) \in (S, \overline{S}) \cup (\overline{S}, S)$ into the set λ. Let $Q_1 = \{c^{\pi}_{ij} \mid (i, j) \in (S, \overline{S}),\ c^{\pi}_{ij} > 0,\ x_{ij<u_{ij}}\ \}$, $Q_2 = \{-c^{\pi}_{ij} \mid (i, j) \in (\overline{S}, S),\ c^{\pi}_{ij} < 0,\ x_{ij} > l_{ij}\ \}$, and $\eta = \min\ (\min Q_1, \min Q_2)$.

It is easy to prove that if $Q_1 \cup Q_2 = \emptyset$, then the primal problem is infeasible. Supposing that $Q_1 \cup Q_2 \neq \phi$, we change the π_i's according to

$$\pi'_i = \begin{cases} \pi_i + \eta, & \text{if} \quad i \in S, \\ \pi_i, & \text{if} \quad i \in \overline{S}. \end{cases} \tag{10}$$

LEMMA 6 *After adjusting π_i's,*
(a) At least one arc in $(S, \overline{S}) \cup (S, \overline{S})$ is entered into the set λ.
(b) The arcs of the set $\theta^- \cup \alpha^- \cup \lambda$ are not entered into the set $\theta^+ \cup \alpha^+$.

PROOF By (10), for each $(i, j) \in (S, \overline{S})$, c^{π}_{ij} is decreased by η and for each $(i, j) \in (\overline{S}, S)$, c^{π}_{ij} is increased by η. Lemma 5 says that if $(i, j) \in (S, \overline{S})$, then $(i, j) \in \alpha$ and if $(i, j) \in (\overline{S}, S)$, then $(i, j) \in B$. Let $Q_1 \cup Q_2 \neq \phi$, so, at least one $c^{\pi}_{ij} \neq 0$ (such that $c^{\pi}_{ij} \in Q_1$ or $-c^{\pi}_{ij} \in Q_2$) is changed to $c^{\pi}_{ij} = 0$. Therefore, at least an arc $(i, j) \in (S, \overline{S}) \cup (\overline{S}, S)$ is entered into the set λ. Hence, we proved (a).

Now, we prove (b). c^{π}_{ij}'s are changed only if $(i, j) \in (S, \overline{S}) \cup (\overline{S}, S)$. Suppose that $(i, j) \in (S, \overline{S})$, so $i \to j \in \alpha$, and c^{π}_{ij} is decreased. We show that if $(i, j) \in \alpha^-$ then it is not entered into $\theta^+ \cup \alpha^+$. There are two cases. 1) $c^{\pi}_{ij} \leq 0$: by decreasing c^{π}_{ij}, the arc (i, j) is retained in the set α^-. 2) $c^{\pi}_{ij} > 0$: it is impossible that c^{π}_{ij} be changed to $c^{\pi}_{ij} < 0$. Thus, after decreasing c^{π}_{ij} , (i, j) will be in α^- or λ. The case $(i, j) \in (\overline{S}, S)$ can be proved in a similar way to the case $(i, j) \in (S, \overline{S})$. □

Therefore, Function-1, in each iteration, selects an arc (w, v) with $k_{wv} = 1$ and finds the set of nodes that are reachable in the β-active network from node v . If $w \in S$, then there is a cycle C containing (w, v) (which is a Minty-cycle). But if $w \notin S$, then we can add at least one new node to the set S (which is a Minty-cut). Algorithm 1 shows Function-1 and the following theorem gives its running time.

Function-1
Begin
Define the β-residual network $D(x, \beta)$ and the β-active network;
1 Do while there is an arc (w, v) with $k_{wv} = 1$;
Begin
2 Let S be the set of nodes that are reachable in the β-active network from node v;
If $w \in S$
Begin
Find, in the β-active network, the cycle C containing (w, v);
Send β units of flow around C;
Update the β-residual network using Definition 2;
Update k_{ij}'s and the β-active network;
Go to 1;
End
If $Q_1 \cup Q_2 = \phi$ then the primal problem is infeasible and break;
Else
Begin
Update π_i's (by (10)), the β-active network and k_{ij}'s;
If $c^{\pi}_{wv} = 0$ then update k_{ij}'s and go to 1;
Else go to 2;
End;
End;
End.

Algorithm 1. Function-1.

THEOREM 4 *The MOK algorithm runs in $O(m^2 n\ log(nU))$ time.*

PROOF By Lemma 6, once an arc (w, v) with $k_{wv} = 1$ changes to $k_{wv} = 0$, it will never change to $k_{wv} = 1$ in the other iterations, therefore the number of iterations, in Function-1, is at most m. In each iteration, in the worst case, we need at most $n - 1$ computations of operations, which run in $O(m)$ time, so, Function-1 runs in $O(m^2 n)$. Therefore, by Lemma 1, we conclude the running time of MOK algorithm. □

The MOK algorithm gives a geometrical explanation to the optimality concept of the minimum cost flow problem. In each phase, it shows how we are away from the optimality conditions by β units.

4. Faster implementations of MOK algorithm

In this section, faster implementations of the MOK algorithm are presented. Let $r_{ij} = u_{ij} - x_{ij}$ and $r_{ji} = x_{ij} - l_{ij}$. The kilter number k_{ij} of arc (i, j) is defined as the increase needed in the flow in (i, j) to bring the arc in to kilter. Thus, 1) if $c^{\pi}_{ij} < 0$ and $r_{ij} \geq 0$, then $k_{ij} = r_{ij}$; 2) If $c^{\pi}_{ij} \leq 0$ and $r_{ij} < 0$, then $k_{ij} = -r_{ji}$.

If the kilter number of (i, j) is at least 1, then sending one unit of flow in (i, j) in the residual network will decrease the kilter number of (i, j) by 1. In the original out-of-kilter algorithm, the kilter number is the deviation from the optimality conditions. In our definition, increasing the flow will decrease the kilter numbers (note that at most one of the arcs (i, j) and (j, i) has a positive kilter number).

Each phase starts with a flow such that kilter number of every arc is at most β. At the end of the scaling phase, each arc has kilter number at most β. We construct a kilter graph at the scaling phase as follows. An arc (i, j) is green (respectively yellow) if $k_{ij} > \beta$ (respectively yellow). If $k_{ij} = k_{ji} = 0$, then (i, j) and (j, i) are both yellow. If (i, j) is yellow, and if β units of flow are sent in (i, j), the kilter numbers of (i, j) drops to zero, and the kilter number of (j, i) is at most β. We refer to the graph as the β-kilter graphs. The following theorem is proven in a similar way to the proof of Minty's lemma.

Theorem 5 *Suppose that (i, j) is a green arc of the β-kilter graph. Either there is a directed cycle C containing (i, j) such that C contains green and yellow arcs or else there is a partition of the nodes into subsets S and T such that $j \in S$, $i \in T$, and there is no green or yellow arc directed from S to T.*

Theorem 1 is much simpler than Minty's colored lemma, but it is all that is needed for the out-of-kilter algorithm. If the MOK algorithm is run using the above ideas, it can be implemented in $O(n^2m)$ time, which would be an improvement over the algorithm presented in the last section by a factor of m/n. The m/n factor improvement is obtained by continuing with the same green arc until a cycle is found (or the problem is proved to be infeasible) and by letting $f(j)$ be minimum reduced cost of an arc from set S to node j, where set S is the set of nodes reachable from node j in the β-kilter graph. The reduced costs would not be stored exactly following a change in node potential. Rather, the $f(j)$ values would be updated. This update takes $O(n)$ time instead of $O(m)$.

Let $S(n, m)$ be the complexity of the shortest path, the running time could be further improved to $O(mS(n, m))$ per scaling scaling phase by waiting till the cycle is found before updating any of the node potentials or reduced costs. But this implementation would look identical to the shortest path implementation of the out-of-kilter. This improved implementation would be much more efficient than the algorithm presented in the last section, but it would determine the same cycles, and it would have the same node potentials following the changes in flow as does the algorithm in the last section.

5. Comparison of MOK algorithm and Ciupala's algorithm

In this section, merits of our algorithm in comparison with Ciupala's algorithm are discussed. For this purpose, we need the following definitions.

DEFINITION 3 *A flow x is defined as a δ-min-cost flow w.r.t. potential π if*

1.
$$\sum_{j\in N} x_{ij} - \sum_{j\in N} x_{ji} = 0, \quad i \in N.$$

2. For each arc $(i,j) \in A$:

$$\begin{aligned}
&\text{If } c^{\pi}_{ij} > 0 \Rightarrow l_{ij} - \delta \le x_{ij} \le l_{ij} + \delta,\\
&\text{If } c^{\pi}_{ij} = 0 \Rightarrow l_{ij} - \delta \le x_{ij} \le u_{ij} + \delta, \text{ and}\\
&\text{If } c^{\pi}_{ij} < 0 \Rightarrow u_{ij} - \delta \le x_{ij} \le u_{ij} + \delta.
\end{aligned}$$

DEFINITION 4 *A flow x is called a δ^*-min-cost flow if it is a δ^*-min-cost flow, but not a $\frac{\delta^*}{2}$-min-cost flow.*

THEOREM 6 *Suppose that network $G(V,A)$ is infeasible and l_{ij} and u_{ij} are the lower and upper bounds for $(i,j) \in A$. Let x^* be a δ^*-min-cost flow w.r.t. potential π, then*
a) Network $G = (V,A)$ with bounds $\overline{l}_{ij}$ and $\overline{u}_{ij}$ defined as follows is feasible:

$$\begin{aligned}
&\text{If } c^{\pi}_{ij} > 0 \text{ and } x^*_{ij} < l_{ij}, \text{ then } \overline{l}_{ij} = x^*_{ij} \text{ and } \overline{u}_{ij} = u_{ij}.\\
&\text{If } c^{\pi}_{ij} > 0 \text{ and } x^*_{ij} \ge l_{ij}, \text{ then } \overline{l}_{ij} = l_{ij} \text{ and } \overline{u}_{ij} = u_{ij}.\\
&\text{If } c^{\pi}_{ij} < 0 \text{ and } x^*_{ij} > u_{ij}, \text{ then } \overline{u}_{ij} = x^*_{ij} \text{ and } \overline{l}_{ij} = l_{ij}.\\
&\text{If } c^{\pi}_{ij} < 0 \text{ and } x^*_{ij} \le u_{ij}, \text{ then } \overline{u}_{ij} = u_{ij} \text{ and } \overline{l}_{ij} = l_{ij}.\\
&\text{If } c^{\pi}_{ij} = 0 \text{ and } x^*_{ij} < l_{ij}, \text{ then } \overline{l}_{ij} = x^*_{ij} \text{ and } \overline{u}_{ij} = u_{ij}.\\
&\text{If } c^{\pi}_{ij} = 0 \text{ and } x^*_{ij} > u_{ij}, \text{ then } \overline{u}_{ij} = x^*_{ij} \text{ and } \overline{l}_{ij} = l_{ij}.\\
&\text{If } c^{\pi}_{ij} = 0 \text{ and } l_{ij} \le x^*_{ij} \le u_{ij}, \text{ then } \overline{l}_{ij} = l_{ij} \text{ and } \overline{u}_{ij} = u_{ij}.
\end{aligned}$$

b) The upper bound for the cost of repairing the infeasible network is $m\delta^*$.

PROOF By (5) and Definition 4, we conclude Claim (a). By Definition 5, for each $(i,j) \in A$, we have $\overline{l}_{ij} - l_{ij} \le \delta^*$ and $\overline{u}_{ij} - u_{ij} \le \delta^*$. On the other hand, there are m arcs, so Claim (b) is obvious. □

It is obvious that each min cost flow is a 0-min-cost flow. There are many infeasible networks, so there is no 0-min-cost flow for these networks. For example, consider the network in Fig. 1. This network is infeasible, since the lower bound for exiting flow from Node 2 is 11 units, but the upper bound for its entering flow is 8. An input for Ciupala's algorithm is a feasible flow (see Ciupala, 2005, page 1171, line -3), so, Ciupala's algorithm can not work for infeasible networks and does not present any result for these networks, but

MOK algorithm presents a δ^*-min-cost flow. By Theorem 6, this flow is an infeasible flow which gives suitable information to estimate the maximum cost for repairing the infeasible network. In the following, by MOK algorithm, first a δ^*-min-cost flow for the infeasible network in Fig. 1 is computed, then an estimate of the maximum cost for repairing this infeasible network is presented.

Initial data in MOK algorithm are $\beta = U = 12$, $\pi = 0$ and $x = 0$. It is obvious that $x = 0$ is 12-optimal. By letting $\beta = 12/2 = 6$, we apply the first phase of MOK algorithm using Function-1(6,0,0). By $r_{ij} = (u_{ij} + \beta) - x_{ij}$ and $r_{ji} = x_{ij} - (l_{ij} - \beta)$, we have

$r_{12} = (8+6) - 0 = 14$ and $r_{21} = 0 - (4-6) = 2$, so $k_{12} = k_{21} = 0$,
$r_{23} = (8+6) - 0 = 14$ and $r_{32} = 0 - (4-6) = 2$, so $k_{23} = k_{32} = 0$,
$r_{24} = (10+6) - 0 = 16$ and $r_{42} = 0 - (7-6) = -1$, so $k_{24} = 1$ and $k_{42} = 0$,
$r_{43} = (12+6) - 0 = 18$ and $r_{34} = 0 - (6-6) = 0$, so $k_{43} = k_{34} = 0$,
$r_{31} = (11+6) - 0 = 17$ and $r_{13} = 0 - (4-6) = 2$, so $k_{31} = k_{13} = 0$.

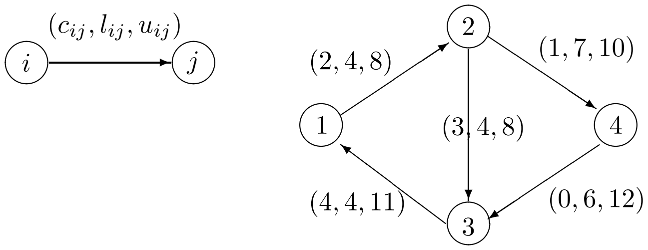

Figure 1. An example network

The values of r_{13}, r_{34}, r_{42}, r_{32} and r_{21} are less than $\beta = 6$, thus arcs $(1,3)$, $(3,4)$, $(4,2)$, $(3,2)$ and $(2,1)$ are deleted and the 6-active network is as in Fig. 2.

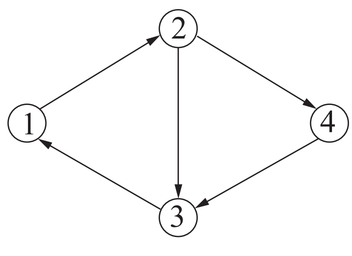

Figure 2. The 6-active network

Note that $(w, v) = (2, 4)$. The set of nodes that are reachable in the 6-active network from node 4 is $S = \{3, 1, 2\}$, hence $w = 2 \in S$. By sending $\beta = 6$ units of flow around cycle $4 - 3 - 1 - 2 - 4$, we have:

$x_{43} = x_{31} = x_{12} = x_{24} = 6$ and $x_{23} = 0$,
$r_{12} = (8+6) - 6 = 8$ and $r_{21} = 6 - (4-6) = 8$, so $k_{12} = k_{21} = 0$,
$r_{23} = (8+6) - 0 = 14$ and $r_{32} = 0 - (4-6) = 2$, so $k_{23} = k_{32} = 0$,

$r_{24} = (10+6) - 6 = 10$ and $r_{42} = 6 - (7-6) = 5$, so $k_{24} = k_{42} = 0$,
$r_{43} = (12+6) - 6 = 12$ and $r_{34} = 6 - (6-6) = 6$, so $k_{43} = k_{34} = 0$,
$r_{31} = (11+6) - 6 = 11$ and $r_{13} = 6 - (4-6) = 8$, so $k_{31} = k_{13} = 0$.

There is no arc (w, v) with $k_{wv} = 1$, so a new phase with $\beta = 6/2 = 3$ is started.

In this phase, we have:

$x_{43} = x_{31} = x_{12} = x_{24} = 6$ and $x_{23} = 0$,
$r_{12} = (8+3) - 6 = 5$ and $r_{21} = 6 - (4-3) = 5$, so $k_{12} = k_{21} = 0$,
$r_{23} = (8+3) - 0 = 11$ and $r_{32} = 0 - (4-3) = -1$, so $k_{23} = 1$ and $k_{32} = 0$,
$r_{24} = (10+3) - 6 = 7$ and $r_{42} = 6 - (7-3) = 2$, so $k_{24} = k_{42} = 0$,
$r_{43} = (12+3) - 6 = 9$ and $r_{34} = 6 - (6-3) = 3$, so $k_{43} = k_{34} = 0$,
$r_{31} = (11+3) - 6 = 8$ and $r_{13} = 6 - (4-3) = 5$, so $k_{31} = k_{13} = 0$.

By $(r_{42} < \beta)$, $(r_{12} > \beta$, $r_{21} > \beta$, and $c^{\pi}_{12} > 0)$ and $(r_{31} > \beta$, $r_{13} > \beta$, and $c^{\pi}_{31} > 0)$, the arcs $(4, 2)$, $(1, 2)$, and $(3, 1)$ are deleted and the 3-active network is as in Fig. 3. In this stage $(w, v) = (2, 3)$ and the set of nodes that are reachable

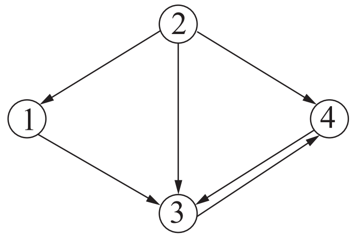

Figure 3. The 3-active network

in the 3-active network from node 3 is $S = \{3, 4\}$, so $w \notin S$, but $Q_1 = \{4\}$ and $Q_2 = \phi$, which means $\eta = 4$. Thus, by (7),

$$\pi_4 = \pi_3 = 4, \pi_1 = \pi_2 = 0, c^{\pi}_{31} = 0, c^{\pi}_{24} = 5, c^{\pi}_{23} = 7, c^{\pi}_{12} = 2, c^{\pi}_{43} = 0.$$

By $c^{\pi}_{31} = 0$, arc $(3, 1)$ is added to the 3-network, so $S = \{3, 4, 1\}$, but $w \notin S$. In this stage, $Q_1 = \{2\}$ and $Q_2 = \phi$, so $\eta = 2$, and

$$\pi_4 = \pi_3 = 6, \pi_1 = 2, \pi_2 = 0, c^{\pi}_{31} = 0, c^{\pi}_{24} = 7, c^{\pi}_{23} = 9, c^{\pi}_{12} = 0, c^{\pi}_{43} = 0.$$

Consequently, $S = \{3, 4, 1, 2\}$ and $w \in S$. By sending $\beta = 3$ units of flow around the cycle $2 - 3 - 1 - 2$, we have:

$$x_{43} = x_{24} = 6,\ x_{31} = x_{12} = 9 \text{ and } x_{23} = 3.$$

Hence,

$$\begin{array}{l} c^{\pi}_{12}=0,\ r_{12}=(8+3)-9=2 \text{ and } r_{21}=9-(4-3)=8, \text{ so } k_{12}=k_{21}=0,\\ c^{\pi}_{23}=9,\ r_{23}=(8+3)-3=8 \text{ and } r_{32}=3-(4-3)=2, \text{ so } k_{23}=k_{32}=0,\\ c^{\pi}_{24}=7,\ r_{24}=(10+3)-6=7 \text{ and } r_{42}=6-(7-3)=2, \text{ so } k_{24}=k_{42}=0,\\ c^{\pi}_{43}=0,\ r_{43}=(12+3)-6=9 \text{ and } r_{34}=6-(6-3)=3, \text{ so } k_{43}=k_{34}=0,\\ c^{\pi}_{31}=0,\ r_{31}=(11+3)-9=5 \text{ and } r_{13}=9-(4-3)=8, \text{ so } k_{31}=k_{13}=0. \end{array}$$

In this stage, for each arc (i,j): $k_{ij}=0$, thus a new phase with $\beta=3/2=1.5$ is started. In the new phase, r_{ij}'s and k_{ij}'s are as follows:

$c^{\pi}_{12}=0$, $r_{12}=(8+1.5)-9=0.5$ and $r_{21}=9-(4-1.5)=6.5$, so $k_{12}=k_{21}=0$,
$c^{\pi}_{23}=9$, $r_{23}=(8+1.5)-3=6.5$ and $r_{32}=3-(4-1.5)=0.5$, so $k_{23}=k_{32}=0$,
$c^{\pi}_{24}=7$, $r_{24}=(10+1.5)-6=5.5$ and $r_{42}=6-(7-1.5)=0.5$, so $k_{24}=k_{42}=0$,
$c^{\pi}_{43}=0$, $r_{43}=(12+1.5)-6=7.5$ and $r_{34}=6-(6-1.5)=1.5$, so $k_{43}=k_{34}=0$,
$c^{\pi}_{31}=0$, $r_{31}=(11+1.5)-9=3.5$ and $r_{13}=9-(4-1.5)=6.5$, so $k_{31}=k_{13}=0$.

Thus, for each arc (i,j), $k_{ij}=0$, and the algorithm goes to the next phase with $\beta=1.5/2=0.75$ and the following values:

$$\begin{array}{l} c^{\pi}_{12}=0,\ r_{12}=-0.25 \text{ and } r_{21}=5.75, \text{ so } k_{12}=0, k_{21}=1.\\ c^{\pi}_{23}=9,\ r_{23}=5.75 \text{ and } r_{32}=-0.25, \text{ so } k_{23}=1, k_{32}=0.\\ c^{\pi}_{24}=7,\ r_{24}=4.75 \text{ and } r_{42}=-0.25, \text{ so } k_{24}=1, k_{42}=0.\\ c^{\pi}_{43}=0,\ r_{43}=6.75 \text{ and } r_{34}=0.75, \text{ so } k_{43}=k_{34}=0.\\ c^{\pi}_{31}=0,\ r_{31}=2.75 \text{ and } r_{13}=5.75, \text{ so } k_{31}=k_{13}=0. \end{array}$$

The values of r_{12}, r_{32}, and r_{42} are less than $\beta=0.75$, so arcs $(1,2)$, $(3,2)$, and $(4,2)$ are deleted and the 0.75-active network is as in Fig. 4.

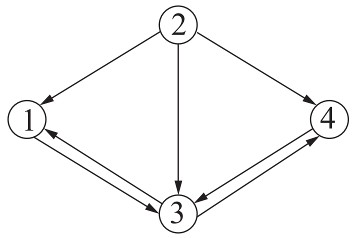

Figure 4. The 0.75-active network

Note that $(w,v)=(2,1)$. The set of nodes that are reachable in the 0.75-active network from node 1 is $S=\{1,3,4\}$, hence $w=1\notin S$, but $Q_1\cup Q_2=\phi$, which means the network presented in Fig 1 is infeasible.

Therefore, $\delta^* = 1.5$ and a δ^*-min-cost flow is $x_{43} = x_{24} = 6$, $x_{31} = x_{12} = 9$ and $x_{23} = 3$. By Theorem 6, in order to have a feasible network, the lower and upper bounds is changed as follows:

$$\begin{array}{l} c^{\pi}_{24} > 0 \text{ and } x^*_{24} = 6 < l_{24} = 7, \text{ so } \bar{l}_{24} = 6 \text{ and } \overline{u}_{24} = u_{24}, \\ c^{\pi}_{43} = 0 \text{ and } l_{43} = 6 = x^*_{43} < u_{43}, \text{ so } \bar{l}_{43} = l_{43} \text{ and } \overline{u}_{43} = u_{43}, \\ c^{\pi}_{23} > 0 \text{ and } x^*_{23} = 3 < l_{23} = 4, \text{ so } \bar{l}_{23} = 3 \text{ and } \overline{u}_{23} = u_{23}, \\ c^{\pi}_{31} = 0 \text{ and } l_{31} < x^*_{31} < u_{31}, \text{ so } \bar{l}_{31} = l_{31} \text{ and } \overline{u}_{31} = u_{31}, \\ c^{\pi}_{12} = 0 \text{ and } x^*_{12} = 9 > u_{12} = 8, \text{ so } \overline{u}_{12} = 9 \text{ and } \bar{l}_{12} = l_{12}. \end{array}$$

Thus, the lower bound in arcs $(2, 4)$ and $(2, 3)$ is decreased by one unit. Also, the upper bound in arc $(1, 2)$ is increased by one unit. Hence, by MOK algorithm, in order to have a feasible network, the sum of relaxations in bounds is 3 units, but Ciupala algorithm does not present any information to repair the infeasible network.

Acknowledgements

I would like to thank the referee no. 1 for his/her suggestions to use residual networks instead of using the original network, which greatly simplify the exposition. The author is also grateful to the referee for his/her valuable comments to present Section 4, which improves the running time of our algorithm to $O(mS(n, m))^*$. I would like to express my very great appreciation to Jan W. Owsinski, Executive Editor, for his valuable and constructive comments, especially for suggesting Section 5, which discusses the novelty and importance of our algorithm.

References

AHUJA R.K., GOLDBERG A.V., ORLIN J.B. and TARJAN R.E. (1992) Finding minimum-cost flows by double scaling. *Mathematical Programming* **53**, 243-266.

AHUJA R.K., MAGNANTI T.L. and ORLIN J.B. (1993) *Network Flows: Theory, Algorithms, and Applications*. Prentice-Hall, Englewood Cliffs, NJ.

BUSAKER R.G. and GOWEN P.J. (1961) A procedure for Determining a Family of minimal-cost Network Flow patterns. Technical Report **15**, O.R.O.

CIUPALA L. (2005) A scaling out-of-kilter algorithm for minimum cost flow. *Control and Cybernetics* **34**(4), 1169-1174.

EDMONDS I. and KARP R.M. (1972) Theoretical improvements in algorithmic efficiency for network flow problems. *Journal of the Association on Computing Machinery* **19**, 248-264.

*Actually, the idea of improvement in Section 4 came from the referee no. 1 (Eds.).

ERVOLINA T.R. and MCCORMICK S.T. (1993) Two strongly polynomial cut canceling algorithms for minimum cost network flow. *Discrete Applied Mathematics* **46**, 133-5.

FULKERSON D.R. (1961) An out-of-kilter method for minimal cost flow problems. *SIAM Journal on Applied Mathematics* **9**, 18-27.

GOLDBERG A.V and TARJAN R.E. (1990) Finding minimum-cost circulations by successive approximation. *Mathematics of Operations Research* **16**, 430-466.

GONDRAN M. and MINOUX M. (1984) *Graphs and Algorithms* (trans. S. Vajda). Wiley, New York.

HASSIN R. (1983) The minimum cost flow problem: A unifying approach to dual algorithms and a new tree-search algorithm. *Mathematical Programming* **25**, 228-239.

HOFFMAN A.J. (1960) Some recent applications of the theory of linear inequalities to extremal combinatorial analysis. In: R. Bellman and M. Hall (eds.), *Combinatorial Analysis. Proc. of Symposia in Applied Mathematics*, **X**. American Mathematical Society, Providence, Rhode Island, 113-127.

JEWELL W.S. (1958) Optimal Flow through Networks. Technical Report **8**, M.I.T.

MINTY G.J. (1960) Monotone networks. *Proc. Roy. Soc. London*, **257**, 194-212.

MINTY G.J. (1966) On the Axiomatic Foundations of the Theories of Directed linear Graphs, Electrical Networks and Programming. *Journal of Mathematics and Mechanics* **15**, 485-520.

ORLIN J.B. (1993) A faster strongly polynomial minimum cost flow algorithm. *Operations Research*, **41**, 338-350.