Journal of Global Optimization **6:** 135–151, 1995.

The Minimum Concave Cost Network Flow Problem with Fixed Numbers of Sources and Nonlinear Arc Costs

HOANG TUY,* SAIED GHANNADAN, ATHANASIOS MIGDALAS and PETER VÄRBRAND
Department of Mathematics, Institute of Technology, Linköping University, 581 83, Linköping, Sweden

(Received: 30 August 1994)

Abstract. We prove that the Minimum Concave Cost Network Flow Problem with fixed numbers of sources and nonlinear arc costs can be solved by an algorithm requiring a number of elementary operations and a number of evaluations of the nonlinear cost functions which are both bounded by polynomials in r, n, m, where r is the number of nodes, n is the number of arcs and m the number of sinks in the network.

Key words: Minimum concave cost network flow, fixed number of sources and nonlinear arc costs, strongly polynomial algorithm, parametric method.

1. Introduction

One of the most challenging problems of network optimization is the Minimum Concave-Cost Network Flow Problem (MCCNFP) which can be formulated as follows.

Let $G = (N_G, A_G)$ be a directed graph with node set $N_G = \{N_1, \ldots, N_r\}$ and arc set $A_G = \{a_1, \ldots, a_n\}$, where each arc a_i is an ordered pair of distinct nodes. Associated with each arc a_i are a *capacity* $q_i \in [0, +\infty]$ and a *concave cost function* $g_i(t) : R_+ \to R_+$. Associated with each node N_j is a *demand* d_j such that $\sum_{j=1}^{r} d_j = 0$. Nodes N_j with $d_j < 0$ are the *sources*, nodes N_j with $d_j > 0$ are the *sinks*. If d_j is the demand of a source N_j then $s_j = -d_j$ is also called its *supply*. A *flow* in G is any vector $x \in R_+^n$ such that $0 \leqslant x_i \leqslant q_i, i = 1, \ldots, n$. The component x_i is the *value of the flow* on the arc a_i. For each j, we denote the set of arcs entering (leaving, resp.) node N_j by $N_j^+ (N_j^-$, resp.) and define

$$\alpha_j(x) = \sum_{i:a_i \in N_j^+} x_i - \sum_{i:a_i \in N_j^-} x_i.$$

* On leave from Institute of Mathematics, P.O. Box 631, Bo Ho, Hanoi, Vietnam.

A flow x is *feasible* (more precisely, feasible to the demand vector $d = (d_1, \dots, d_r)$) if $\alpha_j(x) = d_j$ for every j. The *cost* of the flow x is then the value

$$g(x) = \sum_{i=1}^{n} g_i(x_i).$$

The problem is

MCCNFP: Given the demand vector d, find a feasible flow in G with minimal cost, i.e.:

$$\begin{aligned} &\text{minimize} && g(x) && \\ &\text{s.t.} && \alpha_j(x) = d_j \quad j = 1, \dots, r && (1)\\ & && 0 \leqslant x_i \leqslant q_i \quad i = 1, \dots, n. && (2) \end{aligned}$$

At the expense of introducing additional sources if necessary, one can always reduce the problem to an equivalent uncapacitated one, i.e. to a problem where no q_i is finite (see e.g. [3]). Therefore, throughout the sequel, without loss of generality, we may assume that $q_i = +\infty$ for all arcs a_i. Under these conditions, if there is just one single source, the problem will be referred to as the single source uncapacitated minimum concave-cost network flow problem (SSU MCCNFP).

In view of its relevance to numerous applications in operations research, economics, engineering, etc. MCCNFP has been the subject of intensive research (e.g. [3], [4], [6], [15], [19], [20], [27], [30]). For a discussion on the applications and a recent review of the literature on this problem, we refer the reader to the articles of Guisewite and Pardalos [8] and [9].

MCCNFP is a *linearly constrained concave minimization* problem. It is well known that even special cases of it, such as the fixed charge network flow problem or SSU MCCNFP, are NP-hard (see, e.g. [16], [8]). No wonder that despite the efforts of many researchers, all algorithms so far developed for general MCCNFP run in exponential time and can only handle problem instances of very small size. This has motivated the consideration of additional structures which might make the problem more tractable when present in MCCNFP. In fact, highly efficient polynomial time algorithms have been developed for a number of specially structured variants of MCCNFP ([2], [11], [17], [27], [28], [30], [31]).

The difficulty of a nonconvex global optimization problem critically depends on the number of nonlinear elements (for example the number of variables that enter the objective function or the constraints in a nonlinear way). When this number is fixed certain nonconvex problems become polynomially solvable. Such is the case, for instance, of mixed integer linear programming problems with a fixed number of integer variables, according to a well known result by Lenstra [14]. For MCCNFP the elements on which the difficulty of the problem critically depends are the number of arcs with nonlinear costs and also the number of sources (a set of h sources can be replaced by one single source along with h arcs with bounded capacity). It is therefore convenient to refer to MCCNFP with exactly h sources and k nonlinear arc costs as MCCNFP $(h; k)$ or $FP(h; k)$ for short. The mentioned result of Lenstra and also the result of Tardos [18] on the strong polynomial solvability of linear transportation problems suggest the conjecture that $FP(h; k)$ for fixed h, k should be strongly polynomially solvable, too.

In fact, this has been proved for $FP(1; 1)$ ([13], [23]); see also [10], where a first polynomial algorithm for $FP(1; 1)$ was given), $FP(1; 2)$ and $FP(2; 1)$ [25]. The aim of the present paper is to prove this conjecture for the general case, i.e. for arbitrary fixed natural numbers h and k.

As in our previous paper [25], the method used in the sequel is based on a polynomial reduction of $FP(h; k)$ to a production-transportation problem with linear transportation cost and nonlinear production cost. The latter problem can then be solved by a strongly polynomial algorithm presented in [26] and based on the parametric approach earlier developed for so called rank k quasiconcave minimization problems ([21], [22]). The resulting algorithm for $FP(h; k)$ has a running time polynomial in the number r of nodes, the number n of arcs and the number m of sinks and exponential only in h and k. In this connection, it should be noted that the send-and-split method of Ericksson *et al.* [3] is polynomial in r and n but exponential in m.

The paper is organized as follows. In Section 2 we first reduce the dimension of the problem by constructing a network equivalent to the original one but generally with much less nodes and arcs. Next, in Section 3, we show that $FP(h; k)$ can be reduced to solving a certain production-transportation problem with linear transportation costs and nonlinear production cost. These transformations require a polynomial number of operations. In Section 4 we consider the case when $\min\{h, k\} = 1$. It turns out that in this case the equivalent production-transportation problem has a concave production cost and hence can be solved by a strongly polynomial algorithm presented in our previous paper [26]. Section 5 discusses the general case which requires some modification of the technique developed in the cited paper. Finally Section 6 closes the paper with some concluding remarks.

2. The Reduced Network

As defined in the Introduction, $FP(h; k)$ is the *uncapacitated* MCCNFP with h *sources*, and k *nonlinear arc costs*, i.e. the MCCNFP on a network G such that:

$$d_j < 0 \text{ for } j = 1, \dots, h \quad (N_1, \dots, N_h \text{ are sources});$$
$$d_j > 0 \text{ for } j = h + 1, \dots, h + m \quad (N_{h+1}, \dots, N_{h+m} \text{ are sinks});$$
$$g_i(t) : R_+ \to R_+ \text{ is concave nonlinear for } i = 1, \dots, k;$$
$$g_i(t) = c_i t, \quad c_i \geqslant 0, \quad \text{for all} \quad i > k;$$
$$d_j = 0 \text{ for all } j > h + m; \quad q_i = +\infty \text{ for all } i.$$

For the sake of simplicity we will further assume that there is at least one feasible flow in G (i.e., the system (1)(2) has at least one solution) and that

$$g_i(0) = 0 \quad \text{and} \quad g_i(t) \text{ is nondecreasing on } [0, +\infty). \tag{3}$$

(Note, however, that $g_i(t)$ may be discontinuous at point $t = 0$, as in the case of fixed charge). As is known (see e.g. [3]), under these conditions the problem will always have a finite optimal solution which is an extreme flow (a flow corresponding to a spanning forest).

For convenience the arcs $a_1, \ldots a_k$ with nonlinear arc costs are called *black*; the other arcs are called *white*, and the unit cost $c_i \geqslant 0$ associated with a white arc a_i is its *length*. By splitting certain nodes into two or several nodes connected by white arcs of length zero if necessary, it may be arranged that the initial and terminal nodes of the black arcs are all distinct and none of them is a source or a sink. For reasons which will soon be apparent, we then rename the nodes of G as follows:

- initial node (tail) of black arc a_i : $W_i (i = 1, \ldots, k)$;
- terminal node (head) of black arc a_i : $F_i (i = 1, \ldots, k)$;
- sources $F_{k+1}, \ldots, F_{k+h}$ (so $F_{k+i} = N_i, i = 1, \ldots, h$);
- sinks: $W_{k+1}, \ldots, W_{k+m}$(so $W_{k+j} = N_{h+j}, j = 1, \ldots, m$).

Also we set

$$s_i = -d_i (i = 1, \ldots, h); \quad b_j = d_{h+j} (j = 1, \ldots, m), \quad s = \sum_{i=1}^{h} s_i. \tag{4}$$

Thus G is a network with sources $F_{k+1}, \ldots, F_{k+h}$, sinks $W_{k+1}, \ldots, W_{k+m}$, and black arcs $a_i = (W_i, F_i), i = 1, \ldots, k$, where the supply of a source F_{k+1} is $s_i > 0, i = 1, \ldots, h$, the demand of a sink W_{k+j} is $b_j > 0, j = 1, \ldots, m$, the cost associated with a black arc a_i is a nonnegative valued concave function $g_i(t)$ satisfying (3), while the length of a white arc a_i is $c_i \geqslant 0$. The problem we are concerned with is

$FP(h; k)$: Find a feasible flow x in G with smallest cost

$$\sum_{i=1}^{k} g_i(x_i) + \sum_{i=k+1}^{n} c_i x_i.$$

Note that the network G may contain many other nodes than $F_1, \ldots, F_{k+h}$, $W_1, \ldots, W_{k+m}$. It turns out, however, that in solving $FP(h; k)$ we can replace G by a reduced network G^*, equivalent to it, but having $F_1, \ldots, F_{k+h}, W_1, \ldots, W_{k+m}$ as the only nodes. This reduced network can be constructed as in Figure 1.

Let us call a path π in G a *white path* if it does not include any black arc; the length of a white path π is then $c(\pi) = \sum\{c_i : a_i \in \pi\}$ and a white path π is said to be *shortest* if its length is smallest among all white paths with same origin and same end. Now, observe that if we know the values $\bar{x}_i = u_i, i = 1, \ldots, k$ of an optimal

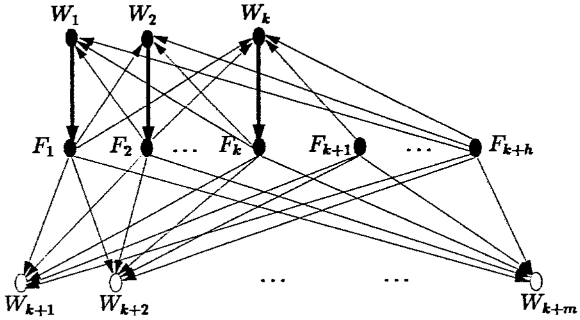

Fig. 1. Network G^*.

flow $\bar{x}$ on the black arcs, then the values of $\bar{x}$ on the white arcs can be determined by solving a linear transportation problem. Indeed, let H^u denote the network obtained from the original network G by removing all the black arcs and replacing the zero demands of the nodes $W_i, F_i, i = 1, \ldots, k$ with $u_i, -u_i$ respectively (so if $u_i > 0$ then W_i becomes a sink and F_i a source in H^u). Clearly the restriction $\bar{x}^u$ of $\bar{x}$ to the set of white arcs of G is an optimal flow on H^u, i.e. an optimal solution of the *minimum cost flow problem on the network H^u with only white arcs*. But it is immediate that the latter problem is equivalent to the *linear transportation problem* on a network with supply points $F_1, \ldots, F_k, F_{k+1}, \ldots, F_{k+h}$, destination points $W_1, \ldots, W_k, W_{k+1}, \ldots, W_{k+m}$, supply quantities $u_1, \ldots, u_k, s_1, \ldots, s_h$, demand quantities $u_1, \ldots, u_k, b_1, \ldots, b_m$, and with cost matrix $[l_{ij}]$, where $l_{ii} = +\infty$ for $i = 1, \ldots, k$, and l_{ij} for $i \neq j$ or $i > k$ is the length of the shortest white path in G from F_i to W_j (if no such white path exists, set $l_{ij} = +\infty$). Now let G^* be the network that results from G by removing all white arcs and all nodes other than $F_1, \ldots, F_{k+h}, W_1, \ldots, W_{k+m}$, and introducing, for each pair (i, j) a white arc (F_i, W_j), with length l_{ij} as just defined (Figure 1).

PROPOSITION 1. *The network G^* is equivalent to the original network G, in the sense that every optimal flow in G corresponds to an optimal flow in G^* with equal cost and conversely.*

Proof. If $x \in R^n_+$ is an optimal flow in G and $x_i = u_i$ for $i = 1, \ldots, k$, then, as said above, the restriction of x to the white arcs of G is an optimal flow in H^u, and so determines an optimal solution z of the corresponding linear transportation problem. Therefore, $\tilde{z} = (u, z)$ is an optimal flow in G^* with same cost as x. Conversely, if $\tilde{z} = (u, z)$ with $u = (u_1, \ldots, u_k) \in R^k_+, z = [z_{ij}], i = 1, \ldots, k + h, j = 1, \ldots, k + m$, is an optimal flow in G^* then an optimal flow $x \in R^n_+$ in G (with equal cost) is given by

$$x_i = u_i \quad (i = 1, \ldots, k) \tag{5}$$

$$x_t = \sum \{z_{ij} : a_t \in \Gamma_{ij}\} \quad (t = k + 1, \ldots, n) \tag{6}$$

where Γ_{ij} denotes the shortest white path in G from F_i to W_j. □

Remark. Let $\eta > \sum_{i=1}^{k}(\gamma_i + g_i^+(0)) + \sum_* c_{ij}$, where $\gamma_i = \lim_{t\to 0^+} g_i(t)$, $g_i^+(0)$ denotes the right derivative of $g_i(t)$ at $t = 0$ and $\sum_*$ means that the sum is extended to all white arcs (F_i, W_j) in G^* such that $c_{ij} < +\infty$. It is easily seen that the optimal flow in G^* does not change if every infinite length of a white arc is replaced by η. Indeed, let (u, z) be a feasible flow in G^* such that $z_{i_0 j_0} > 0$ on some white arc (F_{i_0}, W_{j_0}) of infinite length. If (u', z') denotes a feasible flow in G^* such that $z'_{ij} = 0$ on every white arc (F_i, W_j) of infinite length (such a flow exists by the feasibility of the problem), then one can find a cycle beginning at (F_{i_o}, W_{j_0}), such that $z_{ij} - z'_{ij} > 0$ on every odd arc of this cycle, while $z_{ij} - z'_{ij} < 0$ on every even arc. Upon subtracting a suitable value $\alpha > 0$ from z_{ij} on every odd arc and adding α to z_{ij} on every even arc, the cost of (u, z) will be reduced by at least $\alpha(\eta - \sum_{i=1}^{k}(\gamma_i + g_i^+(0)) - \sum_* c_{ij}) > 0$. This implies that (u, z) cannot be optimal. So the optimal flows will not change when every infinite length of a white arc is replaced by η, and therefore, in the sequel, without loss of generality we can assume that every white arc in G^* has a finite length.

3. The Equivalent Production-Transportation Problem

Let $\tilde{z} = (u, z)$ with $u = (u_1, \dots, u_k)$, $z = [z_{ij}]$, $i = 1, \dots, k+h$, $j = 1, \dots, k+m$, be a feasible flow in the reduced network G^*. For every $i = 1, \dots, k+h$, define y_i to be the total amount of the flow $\tilde{z}$ going from F_i to all the sinks $W_{k+1}, \dots, W_{k+m}$, i.e.

$$y_i = \sum_{j=k+1}^{k+m} z_{ij} \quad i = 1, \dots, k+h. \tag{7}$$

Clearly, $y \in \Omega$, where

$$\Omega = \left\{ y \in R_+^{k+h} : \sum_{i=1}^{k+h} y_i = s \right\}. \tag{8}$$

Now for a given vector $y \in \Omega$ let us partition the network G^* into two subnetworks (Figure 2):

(1) an upper subnetwork G_U^*, with h sources $F_{k+1}, \dots, F_{k+h}$ of supplies $s_1 - y_{k+1}, \dots, s_h - y_{k+h}$, k sinks $F_1, \dots, F_k$ of demands $y_1, \dots, y_k$ and k intermediate nodes (nodes with null demand) $W_1, \dots, W_k$;

(2) a lower subnetwork G_L^* with $k+h$ sources $F_1, \dots, F_{k+h}$ of supplies $y_1, \dots, y_{k+h}$ and m sinks $W_{k+1}, \dots, W_{k+m}$ of demands $b_1, \dots, b_m$.

The upper subnetwork G_U^* has just h sources and k black arcs as the original network G while the lower subnetwork G_L^* has only white arcs.

Denote by $f(y)$ the cost of an optimal flow in G_U^* and for any feasible flow $\tilde{z} = (u, z)$ in G_L^* satisfying (7) let $x_{ij} = z_{i(k+j)}$, $i = 1, \dots, k+h$, $j = 1, \dots, m$. Also set $c_{ij} = l_{i(k+j)}$.

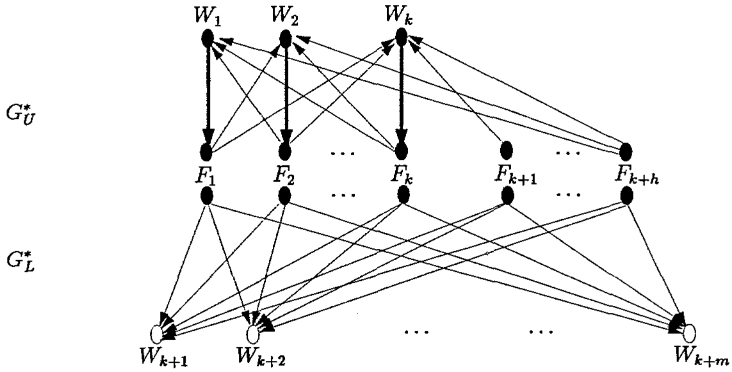

Fig. 2. Lower and upper subnetworks of G^*.

PROPOSITION 2. *$FP(h;k)$ is equivalent to the problem*

$$(Q_{hk}) \quad \text{minimize } f(y) + \sum_{i=1}^{k+h} \sum_{j=1}^{m} c_{ij} x_{ij} \tag{9}$$

$$\text{s.t.} \quad \sum_{j=1}^{m} x_{ij} = y_i \quad i = 1, \ldots, k+h \tag{10}$$

$$\sum_{i=1}^{k+h} x_{ij} = b_j \quad j = 1, \ldots, m \tag{11}$$

$$x_{ij} \geqslant 0 \quad \forall i, j \tag{12}$$

$$y_{k+i} \leqslant s_i \quad i = 1, \ldots, h, y \in \Omega. \tag{13}$$

Proof. As we saw above, $FP(h;k)$ is equivalent to finding an optimal flow in the network G^*. If $\tilde{z} = (u, z)$ is an optimal flow in G^* and y is defined by (7) then $y \in \Omega$ and the part of the flow $\tilde{z}$ in G_U^* has cost $f(y)$, while the part in G_L^* is an optimal solution of the linear transportation problem

$$TP(y): \quad \text{minimize} \sum_{i=1}^{k+h} \sum_{j=1}^{m} c_{ij} x_{ij} \quad \text{s.t.} \quad (10)(11)(12).$$

Hence the conclusion. □

If we regard $F_1, \ldots, F_{k+h}$ as representing factories and $W_{k+1}, \ldots, W_{k+m}$ as representing warehouses, then (Q_{hk}) can be interpreted as the following *Production-Transportation Problem* $(PTP(h+k))$:

Given the demands $b_1, \ldots, b_m$ of the warehouses, the joint cost $f(y)$ of producing y_i units of goods at factory $F_i, i = 1, \ldots, k+h$ and the transportation cost matrix $c = [c_{ij}]$, determine the production levels $y_1, \ldots, y_{k+h}$ to be assigned to the factories together with the transportation pattern $[x_{ij}]$ so as to meet all the demands with the cheapest total production-transportation cost.

PROPOSITION 3. *The transformation of $FP(h;k)$ to (Q_{hk}) requires $O(r \log_2 r + n)$ elementary operations.*

Proof. This transformation involves solving $(h+k)$ single-source multiple-sinks $(k+m$ sinks) shortest path problems in a network with n arcs, r nodes. Therefore, using Fredman and Tarjan's implementation of Dijikstra's algorithm [5], it can be completed in time $O(r \log_2 r + n)$. When an optimal solution (y, x) of (Q_{hk}) has been found then an optimal flow (u, z) on G^* can be obtained such that $z_{i(k+j)} = x_{ij}$ for $i = 1, \ldots, k+h, j = 1, \ldots, m$ while $u, z_{ij}, i = 1, \ldots, k+h, j = 1, \ldots, k$ are given by an optimal flow in G^*_U. Since the latter network involves exactly $h + 2k$ nodes (h sources, k sinks, k intermediate nodes) and $k(h+k)$ arcs, it has at most $\binom{k(h+k)}{h+2k-1}$ spanning trees (extreme feasible flows), so the search for an optimal flow in this network (which can be carried out at least by enumerating all spanning trees) requires a bounded time. Finally, from an optimal flow (u, z) in G^* one can derive an optimal flow $x \in R^n$ in G by formulas (5) and (6), which requires $O(n)$ elementary operations. □

4. Algorithm for the Case min$\{h, k\} = 1$

When $\min\{h, k\} = 1$, i.e. there is only one source or only one nonlinear arc cost, we can prove the following

PROPOSITION 4. *The function $f(y)$ is concave on Ω.*

Proof. Note that there is in G^* a path from every source to every sink. If $h = 1$ then G^*_U is a single source network and it is well known that an extreme optimal flow on G^*_U exists which is a spanning tree T such that for each $i = 1, \ldots, k, T$ contains a (unique) path from the (unique) source F_{k+1} to the sink F_i (see e.g. [6]). Let $Sp(G^*_U)$ be the set of all such spanning trees of G^*_U. Clearly every $T \in Sp(G^*_U)$ determines a flow x^T in G^*_U such that if T_i denotes the path in T from the source F_{k+1} to F_i then the value of x^T on every arc a of G^*_U is

$$x^T(a) = \sum \{y_i : \text{arc } a \text{ belongs to } T_i\}, \tag{14}$$

which is an affine function of y on Ω. Therefore, the cost $f_T(y)$ of x^T is a concave function of y on Ω. Since

$$f(y) = \min\{f_T(y) : T \in Sp(G^*_U)\}, \tag{15}$$

the concavity of $f(y)$ on Ω follows.

If $k = 1$ then the cost $f(y)$ of the part of the flow in G^*_U is

$$f(y) = g_1(y_1) + \sum_{i=2}^{h+1} l_{i1}(s_{i-1} - y_i)$$

and the concavity of $f(y)$ is obvious. □

From Proposition 4 it follows that when $\min\{h, k\} = 1$, (Q_{hk}) is a Production-Transportation Problem of type $PTP(h + k)$ studied in our previous paper [26]. Therefore, $FP(h, k)$ with $\min\{h, k\} = 1$ can be solved by the following

ALGORITHM 1.
(1) Transform $FP(h, k)$ into (Q_{hk}).
(2) Solve (Q_{hk}) by the algorithm for $PTP(h + k)$ given in [26].

THEOREM 1. *For fixed h, k, such that $\min\{h, k\} = 1$ Algorithm 1 solves $FP(h; k)$, requiring at most $O(r \log_2 r + n) + P(m)$ elementary operations and $Q(m)$ evaluations of the functions $g_i(t), i = 1, \ldots, k$, where n is the number of arcs, r the number of nodes in G, m the number of sinks, and $P(m), Q(m)$ are polynomials in m.*

Proof. As proved in [26] Step 2) requires at most $P(m)$ elementary operations and $Q(m)$ evaluations of $g_i(t), i = 1, \ldots, k$. The conclusion then follows from Proposition 3. □

Remarks. (i) For $h = 1$, if (y, k) is an optimal solution of (Q_{1k}) then an optimal flow $\tilde{z} = (u, z)$ in G^* is given by

$$u_i = \sum\{y_\nu : (W_i, F_i) \in T_\nu\}, \quad \nu = 1, \ldots, k, \tag{16}$$

$$z_{ij} = \begin{cases} \sum\{y_\nu : (F_i, W_j) \in T_\nu\} & i = 1, \ldots, k+1, \quad j = 1, \ldots, k \\ x_{i(j-k)} & i = 1, \ldots, k+1, \quad j = k+1, \ldots, k+m, \end{cases} \tag{17}$$

where T_ν is the unique path in T from the source F_{k+1} to $F_\nu, \nu = 1, \ldots, k$, and T is the spanning tree corresponding to an optimal extreme flow in the network G^*_U (for the given y).

(ii) For $k = 1$, if (y, x) is an optimal solution of (Q_{h1}) then an optimal flow $\tilde{z} = (u, z)$ in G^* is given by

$$u = y_1, \quad z_{i1} = s_{i-1} - \sum_{j=1}^{m} x_{ij} \quad (i = 2, \ldots, h+1) \tag{18}$$

$$z_{ij} = x_{i(j-1)} \quad (i = 1, \ldots, h+1; j = 2, \ldots, 1+m). \tag{19}$$

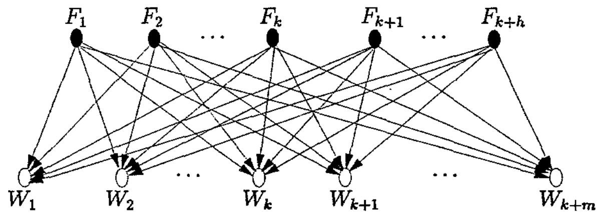

Fig. 3. Bipartite graph of $TP^*(u)$.

5. Algorithm for General $FP(h; k)$

In the general case, when $\min\{h, k\} > 1$ the function $f(y)$ is no longer concave.[1] Although the previous method could be modified to overcome this difficulty, it would involve solving a transportation problem depending on a $(k+h)$-dimensional parameter $y = (y_1, \ldots, y_{k+h})$, where y_i is the sum of all amounts of flow in G^* going from F_i to all the $W_j, j = k+1, \ldots, k+m$. A more convenient method is to use, instead, the parameter $u = (u_1, \ldots, u_k)$, where u_i is the amount of flow passing through the black arc a_i.

Clearly, $u \in \Lambda$, where

$$\Lambda = \{u \in R^k : 0 \leqslant u_i \leqslant s, \quad i = 1, \ldots, k\}, \tag{20}$$

and $s = \sum_{i=1}^{h} s_i$, see (4). Setting $g(u) = \sum_{i=1}^{k} g_i(u_i)$, it is nearly obvious that:

PROPOSITION 5. *$FP(h; k)$ is equivalent to the problem*

$$(Q^*_{hk}) \quad \textit{minimize} \quad g(u) + \sum_{i=1}^{k+h} \sum_{j=1}^{k+m} l_{ij} z_{ij} \tag{21}$$

$$\textit{s.t.} \qquad \sum_{j=1}^{k+m} z_{ij} = \begin{cases} u_i & i = 1, \ldots, k \\ s_{i-k} & i = k+1, \ldots, k+h \end{cases} \tag{22}$$

$$\sum_{i=1}^{k+h} z_{ij} = \begin{cases} u_j & j = 1, \ldots, k \\ b_{j-k} & j = k+1, \ldots, k+m \end{cases} \tag{23}$$

$$z_{ij} \geqslant 0 \quad \forall i, j \tag{24}$$

$$u \in \Lambda. \tag{25}$$

Proof. (Q^*_{hk}) is nothing but the optimal flow problem on the network G^* which by Proposition 1 is equivalent to the optimal flow problem on G. □

For $u \in \Lambda$ denote by $\psi(u)$ the optimal value of the parametric transportation problem

$$TP^*(u): \quad \text{minimize} \sum_{i=1}^{k+h} \sum_{j=1}^{k+m} l_{ij} z_{ij} \quad \text{s.t.} \quad (22)(23)(24).$$

As is well known, $\psi(u)$ is a convex piecewise affine function. Let $\mathbf{P}^*$ be the collection of all linearity pieces (cells) of $\psi(u)$ and for each cell $\Pi \in \mathbf{P}^*$ let V_Π be its vertex set.

PROPOSITION 6. *If*

$$u^* \in \mathit{argmin}\{g(u) + \psi(u): \quad u \in V_\Pi, \quad \Pi \in \mathbf{P}^*\} \tag{26}$$

and z^* *is an optimal solution of* $TP^*(u^*)$ *then* (u^*, z^*) *is an optimal solution of* (Q^*_{hk}).

Proof. For $u \in \Pi$, the function $\psi(u)$ is affine, so $g(u) + \psi(u)$ is concave and hence, attains its minimum over Π at a point in V_Π. Therefore, u^* defined by (26), is a minimizer of $g(u) + \psi(u)$ over the whole Λ and (u^*, z^*) is an optimal solution of (Q^*_{hk}). □

We are thus led to the following

ALGORITHM 2.
(1) Transform $FP(h;k)$ into (Q^*_{hk}).
(2) Generate the collection $\mathbf{P}^*$ of all cells of $\psi(u)$ and
for each cell Π compute its vertex set V_Π. Then compute
u^* satisfying (26) and an optimal solution z^* of $TP^*(u^*)$.
(3) From (u^*, z^*) deduce an optimal solution of $FP(h;k)$.

THEOREM 2. *Algorithm 2 solves* $FP(h;k)$, *requiring at most* $O(r \log_2 r + n) + \tilde{P}(m)$ *elementary operations and* $\tilde{Q}(m)$ *evaluations of the functions* $g_i(t), i = 1, \ldots, k$, *where* $\tilde{P}(m)$ *and* $\tilde{Q}(m)$ *are polynomials in* m.

For the proof of this theorem, we first recall some results from [26].

Consider the bipartite graph H of the linear transportation problem $TP^*(u)$, as depicted in Fig. 3. To simplify the language, we will say arc (i, j) to mean arc (F_i, W_j). In this graph, if μ is an elementary chain beginning at one of the nodes $F_1, \ldots, F_{k+h}$ then, dividing its arc set into two groups of alternating odd and even arcs such that the first arc is odd, we can define

$$l(\mu) = \sum_{(i,j)\ \text{odd}} l_{ij} - \sum_{(i,j)\ \text{even}} l_{ij}. \tag{27}$$

Assume that:

(Nondegeneracy assumption) Every elementary chain μ in H has $l(\mu) \neq 0$.

The following properties have been established in [26]:

(i) A spanning tree T of H is said to be *dual feasible* if there exist real numbers $\{v_i, w_j\}, i = 1, \ldots, k+h; j = 1, \ldots, k+m$ satisfying

$$v_1 = 0, \quad w_j - v_i = l_{ij} \quad (i,j) \in T, \tag{28}$$

$$w_j - v_i < l_{ij} \quad (i,j) \notin T \tag{29}$$

A tree L in H is called a *shoot* if all the nodes $F_1, \ldots, F_{k+h}$ belong to L, while every node $W_j, j = 1, \ldots, k+m$ either does not belong to L or is incident to at least two arcs of L; the set $N = \{j : W_j \text{ belongs to } L\}$, which is then called the *base* of the shoot, always satisfies $|N| \leqslant k+h-1$. A shoot L is said to be *proper* if there exist real numbers $\{v_i, w_j\}$ such that

$$v_1 = 0, \quad w_j - v_i = l_{ij} \quad (i,j) \in L, \tag{30}$$

$$w_j - v_i < l_{ij} \quad (i,j) \notin L, j \in N. \tag{31}$$

Every dual feasible spanning tree T contains a unique proper shoot L which is obtained by removing from T all the nodes W_j that are leaf nodes together with the arcs incident to these. Conversely, every proper shoot L can be uniquely extended to a dual feasible spanning tree T by defining

$$w_j = \min\{v_q + l_{qj} : q = 1, \ldots, k+h\} \quad j \notin N \tag{32}$$

$$J_i = \{j : v_i + l_{ij} < v_q + l_{qj} \quad \forall q \neq i\} \quad i = 1, \ldots, k+h \tag{33}$$

$$T = L \cup \{(i,j) : j \in J_i, \quad i = 1, \ldots, k+h\}. \tag{34}$$

(note that $J_i \subset \{1, \ldots, k+m\} \backslash N, J_i \cap J_q = \emptyset \quad \forall i \neq q$ and $J_1 \cup \ldots \cup J_{k+h} \cup N = \{1, \ldots, k+m\}$)

(ii) If L is a proper shoot of base N and T the associated dual feasible spanning tree, then there exists a uniquely defined basic solution $x^T = [\xi_{ij}(u)]$ of system (22)(23) such that

$$\xi_{ij}(u) = 0 \quad \forall (i,j) \notin T \tag{35}$$

Thus for every $(i,j) \in T, \xi_{ij}(u)$ is an affine function of u and

$$\xi_{ij}(u) = \begin{cases} u_j & j \in J_i \cap \{1, \ldots, k\} \\ b_{j-k} & j \in J_i \cap \{k+1, \ldots, k+m\} \end{cases} \tag{36}$$

while the polytope

$$\Pi = \{u \in \Lambda : \xi_{ij}(u) \geqslant 0 \quad \forall (i,j) \in L\} \tag{37}$$

is a cell of $\psi(u)$. Conversely, for any cell Π of $\psi(u)$ there exists exactly one proper shoot L such that Π is defined by (37).

Also note that in (37) the inequalities $\xi_{ij}(u) \geqslant 0 \quad \forall (i,j) \in L$ imply $0 \leqslant u_i \leqslant s, i = 1, \ldots, k$, so actually

$$\Pi = \{u \in R^k : \xi_{ij}(u) \geqslant 0 \quad \forall (i,j) \in L\}. \tag{38}$$

(iii) The vector $(v, w) = \{v_i, i = 1, \ldots, k + h; w_j, j \in N\}$ associated with a proper shoot L of base N is a basic solution of the system

$$v_1 = 0, \quad w_j - v_i \leqslant l_{ij}, \quad i = 1, \ldots, k + h; \quad j \in N. \tag{39}$$

Conversely, any basic solution (v, w) of this system determines a proper shoot $L = \{(i, j) : w_j - v_i = l_{ij}, i = 1, \ldots, k + h; j \in N\}$ whose base is contained in N.

Thus, the collection of cells of $\psi(u)$ can be found by computing the basic solutions of systems of the form (39), with $N \subset \{1, \ldots, k+m\}$, $|N| \leqslant k+h-1$.

(iv) Since $|L| = 2(k + h - 1)$, by (38) each cell Π is a polytope defined by a system of $2(k + h - 1)$ linear inequalities in R^k, hence has a bounded number of vertices which can be computed in bounded time.

It follows from the above that Step 2 of Algorithm 2, i.e. the computation of the point u^* that achieves the minimum of $g(u) + \psi(u)$ among the set of all vertices of all cells $\Pi \in \mathbf{P}^*$ can be carried out according to the following

Main Procedure (under nondegeneracy assumption).

For each set $N \subset \{1, \ldots, k + m\}$ such that $|N| \leqslant k + h - 1$ **do:**

1. Compute the vertex set of the polytope (39) and retain only the vertices (v, w) such that, for every $j \in N$ there exist at least two indices i satisfying $w_j - v_i = l_{ij}$.
2. For each vertex (v, w) thus obtained **do:**

   2.1 Form the proper shoot
   $$L = \{(i, j) : w_j - v_i = l_{ij}, i = 1, \ldots, k + h; j \in N\}$$
   and using formulas (32)(33)(34) extend L to a dual feasible spanning tree T.

   2.2 Compute $x^T = [\xi_{ij}(u)]$ and define the cell
   $$\Pi = \{u \in R^k : \xi_{ij}(u) \geqslant 0 \; \forall (i, j) \in L\}$$

   2.3 Compute the vertex set of Π. Whenever a vertex u is obtained and u^* has not yet been defined, or $g(u) + \psi(u) < g(u^*) + \psi(u^*)$ then reset $u^* = u$.

   End do

End do

PROPOSITION 7. *The Main Procedure requires at most $\tilde{P}(m)$ elementary operations and $\tilde{Q}(m)$ evaluations of the functions $g_i(t), i = 1, \ldots, k$, where $\tilde{P}(m)$ and $\tilde{Q}(m)$ are polynomials in m.*

Proof. Since h, k are fixed, for each set N the system (39) has a bounded number of basic solutions which can be computed in bounded time. So for each set N, step 1 of the Main Procedure requires a bounded time and generates a bounded number of points (v, w). Then for each point (v, w) step 2 of the Main Procedure requires a bounded number of elementary operations and a bounded number of evaluations of the functions $g_i(t), i = 1, \ldots, k$. Therefore, every set N is processed in bounded time with a bounded number of evaluations of the functions $g_i(t), i = 1, \ldots, k$. The conclusion is then immediate because there are in all $\sum_{q=1}^{k+h-1} \binom{k+m}{q}$ distinct sets $N \subset \{1, , \ldots, k+m\}$ of size at most $k+h-1$ and this number is polynomial in m. □

Remarks. (i) By a result of Balinski and Wallace (see [29]), the total number of cells is $\binom{m+2k+h-2}{k+h-1}$. Also note that since $l_{ii} = \eta$ for $i = 1, \ldots, k$ (see Remark to Proposition 1) if $u \in V_{\Pi}$ is such that $\xi_{ii}(u) > 0$ for some $i = 1, \ldots, k$, then u cannot be optimal for (26), so in (28) it suffices to consider only those u satisfying $\xi_{ii} = 0, i = 1, \ldots, k$.

(ii) As shown in [26], the nondegeneracy assumption can always be made to hold by replacing each l_{ij} with $l_{ij} + j\varepsilon^i$, where $\varepsilon > 0$ is arbitrarily small. Since (28) implies that $v_i = l(\mu_i), w_j = l(\nu_j)$, where $\mu_i(\nu_j$, resp.) is the chain in T from F_1 to $F_i(W_j$, resp.), v_i, w_j will become polynomials in ε. Then, in all the above formulas, an inequality like (29) for all arbitrarily small $\varepsilon > 0$ should mean that the vector of coefficients of the polynomial (ordered by increasing powers of ε)

$$l(\nu_j) - l(\mu_i) - (l_{ij} + j\varepsilon^i)$$

is lexicographically inferior to 0.

Note that all the polynomials in ε that can appear have at most degree $k + h$, so all the vectors that have to be compared when using the ε-perturbation have dimension at most $k + h + 1$. For fixed h, k this adds only a bounded number of elementary operations for each comparison considered. Therefore, in any case, all the cells can be computed in polynomial time. We are now in a position to complete the proof of Theorem 2.

Proof of Theorem 2. By Proposition 7 and the above discussion Step 2 of Algorithm 2 requires, in any case, at most $\tilde{P}(m)$ elementary operations and $\tilde{Q}(m)$ evaluations of the functions $g_i(t), i = 1, \ldots, k$. On the other hand, by Proposition 3, Steps 1 and 3 require $O(r \log_2 r + n)$ elementary operations. □

6. Conclusion

In this paper, we have proved that MCCNFP with a fixed number of sources and nonlinear arc costs is strongly polynomially solvable. The proposed algorithms are practical for small values of h, k, and especially efficient for $h + k \leqslant 3$ (see

[23], [25]). For larger values of h, k they quickly loose practicability, which can be seen from the fact that their time bound is exponential in h, k, as should be expected from the NP-hardness of MCCNFP. There are, however, at least two ways to alleviate the difficulty when h, k are too large. First, by the above approach $FP(h; k)$ is reduced to a problem of minimizing a certain nonlinear function over a discrete set, namely the collection of cells of the optimal cost function of a parametric linear transportation problem. Since a natural concept of neighbouring elements can be defined in this discrete set (two cells being neighbouring if the corresponding vertices of the dual transportation polytope are adjacent), some heuristic or stochastic procedures, such as tabu search or simulated annealing, could be incorporated in the basic algorithms to enhance their practicability. In a subsequent paper we will show how this can be done and report computational experience with the resulting hybrid algorithm. Second, using the reduced network, $FP(h; k)$ which originally involves n variables is converted into problem (26) with only k variables. Since in many cases k is much smaller than n, problem (26) can be practically solved by an adaptation of recently developed global optimization algorithms [21]. We refer the interested reader to the paper [12], where such an approach has been applied to a cheapest flow problem in a network involving simultaneously arcs with concave and arcs with convex costs.

Acknowledgements

This paper was completed during the visit of Hoang Tuy to Linköping Institute of Technology (1991-93), with the support of Marianne and Marcus Wallenberg's Foundation.

The authors should like to thank Dr. P.T. Thach for pointing out that the function $f(y)$ needs not be concave when $\min\{h, k\} > 1$.

Note

[1] In the proof of Proposition 4 the formula (14) is no longer true when $\min\{h, k\} > 1$. To convince the reader consider the special case when all costs in G^*_U are linear. Then $f(y)$ is the cost of an optimal flow, i.e. the optimal value of a linear program of the form

$$\min cx \text{ s.t. } Ax = y, \quad x \geqslant 0,$$

hence as is well known must be a convex function of y. It is easy to construct an example where the optimal value of the above program is convex but not affine, hence not concave.

References

1. Bellman, R.E. (1958), On a routing problem, *Quart. Appl. Math.* **16**, 87–90.
2. Du, D.-Z. and P.M. Pardalos (eds.) (1993), *Network Optimization Problems,* World Scientific.
3. Ericksson, R.E., C.L. Monma, and A.F. Veinott (1987), Send-and-split method for minimum concave-cost network flows, *Mathematics for Operations Research* **12**, 634–664.
4. Florian, M. and P. Robillard (1971), An implicit enumeration algorithm for the concave cost network flow problem, *Management Science* **18**, 184–193.

5. Fredman, M.L. and R.E. Tarjan (1984), Fibonaccy heaps and their uses in improved network optimization algorithms, *Proc. 25th IEEE Sympos. Foundations Computer Sci.,* 338–346.
6. Gallo, G., C. Sandi, and C. Sodini (1980), An algorithm for the min concave cost flow problem, *European Journal of Operations Research* **4**, 248–259.
7. Gallo, G. and C. Sodini (1979), Adjacent extreme flows and applications to min concave-cost flow problems, *Networks* **9**, 95–121.
8. Guisewite, G. and P.M. Pardalos (1990), Minimum concave-cost network flow problems: Applications, complexity and algorithms, *Annals of Operations Research* **25**, 75–100.
9. Guisewite, G. and P.M. Pardalos (1991), Algorithms for the single source uncapacitated minimum concave-cost network flow problem, *Journal of Global Optimization* **1**, 245–265.
10. Guisewite, G. and P.M. Pardalos (1992), A polynomial time solvable concave network flow problem, *Network* **23**, 143–147.
11. Guisewite, G. and P.M. Pardalos (1993), Complexity issues in nonconvex network flow problems, in *Complexity in Numerical Optimization*, ed. P.M. Pardalos, World Scientific, 163–179.
12. Holmberg, K. and H. Tuy (1993), A production-transportation problem with stochastic demands and concave production costs, Preprint, Department of Mathematics, Linköping University. Submitted.
13. Klinz, B. and H. Tuy (1993), Minimum concave-cost network flow problems with a single nonlinear arc cost, in *Network Optimization Problems*, eds. P.M. pardalos and D.-Z. Du, World Scientific, 125–143.
14. Lenstra, H.W. Jr. (1983), Integer programming with a fixed number of variables, *Mathematics of Operations Research* **8**, 538–548.
15. Minoux, M. (1989), Network synthesis and optimum network design problems: models, solution methods and applications, *Networks* **19**, 313–360.
16. Nemhauser, G.L. and L.A. Wolsey (1988), *Integer and Combinatorial Optimization*, John Wiley & Sons, New York.
17. Pardalos, P.M. and S.A. Vavasis (1992), Open questions in complexity theory for nonlinear optimization, *Math. Prog.* **57**, 337–339.
18. Tardos, E. (1985), A strongly polynomial minimum cost circulation algorithm, *Combinatorika* **5**, 247–255.
19. Thach, P.T. (1987), A decomposition method for the min concave-cost flow problem with a special structure, Preprint, Institute of Mathematics, Hanoi.
20. Thach, P.T. (1991), A dynamic programming method for min concave-cost flow problems on circuitless single source uncapacitated networks, Preprint, Institute of Mathematics, Hanoi.
21. Tuy, H (1992), The complementary convex structure in global optimization, *Journal of Global Optimization* **2**, 21–40.
22. Tuy, H. and B.T. Tam (1992), An efficient solution method for rank two quasiconcave minimization problems, *Optimization* **24**, 43–56.
23. Tuy, H., N.D. Dan, and S. Ghannadan (1993), Strongly polynomial time algorithm for certain concave minimization problems on networks, *Operations Research Letters* **14**, 99–109.
24. Tuy, H., S. Ghannadan, A. Migdalas, and P. Värbrand (1993), Strongly polynomial algorithm for a production-transportation problem with concave production cost, *Optimization* **27**, 205–228.
25. Tuy, H., S. Ghannadan, A. Migdalas, and P. Värbrand (1993), Strongly polynomial algorithms for two special minimum concave-cost network flow problems, *Optimization (to appear).*
26. Tuy, H., S. Ghannadan, A. Migdalas, and P. Värbrand (1993), Strongly polynomial algorithm for a production-transportation problem with a fixed number of nonlinear variables, *Mathematical Programming (to appear).*
27. Veinott, A.F. (1969), Minimum concave-cost solution of Leontiev substitution models of multi-facility inventory systems, *Operations Research* **17**, 262–291.
28. Wagner, H.M., and T.M. Whitin (1959), Dynamic version of the economic lot size model, *Management Science* **5**, 89–96.
29. Wallace, S.W. (1986), Decomposition of the requirement space of a transportation problem into polyhedral cones, *Mathematical Programming* **28**, 29–47.
30. Zangwill, W.I. (1968), Minimum concave cost flows in certain networks, *Management Science* **14**, 429–450.

31. Zangwill, W.I. (1969) A backlogging model and a multi-echelon model on a dynamic lot size production system—a network approach, *Management Science* **15**, 509–527.