# A Decomposition Algorithm for Solving Certain Classes of Production-Transportation Problems with Concave Production Cost

Takahito Kuno*
Takahiro Utsunomiya

ISE-TR-94-113

July 11, 1994

Institute of Information Sciences and Electronics
University of Tsukuba
Tsukuba, Ibaraki 305, Japan
Phone: +81-298-53-5540, Fax: +81-298-53-5206, E-mail: takahito@is.tsukuba.ac.jp

* The author was partly supported by Grant-in-Aid for Scientific Research of the Ministry of Education, Science and Culture, Grant No. (C)05650061.

# A Decomposition Algorithm for Solving Certain Classes of Production-Transportation Problems with Concave Production Cost

Takahito Kuno* and Takahiro Utsunomiya
*Institute of Information Sciences and Electronics*
*University of Tsukuba*

July 1994

**Abstract.** This paper addresses a method for solving two classes of production-transportation problems with concave production cost. By exploiting a special network structure both the problems are reduced to a kind of resource allocation problem. It is shown that the resulting problem can be solved by using dynamic programming in time polynomial in the problem input length, i.e., the number of supply and demand points and the total demand.

**Key words:** Concave minimization, global optimization, production-transportation problem, resource allocation problem, dynamic programming

## 1. Introduction

In this paper we will discuss special classes of production-transportation problems which arise in many practical applications, for instance:

There are one factory and a number of warehouses in each of several regions. Every factory produces a certain amount of goods, and can transport them to only warehouses in its assigned region. Except these branch factories there is a head factory, which can transport the products to every warehouse. The decision maker has to decide how many goods each factory should produce, and which warehouses the head factory should supply, so as to minimize the total production and transportation cost.

In the above situation (see also Figure 2.1), we are concerned with two cases:

(P1): The production cost of the head factory need not to be considered but the total supply from it is restricted.

(P2): The total supply from the head factory is not restricted but its production cost has to be considered.

*The author was partially supported by Grand-in-Aid for Scientific Research of the Ministry of Education, Science and Culture, Grant No. (C)05650061.

The production cost of each factory is in general a nondecreasing and concave function of the output, whence both the problems (P1) and (P2) have multiple locally optimal solutions, many of which need not to be globally optimal. If the number of factories are fixed at $k$, we can solve the problems in strongly polynomial time by using the algorithms proposed by Tuy, et al. in their recent series of articles [9, 10, 11]. In this case, the total cost function possesses rank-$k$ property [8] and its global minimum can be found in the course of solving a transportation problem parametrically. We will show in this paper that both the problems can be solved in time polynomial in the problem input length, i.e., the number of factories and warehouses and the total demand of warehouses, without assuming the fixed number of branch factories.

The organization of the paper is as follows: In Section 2, we will transform the problem (P1) into a kind of resource allocation problem, referred to as the *master problem* of (P1), by exploiting the special network structure stated above. The objective function of the master problem is defined by solving $m$ Hitchcock transportation problems, where $m$ represents the number of branch factories. In Section 3, to solve the master problem we will propose an algorithm using dynamic programming, and show that it requires $(mnb)$ arithmetic operations and $O(nb)$ evaluations of the production cost function of each factory, where $n$ and $b$ represent the number of warehouses and their total demand respectively. In Section 4, we will show that the problem (P2) can also be transformed into a resource allocation problem of the same form as the master problem of (P1).

## 2. Decomposition of (P1) into subproblems

The problem we first consider is defined below (see Figure 2.1):

$$\left|\begin{array}{lll} \text{minimize} & \sum_{i=0}^{m}\sum_{j\in V_i} c_{ij}x_{ij} + \sum_{i=1}^{m} f_i(z_i) & \\ \text{subject to} & \sum_{i=1}^{m} y_i \le b, & \\ & \sum_{j\in V_i} x_{0j} = y_i, \quad \sum_{j\in V_i} x_{ij} = z_i, & i = 1, \ldots, m, \\ & x_{0j} + x_{ij} = b_j, \qquad j \in V_i, & i = 1, \ldots, m, \\ & x_{0j} \ge 0, \quad x_{ij} \ge 0, \quad j \in V_i, & i = 1, \ldots, m, \\ & y_i \ge 0, \quad z_i \ge 0, & i = 1, \ldots, m, \end{array}\right. \tag{2.1}$$

where $b$, $b_j > 0$, $j \in V_i$, $i = 1, \ldots, m$, are integral, $c_{ij} \ge 0$, $j \in V_i$, $i = 0, 1, \ldots, m$, are real, $f_i : R^1 \to R^1$, $i = 1, \ldots, m$, are nondecreasing and concave functions, and $V_i$, $i = 0, 1, \ldots, m$, are index sets such that

$$\bigcup_{i=1}^{m} V_i = V_0 = \{1, \ldots, n\}, \quad \bigcap_{i=1}^{m} V_i = \emptyset. \tag{2.2}$$

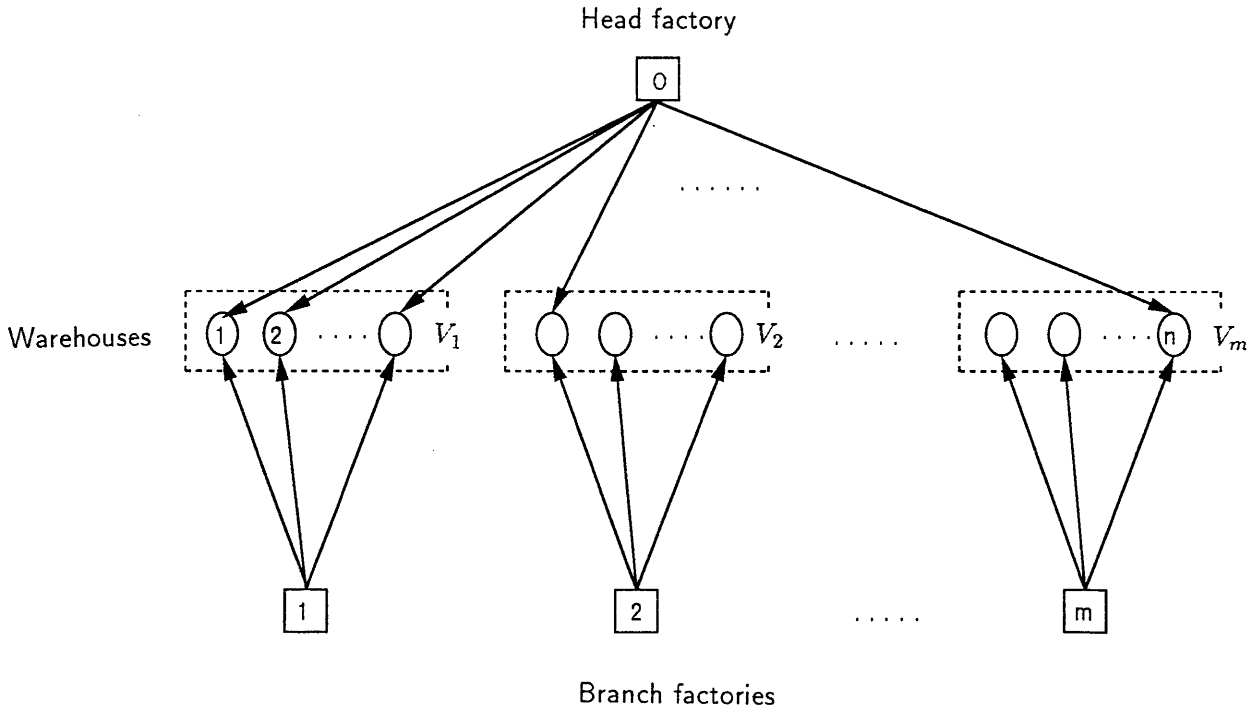


Figure 2.1. Illustration of the problem.

A special case of (2.1), where $m = 1$, involves the problem studied by Tuy et al. in [9], and can be solved in $O(n \log n)$ elementary operations and $n$ evaluations of function $f_1$ if we use the algorithm proposed in [9].

Any feasible solution of (2.1) has to satisfy

$$z_i = a_i - y_i, \quad i = 1, \ldots, m, \tag{2.3}$$

where $a_i = \sum_{j \in V_i} b_j$. We can therefore eliminate all $z_i$'s from (2.1) by defining

$$\bar{f}_i(y_i) = f_i(a_i - y_i), \quad i = 1, \ldots, m. \tag{2.4}$$

Obviously $\bar{f}_i$'s are still concave but nonincreasing. The first problem is then as follows:

$$\text{(P1)} \left| \begin{array}{lllll} \text{minimize} & \sum_{i \in 0}^{m} \sum_{j \in V_i} c_{ij} x_{ij} + \sum_{i=1}^{m} \bar{f}_i(y_i) & & & \\ \text{subject to} & \sum_{i=1}^{m} y_i \leq b, & & & \\ & \sum_{j \in V_i} x_{0j} = y_i, \quad \sum_{j \in V_i} x_{ij} = a_i - y_i, & & i = 1, \ldots, m, & \\ & x_{0j} + x_{ij} = b_j, & j \in V_i, & i = 1, \ldots, m, & \\ & x_{0j} \geq 0, \quad x_{ij} \geq 0, & j \in V_i, & i = 1, \ldots, m, & \\ & y_i \geq 0, & & i = 1, \ldots, m. & \end{array} \right. \tag{2.5}$$

## 2.1. Definition of master problem

For any fixed $\boldsymbol{y} = (y_1, \ldots, y_m)$, we have a linear programming problem:

$$
(\mathrm{P}(\boldsymbol{y}))\quad
\left|
\begin{array}{lllll}
\text{minimize} & \displaystyle\sum_{i\in 0}^{m}\sum_{j\in V_i} c_{ij}x_{ij} & & & \\
\text{subject to} & \displaystyle\sum_{j\in V_i} x_{0j} = y_i, & \displaystyle\sum_{j\in V_i} x_{ij} = a_i - y_i, & i = 1, \ldots, m, & \\
& x_{0j} + x_{ij} = b_j, & j \in V_i, & i = 1, \ldots, m, & \\
& x_{0j} \geq 0, \quad x_{ij} \geq 0, & j \in V_i, & i = 1, \ldots, m. &
\end{array}
\right.
\tag{2.6}
$$

Due to the condition (2.2), problem $(\mathrm{P}(\boldsymbol{y}))$ can be decomposed into $m$ subproblems $(\mathrm{P}_i(y_i))$, $i = 1, \ldots, m$, each of which is a Hitchcock transportation problem with two supply points:

$$
(\mathrm{P}_i(y_i))\quad
\left|
\begin{array}{lll}
\text{minimize} & \displaystyle\sum_{j\in V_i} (c_{0j}x_{0j} + c_{ij}x_{ij}) & \\
\text{subject to} & \displaystyle\sum_{j\in V_i} x_{0j} = y_i, \quad \sum_{j\in V_i} x_{ij} = a_i - y_i\ , & \\
& x_{0j} + x_{ij} = b_j, & j \in V_i, \\
& x_{0j} \geq 0, \quad x_{ij} \geq 0, & j \in V_i.
\end{array}
\right.
\tag{2.7}
$$

If $0 \leq y_i \leq a_i$, then $(\mathrm{P}_i(y_i))$ has an optimal solution. We denote it by a vector $\boldsymbol{x}_i^*(y_i)$, every element of which is either $x_{0j}^*(y_i)$ or $x_{ij}^*(y_i)$, $j \in V_i$, and by $g_i(y_i)$ its optimal value. Obviously $\boldsymbol{x}^*(\boldsymbol{y}) = (x_1^*(y_1), \ldots, x_m^*(y_m))$ is an optimal solution of $(\mathrm{P}(\boldsymbol{y}))$ and $\sum_{i=1}^m g_i(y_i)$ is the optimal value. The original problem (P1) can be solved if we solve $(\mathrm{P}(\boldsymbol{y}))$ for all $\boldsymbol{y}$ satisfying $\sum_{i=1}^m y_i \leq b$ and $0 \leq y_i \leq a_i$ for every $i$. Let

$$
h_i(y_i) = \bar{f}_i(y_i) + g_i(y_i), \quad i = 1, \ldots, m. \tag{2.8}
$$

Then (P1) is reduced to a kind of resource allocation problem with $m$ variables:

$$
(\mathrm{MP1})\quad
\left|
\begin{array}{ll}
\text{minimize} & \displaystyle\sum_{i=1}^{m} h_i(y_i) \\
\text{subject to} & \displaystyle\sum_{i=1}^{m} y_i \leq b, \\
& 0 \leq y_i \leq a_i, \quad i = 1, \ldots, m,
\end{array}
\right.
\tag{2.9}
$$

which we call the *master problem* of (P1). Without loss of generality we may assume that

$$
b \leq \sum_{i=1}^{m} a_i \ (= \sum_{j=1}^{n} b_j). \tag{2.10}
$$

The following lemma summarizes the above arguments:

**Lemma 2.1.** *If $\boldsymbol{y}^*$ is an optimal solution of (MP1), then $(\boldsymbol{x}^*(\boldsymbol{y}^*), \boldsymbol{y}^*)$ solves (P1), where $\boldsymbol{x}^*(\boldsymbol{y}^*) = (\boldsymbol{x}_1^*(y_1^*), \ldots, \boldsymbol{x}_m^*(y_m^*))$ and $\boldsymbol{x}_i^*(y_i^*)$ is an optimal solution of $(P_i(y_i^*))$.* □

## 2.2. Analytic form of function $h_i$

To solve (MP1) we have to know the form of function $h_i$, which is a composition of two functions $\bar{f}_i$ and $g_i$. While the former is given beforehand, the latter requires one to solve the Hitchcock transportation problem ($\mathrm{P}_i(y_i)$) as varying the value of $y_i$ in the interval $[0,\ a_i]$.

Note that the constraint $\sum_{j \in V_i} x_{ij} = a_i - y_i$ is expressed by the others and hence can be deleted from the definition of ($\mathrm{P}_i(y_i)$), i.e.,

$$\left|\begin{array}{ll} \text{minimize} & \sum_{j \in V_i} (c_{0j} x_{0j} + c_{ij} x_{ij}) \\ \text{subject to} & \sum_{j \in V_i} x_{0j} = y_i, \\ & x_{0j} + x_{ij} = b_j, \qquad j \in V_i, \\ & x_{0j} \geq 0, \quad x_{ij} \geq 0, \quad j \in V_i. \end{array}\right. \tag{2.11}$$

We should also note that any feasible solution satisfies

$$x_{ij} = b_j - x_{0j}, \quad \forall j \in V_i. \tag{2.12}$$

Then, by substituting (2.12) into (2.11), we have an equivalent problem with $|V_i|$ variables:

$$(\mathrm{Q}_i(y_i)) \left|\begin{array}{ll} \text{minimize} & \sum_{j \in V_i} (c_{0j} - c_{ij}) x_{0j} + \sum_{j \in V_i} c_{ij} b_j \\ \text{subject to} & \sum_{j \in V_i} x_{0j} = y_i, \\ & 0 \leq x_{0j} \leq b_j, \quad j \in V_i. \end{array}\right. \tag{2.13}$$

The resulting problem ($\mathrm{Q}_i(y_i)$) is a continuous knapsack problem, which can be solved very efficiently for all $y_i \in [0,\ a_i]$. Let

$$\bar{c}_j = c_{0j} - c_{ij}, \quad j \in V_i, \tag{2.14}$$

and let

$$\bar{c}_{j_1} \leq \bar{c}_{j_2} \leq \cdots \leq \bar{c}_{j_{m(i)}}, \tag{2.15}$$

where $m(i) = |V_i|$. The following is a well-known theorem (see e.g. [1]), which characterizes the form of $g_i$:

**Lemma 2.2.** *If $0 \leq y_i \leq a_i$, then there exists some $p$ such that*

$$x^*_{0j_k} = \begin{cases} b_{j_k}, & k = 1, \ldots, p-1, \\ y_i - \sum_{\ell=1}^{p-1} b_{j_\ell}, & k = p, \\ 0, & k = p+1, \ldots, |V_i| \end{cases} \tag{2.16}$$

*is an optimal solution of* $(Q_i(y_i))$ *and*

$$g_i(y_i) = \sum_{\ell=1}^{p-1} \bar{c}_{j_\ell} b_{j_\ell} + \bar{c}_p(y_i - \sum_{\ell=1}^{p-1} b_{j_\ell}) + \sum_{j \in V_i} c_{ij} b_j \tag{2.17}$$

*is the optimal value.* □

Let

$$a_{i0} = 0, \quad a_{ik} = \sum_{\ell=1}^{k} b_{j_\ell}, \quad k = 1, \ldots, |V_i|, \tag{2.18}$$

and let

$$I_{ik} = [a_{i,k-1}, \, a_{ik}], \quad k = 1, \ldots, |V_i|. \tag{2.19}$$

The analytic form of $h_i$ is now identified:

**Lemma 2.3.** *Function* $h_i$ *is concave on* $I_{ik}$ *for every* $k = 1, \ldots, |V_i|$.

*Proof:* We immediately see from (2.16) and (2.17) that $g_i$ is a convex and piecewise linear function with break points among $a_{ik}$, $k = 0, 1, \ldots, |V_i|$. Hence $h_i = \bar{f}_i + g_i$ is concave on each linear piece $I_{ik} = [a_{i,k-1}, \, a_{ik}]$ of $g_i$, because $\bar{f}_i$ is a concave function defined on $R$. □

In [9] Tuy et al. have derived the same result as Lemma 2.3. They straightforward used the network structure of $(\mathrm{P}_i(y_i))$ instead of transforming it into the continuous knapsack problem.

## 3. Solution Method for the Master Problem (MP1)

Let us proceed to the algorithm for solving the master problem (MP1). We will show that (MP1) can be solved using dynamic programming. For this purpose let us note some properties of its optimal solutions.

**Lemma 3.1.** *Problem (MP1) has an optimal solution* $\boldsymbol{y}^* = (y_1^*, \ldots, y_m^*)$, *at least* $m-1$ *elements of which lie in* $a_{ik}$, $k = 1, \ldots, |V_i|$, $i = 1, \ldots, m$.

*Proof:* Since $b$ and all $a_i$'s are positive, the feasible region of (MP1) is nonempty and bounded. Every $h_i$ is continuous on $[0, \, a_i]$, and hence the objective function of (MP1) attains the minimum at some $\boldsymbol{y}^*$ in the feasible region. Suppose there are two elements of $\boldsymbol{y}^*$, say $y_p^*$ and $y_q^*$, which are not in $a_{ik}$'s. Let $y_p^* \in \operatorname{int} I_{ps} = (a_{p,s-1}, \, a_{ps})$, $y_q^* \in \operatorname{int} I_{qt} = (a_{q,t-1}, \, a_{qt})$, and let

$$h_{pq}(y_p) = h_p(y_p) + h_p(b - y_p),$$

where $b = y_p^* + y_q^*$. Also let

$$\underline{d} = \max\{a_{p,s-1},\ b - a_{qt}\},\quad \overline{d} = \min\{a_{ps},\ b - a_{q,t-1}\}.$$

Then $y_p^* \in (\underline{d},\ \overline{d})$ and $h_{pq}$ is concave on $[\underline{d},\ \overline{d}]$. Hence we have

$$h_{pq}(y_p^*) \geq \min\{h_{pq}(\underline{d}),\ h_{pq}(\overline{d})\},$$

which implies that if we replace $y_p^*$, $y_q^*$ with either $\underline{d}$, $b - \underline{d}$ or $\overline{d}$, $b - \overline{d}$ then another optimal solution $\boldsymbol{y}'$ of (MP1) is provided. In this case, either $y_p'$ or $y_q'$ coincides with an extreme point of its interval. □

Consider $m$ discrete optimization problems $(\mathrm{DP}_i(y_i))$, $i = 1, \ldots, m$, associated with (MP1):

$$(\mathrm{DP}_i(y_i))\ \left|\ \begin{array}{ll} \text{minimize} & \sum_{\ell \neq i} h_\ell(y_\ell) \\ \text{subject to} & \sum_{\ell \neq i} y_\ell \leq b - y_i, \\ & y_\ell \in \{a_{\ell 0},\ \ldots,\ a_{\ell m(\ell)}\},\quad \ell \neq i. \end{array}\right. \tag{3.1}$$

We denote by $H_i(y_i)$ the optimal value of $(\mathrm{DP}_i(y_i))$. It follows from Lemma 3.1 that an optimal solution $\boldsymbol{y}^*$ of (MP1) will be found if we solve every $(\mathrm{DP}_i(y_i))$ for $y_i \in [0,\ a_i]$. Namely,

$$\min\{\min\{h_i(y_i) + H_i(y_i) \mid y_i \in [0,\ a_i]\} \mid i = 1,\ \ldots,\ m\} \tag{3.2}$$

is the minimum value of the objective function of (MP1).

**Lemma 3.2.** *For each $i$ there exists an integer $y_i' \in [0,\ a_i]$ such that*

$$h_i(y_i') + H_i(y_i') = \min\{h_i(y_i) + H_i(y_i) \mid y_i \in [0,\ a_i]\}. \tag{3.3}$$

*Proof:* Let $y_i' \in I_{is}$ and suppose $y_i'$ is not integral. Since $b$ and all $a_{\ell k}$'s are integral, it must hold that

$$H_i(y_i') = H_i(\lceil y_i' \rceil) \geq H_i(\lfloor y_i' \rfloor),$$

where $\lceil\cdot\rceil$ and $\lfloor\cdot\rfloor$ represent the integers obtained by rounding up and down · respectively. Hence we have

$$h_i(y_i') + H_i(y_i') \geq \min\{h_i(\lceil y_i' \rceil) + H_i(\lceil y_i' \rceil),\ h_i(\lfloor y_i' \rfloor) + H_i(\lfloor y_i' \rfloor)\}$$

by noting concavity of $h_i$ on $[\lfloor y_i' \rfloor,\ \lceil y_i' \rceil] \subset I_{is}$. □

Thus (3.2) turns out to be

$$\min\{\min\{h_i(y_i) + H_i(y_i) \mid y_i = 0,\ 1,\ \ldots,\ a_i\} \mid i = 1,\ \ldots,\ m\}. \tag{3.4}$$

### 3.1. Dynamic programming recursion

Let us define a partial problem of $(\mathrm{DP}_i(y_i))$:

$$(\mathrm{DP}_i^p(y_i)) \left| \begin{array}{ll} \text{minimize} & \sum_{\ell \in M(i,p)} h_\ell(y_\ell) \\ \text{subject to} & \sum_{\ell \in M(i,p)} y_\ell \leq b - y_i, \\ & y_\ell \in \{a_{\ell 0}, \ldots, a_{\ell m(\ell)}\}, \quad \ell \in M(i,p), \end{array} \right. \tag{3.5}$$

where $M(i,p) = \{1, \ldots, i-1, i+1, \ldots, p\}$. Denote by $H_i^p(y_i)$ the optimal value of $(\mathrm{DP}_i^p(y_i))$ and let

$$H_i^p(y_i) = \begin{cases} 0 & \text{if } y_i \leq b,\ p = 0, \\ +\infty & \text{if } y_i > b,\ p = 0 \text{ or } y_i \geq b,\ p > 0. \end{cases} \tag{3.6}$$

**Lemma 3.3.** *The values $H_i^p(y_i)$'s satisfy the following recursive formula:*

$$H_i^p(y_i) = \min\{h_p(a_{pk}) + H_i^{p-1}(y_i + a_{pk}) \mid k = 1, \ldots, |V_p|\} \tag{3.7}$$

*Proof:* By definition we have

$$\begin{aligned} H_i^p(y_i) &= \min\{h_p(y_p) + H_i^{p-1}(y_i + y_p) \mid y_p \in \{a_{p0}, \ldots, a_{pm(p)}\}\} \\ &= \min\{h_p(a_{pk}) + H_i^{p-1}(y_i + a_{pk}) \mid k = 1, \ldots, |V_p|\} \qquad \square \end{aligned}$$

Since $H_i(y_i) = H_i^m(y_i)$, to obtain $H_i(y_i)$ for all $y_i \in [0, a_i]$ it is enough to compute $H_i^p(y_i)$ for $p = 1, \ldots, i-1, i+1, \ldots, m$ and $y_i = b, b-1, \ldots, 1, 0$.

**Remark.** Problem $(\mathrm{DP}_i(y_i))$ can easily be transformed into a multiple-choice knapsack problem (see also [6]): Letting

$$w_{\ell k} = \begin{cases} 1 & \text{if } y_\ell = a_{\ell k}, \\ 0 & \text{otherwise}, \end{cases} \quad k = 1, \ldots, |V_\ell|, \quad \ell \neq i, \tag{3.8}$$

then we have an equivalent problem:

$$\left| \begin{array}{ll} \text{minimize} & \sum_{\ell \neq i} \sum_{k=1}^{m(\ell)} h_\ell(a_{\ell k}) w_{\ell k} \\ \text{subject to} & \sum_{\ell \neq i} \sum_{k=1}^{m(\ell)} a_{\ell k} w_{\ell k} \leq b - y_i, \\ & \sum_{k=1}^{m(\ell)} w_{\ell k} = 1, \quad \ell \neq i, \\ & w_{\ell k} \in \{0, 1\}, \quad k = 1, \ldots, |V_\ell|, \quad \ell \neq i. \end{array} \right. \tag{3.9}$$

We will obtain a similar recursive formula as (3.7) from this 0-1 integer programming problem [2]. $\square$

We are now ready to present the algorithm for solving the target problem (P1):

**Algorithm A.**

*Step 1.* For $i = 1, \ldots, m$ do the following:

1° Compute $\bar{c}_j$, $j \in V_i$, from (2.14) and sort them as $\bar{c}_{j_1} \leq \bar{c}_{j_2} \leq \cdots \leq \bar{c}_{j_{m(i)}}$, where $m(i) = |V_i|$.

2° Let $a_{i0} = 0$, $a_{ik} = \sum_{\ell=1}^{k} b_{j_\ell}$, $k = 1, \ldots, |V_i|$.

*Step 2.* For $i = 1, \ldots, m$ do the following:

1° Compute $H_i^p(y_i)$ according to (3.6) and (3.7) in the order $p = 1, \ldots, i-1, i+1, \ldots, m$; $y_i = b-1, b-2, \ldots, 1, 0$.

2° Let

$$y_i' = \operatorname{argmin}\{h_i(y_i) + H_i^m(y_i) \mid y_i = 0, 1, \ldots, a_i\} \tag{3.10}$$

and let $v_i = h_i(y_i') + H_i^m(y_i')$.

*Step 3.* Let

$$v_r = \min\{v_1, v_2, \ldots, v_m\}, \tag{3.11}$$

and let $y_r^* = y_r'$. Also let $y_i^*$, $i \in M(r, m)$, be an optimal solution of $(\mathrm{DP}_r(y_r'))$. Then an optimal solution $\boldsymbol{x}^*(\boldsymbol{y}^*)$ of $(\mathrm{P}(\boldsymbol{y}^*))$ is optimal to (P1). □

**Theorem 3.4.** *Algorithm A requires $O(mnb)$ arithmetic operations and $O(nb)$ evaluations of $f_i$ for each $i = 1, \ldots, m$.*

*Proof:* Step 1 can be carried out in time $O(n \log n)$ for sorting $\bar{c}_j$'s. If Step 1 is over, then for any $\boldsymbol{y}^*$ we will have $\boldsymbol{x}^*(\boldsymbol{y}^*)$ in time $O(\log n)$ using binary search. The total computational time of the algorithm is therefore dominated by Step 2. 1°: It takes $2|V_p|$ additions, $|V_p| - 1$ comparisons and $|V_p|$ evaluations of $f_i$ to compute every $H_i^p(y_i)$. For each $i$ those numbers are bounded by

$$\sum_{y_i=0}^{b} \sum_{p \in M(i,m)} O(|V_p|) = O(nb).$$

Hence the total number of arithmetic operations is $\sum_{i=1}^{m} O(nb) = O(nmb)$. □

In general, Algorithm A does not run in time polynomial in the size of $m$, $n$, even though the values of $f_i$ are provided by an oracle. However, when all $b_j$'s have a common value, say one, the number of arithmetic operations is a polynomial function of $m$ and $n$. In this case, the value of $b$ is bounded by $\sum_{i=1}^{m} |V_i| = n$ under the assumption (2.10), and hence the total number of arithmetic operations becomes $O(mnb) = O(mn^2)$.

## 4. Application of the Algorithm to (P2) and Other Problems

The second problem is as follows:

$$
\text{(P2)}\quad \left|\begin{array}{lllr}
\text{minimize} & \displaystyle\sum_{i\in 0}^{m}\sum_{j\in V_i} c_{ij}x_{ij} + \sum_{i=1}^{m} \bar{f}_i(y_i) + f_0(z_0) & & \\
\text{subject to} & \displaystyle\sum_{i=1}^{m} y_i = z_0, & & \\
 & \displaystyle\sum_{j\in V_i} x_{0j} = y_i, \quad \sum_{j\in V_i} x_{ij} = a_i - y_i, & i = 1, \ldots, m, & \\
 & x_{0j} + x_{ij} = b_j, \qquad j \in V_i, & i = 1, \ldots, m, & \\
 & x_{0j} \geq 0, \quad x_{ij} \geq 0, \quad j \in V_i, & i = 1, \ldots, m, & \\
 & z_0 \geq 0, \quad y_i \geq 0, & i = 1, \ldots, m, &
\end{array}\right. \tag{4.1}
$$

where $f_0$ is a nondecreasing and concave function of $z_0$, and all of the other notations are the same as (P1). As before, we can define the master problem of (P2):

$$
\text{(MP2)}\quad \left|\begin{array}{ll}
\text{minimize} & \displaystyle\sum_{i=1}^{m} h_i(y_i) + f_0(z_0) \\
\text{subject to} & \displaystyle\sum_{i=1}^{m} y_i = z_0, \\
 & z_0 \geq 0, \quad 0 \leq y_i \leq a_i, \quad i = 1, \ldots, m,
\end{array}\right. \tag{4.2}
$$

where $a_i = \sum_{j\in V_i} b_j$, $h_i(y_i) = \bar{f}_i(y_i) + g_i(y_i)$ and $g_i(y_i)$ is the optimal value of the Hitchcock transportation problem $(\mathrm{P}_i(y_i))$. If we obtain an optimal solution $(\boldsymbol{y}^*, z_0^*)$ of (MP2), then $(\boldsymbol{x}^*(\boldsymbol{y}^*),\ \boldsymbol{y}^*,\ z_0^*)$ solves (P2), where $\boldsymbol{x}^*(\boldsymbol{y}^*)$ is an optimal solution of $(\mathrm{P}(\boldsymbol{y}^*))$ defined by (2.6).

Let $a_0 = \sum_{i=1}^{m} a_i$ and let

$$
y_0 = a_0 - z_0. \tag{4.3}
$$

For any feasible solution of (MP2) we have $0 \leq y_0 \leq a_0$, since $0 \leq z_0 \leq a_0$ must be satisfied. Also let

$$
h_0(y_0) = f_0(a_0 - y_0). \tag{4.4}
$$

Then $h_0$ is a concave function on $I_{01} = [0,\ a_0]$ , and (MP2) is reformulated as

$$
\left|\begin{array}{ll}
\text{minimize} & \displaystyle\sum_{i=0}^{m} h_i(y_i) \\
\text{subject to} & \displaystyle\sum_{i=0}^{m} y_i = a_0, \\
 & 0 \leq y_i \leq a_i, \quad i = 0, 1, \ldots, m,
\end{array}\right. \tag{4.5}
$$

which is of just the same form as (MP1). We again apply dynamic programming to (4.5), then an optimal solution of (P2) will be generated by Algorithm A in $O(mnb)$ arithmetic operations and $O(nb)$ evaluations of $f_i$ for $i = 0, 1, \ldots, m$, where $b = \sum_{j=1}^{n} b_j$.

### 4.1. Network flow problems associated with (P1) and (P2)

Minimum concave-cost flow problems is one of the most important and most difficult classes in both combinatorial and global optimization. To solve them many algorithms have been proposed so far (see [5, 3] and references therein), and some of them have turned out to be practically efficient for special problems. In particular, when the number of concave-cost arcs is fixed, one can solve the problem in polynomial time [4, 9, 12].

As well known, every Hitchcock transportation problem can be transformed into a minimum cost flow problem and vice versa (see e.g. [7]). Similarly, we can generate a minimum concave-cost flow problem from either (P1) or (P2) by equipping a super-source and $m$ additional concave-cost arcs with the underlying network. The converse is also possible in a similar way as in [9, 12], i.e., a certain class of minimum concave-cost network flow problems with $m$ concave-cost arcs can be transformed into either (P1) or (P2), the detail of which will be discussed in the subsequent paper.

## References


[1] Dantzig, G.B., *Linear Programming and Extensions,* Princeton University Press (Princeton, NJ, 1963).

[2] Dudziński, k. and S. Walukiewicz, "Exact methods for the knapsack problem and its generalizations," *European Journal of Operational Research* **28** (1987), 3 – 21.

[3] Guisewite, G.M. and P.M. Pardalos, "Minimum concave-cost network flow problems: applications, complexity and algorithms," *Annals of Operations Research* **25** (1990), 75 – 100.

[4] Guisewite, G.M. and P.M. Pardalos, "A polynomial time solvable concave network flow problem," *Networks* **23** (1993), 143 – 149.

[5] Horst, R. and H. Tuy, *Global Optimization: Deterministic Approaches,* Springer-Verlag (Berlin, 1990).

[6] Nauss, R.M., "The 0-1 Knapsack problem with multiple choice constraints," *European Journal of Operational Research* **2** (1978), 125 – 131.

[7] Papadimitriou, C.H. and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity,* Princeton-Hall Inc. (Englewood Cliffs, NJ, 1982).

[8] Tuy, H., "The complementarity convex structure in global optimization," *Journal of Global Optimization* **2** (1992), 21 – 40.

[9] Tuy, H., N.D. Dan and S. Ghannadan, "Strongly polynomial time algorithms for certain concave minimization problems on networks," *Operations Research Letters* **14** (1993), 99 – 109.

[10] Tuy, H., S. Ghannadan, A. Migdalas and P. Värbrand, "Strongly polynomial algorithm for a production-transportation problem with concave production cost,"

*Optimization* **27** (1993), 205 - 227.

[11] Tuy, H., S. Ghannadan, A. Migdalas and P. Värbrand, "Strongly polynomial algorithm for a production-transportation problem with a fixed number of nonlinear variables," Preprint, Department of Mathematics, Linköping University (Linköping, 1993).

[12] Tuy, H., S. Ghannadan, A. Migdalas and P. Värbrand, "The minimum concave cost network flow problem with a fixed number of sources and nonlinear arc costs," Preprint, Department of Mathematics, Linköping University (Linköping, 1993).