International Journal of Management Science
Vol. 1, No. 1, December 1995

A Decomposition Method for the Multi-stage Dynamic Location Problem[1)]

Hyunkon Kahng

Abstract

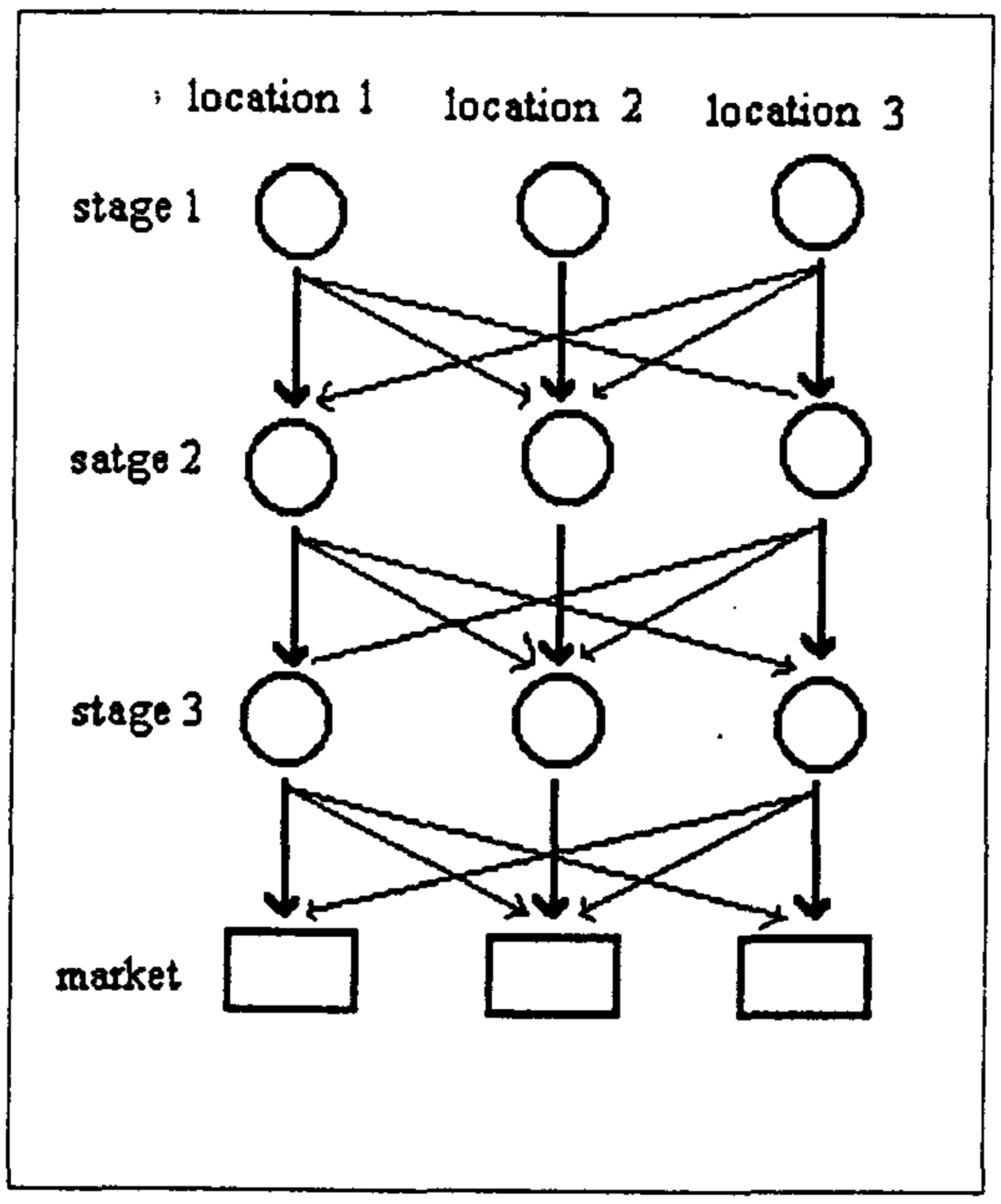
This paper suggests a procedure of decomposing a multi-stage dynamic location problem into stages with respect stage. The problem can be formulated as a mixed integer programming problem, which is difficult to be solved directly. We perform a series of transformations in order to divide the problem into stages. In the process, the assumption of PSO (production-system-only) plays a critical role. The resulting subproblem becomes a typical single-stage dynamic location problem, whose efficient algorithms have been developed efficiently. An extension of this study is to find a method to integrate the solutions of subproblems for a final solution of the problem.

1. Introduction

In many industries, it becomes common for a firm to operate plants in several countries. Through such a global manufacturing system the firm may be able to save costs or avoid trade barriers. As a result, more and more products manufactured in one country are finding their way into other countries. These products are not just finished goods; they also include intracompany transfers such as components and processed materials.

1) This research is supported by the 1994 fund from the office of research affai, Yonsei University.

Television manufacturing is typical of this type of industry. For example, Panasonic has been producing color chassis in its Singapore plants. Such a firm has to decide whether to establish and expand an international network of plants. In other words, it must decide when and where to build plants (or add capacity) over a planning horizon of several years. It also must determine the optimal production / logistics schedule for the products.



[Figure 1] An example of Problem (P)

This problem (P) generically belongs to the multi-stage dynamic location problem. Figure 1 shows a simple example of problem (P) that has a serial structure and three stages and three locations for a given time period. Problem (P) can be easily formulated as a mixed integer programming (MIP) problem. Unfortunately, a MIP problem is typically difficult to solve directly. A popular way to attack a MIP problem is a decomposition approach. The most famous work to attack a multi-stage static location problem was done by Geoffrion and Graves [6]. Their approach is to develop a single-stage path formulation from a multi-stage network formulation and to apply Bender's decomposition algorithm. Another study with a multi-stage context is the multi-stage inventory problem. It is decomposed into easily-solvable single-stage subproblems by using the "echelon inventory" concept (Clark and Scarf [2]).

The purpose of this paper is to suggest a procedure of decomposing the problem (P) into small subproblems with respect to stage. We use an extended version of the "echelon inventory" concept in decomposing the problem. After a successful decomposition, the resulting subproblem becomes the typical single-stage dynamic location problem, whose efficient solution approaches have been developed (Erlenkotter [3], Van Roy and Erlenkotter [9], Fong and Srinivasan [4, 5], Schulman [8]).

We first model the problem as a mixed integer programming problem (MIP). One key assumption of this problem is the assumption of production-system-only (PSO), which says that only production plants are considered. (no warehouses) The model itself is not appropriate to be decomposed by stages. We perform a series of transformations for the model. The modified problem is decomposed by stages. Since the resulting subproblem, which is an arc-based formulation, no longer maintains the assumption of PSO, it is transformed into a path-based formulation by finding the shortest path from each source and adding its cost to the production costs of each source. We find the shortest paths by applying the Bellman-Ford algorithm with the upper bound of the number of arcs which represents the assumption of PSO. The path-based subproblem becomes a single-stage dynamic location problem. Finally, we make some conclusive remarks and suggest some extensions of this paper. One of such extensions is to study how to integrate solutions of the subproblems to find a final solution for this problem.

2. A model of the Problem and Associated Transformations

2. 1. An MIP Model of the Problem: Problem (P1)

2. 1. 1. Notation

We use the following notation in formulating the problem.

k, j = index of the plants

t = index of the time periods

i = index of the items

X_{kti} = units produced of item i at plant k in time period t. It has a production cost c_{kti}

T_{kjti} = the number of units of item i transferred from plant k to plant j in time period t. It has a transfer cost u_{kjti}

Z_{kt} = the number of newly opened plants of plant k in time period t. It has a fixed cost f_{kt}

$S(k)$ = a set of immediate successors of plant k

$s^*(k)$ = the local immediate successor of plant k. "Local" means "located in the same location"

$S'(k)$ = $S(k)-s^*(k)$. a set of foreign immediate successors of plant k. "Foreign" means "located in a different location"

$p^*(k)$ = a set of local immediate predecessors of plant k

$J(k)$ = a set of foreign plants of the same stage[2] as plant k

$m(i)$ = the market for item i

$K(i)$ = a set of plants located at the location of $m(i)$

F = a set of plants at the final stage

$F1(i)$ = $F \cap K(i)$. This is the final plant located at the location of $m(i)$

$F2(i)$ = $F - F1(i)$. A set of final plants that are not located at the location of $m(i)$

r_{ki} = the number of units of the product produced at plant k that are required for one unit of product produced at its immediate successor for item i. $r_{ki} = r_{ji}$ where $j \in J(k)$

2) A "stage" is a set of plants in various locations, all of which make the same part(or more generally, perform the same manufacturing process).

r^*_{ki} = the number of units of the product produced at plant k that are required for one unit of the final product for item i. $r^*_{ki} = r^*_{ji}$ where $j \in J(k)$

q_k = capacity of plant k. $q_k = q_j$ where $j \in J(k)$

d_{ti} = demand for item i in time period t

2. 1. 2. Formulation

We have a mixed integer programming (MIP) problem, denoted Problem (P1).

Problem (P1)

$$\text{MIN} \sum_k \sum_t \sum_i (c_{kti} \cdot X_{kti} + \sum_{j \in S(k)} u_{kjti} \cdot T_{kjti}) + \sum_k \sum_t f_{kt} \cdot Z_{kt}$$

subject to

$X_{kti} = \sum_{j \in S(k)} T_{kjti}$	for $k \notin F$, t, i	(11)
$X_{kti} = T_{km(i)ti}$	for $k \in F$, t, i	(12)
$T_{ks^*(k)ti} + \sum_{j \in J(k)} T_{js^*(k)ti} = r_{ki} \cdot X_{s^*(k)ti}$	for $k \notin F$, t, i	(13)
$\sum_{k \in F} T_{km(i)ti} = d_{ti}$	for t, i	(14)
$\sum_i X_{kti} \leq q_k \cdot \sum_{1 \leq n \leq t} Z_{kn}$	for k, t	(15)
$X_{kti} \geq 0$, $T_{kjti} \geq 0$, and Z_{kt} = GIN	for k, j, t, i	(16)

Problem (P1) is a network model with fixed costs. A node represents a plant where production activity occurs and has production costs and fixed costs. An arc represents a flow of a product from a plant at a stage to a plant at the next stage, and has transfer costs. For each node, constraints (11) and (12) are the outgoing conservation flow equations and constraints (13) and (14) are the incoming conservation flow equations and constraint (15) is the capacity / forcing constraint.

2. 1. 3. The Assumption of Production-System-Only (PSO)

Note that the node represents only a production system (plant), but not a distribution system (warehouse). This assumption of Production-System-Only (PSO) has the following properties, which is used critically in decomposing the problem.

(Property 2.1) A product has to be directly shipped from a node at a stage to a node at its immediate successor. Hence no node can serve as a transshipment node.

(Property 2.2) From Property 2.1, a product visits only one node per stage. Hence, a product produced at stage s visits $|B(s)|+1$ (including the node at stage s) nodes before reaching its market where $B(s)$ is a set of successors of stage s.

(Property 2.3) Let $Y(s)_{ti}$ be the a sequence of nodes visited by the product produced at stage s for (t, i) and let b be a successor of stage s. Then $Y(s)_{ti} = Y'_{ti} + Y(b)_{ti}$ where Y'_{ti} is the one whose last node is the first node of $Y(b)_{ti}$. For example, $Y(s)_{ti} = \{n^1 \to n^2 \to, \text{--}, \to n^{k-1} \to n^k \to n^{k+1} \to, \text{--}, \to n^m\}$, $Y'_{ti} = \{n^1 \to n^2 \to, \text{--}, \to n^{k-1} \to n^k\}$, and $Y(b)_{ti} = \{n^k \to n^{k+1} \to, \text{--}, \to n^m\}$ where n^1 (n^k) is the node producing a product at stage s (b) and $n^i \in S(n^{i-1})$ for $2 \leq i \leq m$. In words, a product produced at stage s has to be assembled with and travel to the market with a product produced at its successor b.

(Property 2.4) Problem (P1) assumes an assembly structure or a serial structure (which is a typical production structure), but not an arborescent structure (which is a typical distribution structure).

2. 1. 4. Decomposition by Stages and Coupling Variables in Problem (P1)

In the real situation, Problem (P1) is often a large mixed integer programming problem which is difficult to solve directly. A very popular, if not standard, way to attack such a large problem is to decompose the problem into smaller subproblems and to integrate their solutions into a global solution. One could think of decomposing Problem (P1) along any of its four dimensions: stages, locations, time periods, and items. In this study we choose to decompose the problem by stages. However, Problem (P1) itself is not appropriate to be decomposed by the stages since its transfer variables, T_{kjti}, are the coupling variables over the stages. These coupling variables link a stages to other stages not only explicitly but also implicitly. Three alternative formulations are developed in order to eliminate the explicit and implicit linkages in Problem (P1): Problem (P2) and (P3) are for eliminating the explicit linkages, and Problem (P4) is for eliminating the implicit linkages.

2. 2. Modified Formulations; Problem (P2) and (P3)

2. 2. 1. Problem (P2)

The explicit linkage is shown in T_{kjti} where $j \in S(k)$. The explicit linkage is eliminated in two steps; the first step is to delete intra-location transfer variables ($T_{ks^*(k)ti}$) from the model by substitution and the second step is to delete the remaining T_{kjti} where $j \in S'(k)$ by a notational change.

(1) Deleting intra-location transfer variables

In Problem (P1), $T_{ks^*(k)ti}$ where $k \notin F$ and $T_{km(i)ti}$ where $k \in F1(i)$ are intra-location transfer variables. By definition, $S(k) = s^*(k) + S'(k)$. Hence constraint (11) can be expressed as follows;

$$\mathbf{X}_{kti} = \mathbf{T}_{ks^*(k)ti} + \textstyle\sum_{j \in S'(k)} \mathbf{T}_{kjti} \qquad \text{for } k \notin F, t, i$$

Or

$$\mathbf{T}_{ks^*(k)ti} = \mathbf{X}_{kti} - \textstyle\sum_{j \in S'(k)} \mathbf{T}_{kjti} \qquad \text{for } k \notin F, t, i \qquad (11')$$

$T_{ks^*(k)ti}$ can be deleted from the model by adding (11′) to (13), which gives new conservation flow equations;

$$\mathbf{X}_{kti} - \textstyle\sum_{j \in S'(k)} \mathbf{T}_{kjti} = \mathbf{r}_{ki} \cdot \mathbf{X}_{s^*(k)ti} - \textstyle\sum_{j \in J(k)} \mathbf{T}_{js^*(k)ti} \qquad \text{for } k \notin F, t, i \qquad (21')$$

From constraint (11′) and the non-negativity of $T_{ks^*(k)ti}$,

$$\mathbf{X}_{kti} - \textstyle\sum_{j \in S'(k)} \mathbf{T}_{kjti} \geq 0 \qquad \text{for } k \notin F, t, i \qquad (20')$$

Again, $F = F1(i) + F2(i)$ for each i. Constraint (12) and (14) can be expressed as follows:

$$\mathbf{X}_{kti} = \mathbf{T}_{km(i)ti} \qquad \text{for } k \in F1(i), t, i \qquad (12')$$

$$\mathbf{X}_{kti} = \mathbf{T}_{km(i)ti} \qquad \text{for } k \in F2(i), t, i \qquad (12'')$$

$$\mathbf{X}_{km(i)ti} + \textstyle\sum_{j \in F2(i)} \mathbf{T}_{jm(i)ti} = \mathbf{d}_{ti} \qquad \text{for } k \in F1(i), t, i \qquad (14')$$

Similarly, $T_{km(i)ti}$ where $k \in F1(i)$ can be deleted by adding constraint (12′) to constraint (14′), which gives,

$$\mathbf{X}_{kti} + \textstyle\sum_{j \in F2(i)} \mathbf{T}_{jm(i)ti} = \mathbf{d}_{ti} \qquad \text{for } k \in F1(i), t, i \qquad (22')$$

From constraint (12″),

$$\mathbf{X}_{kti} - \mathbf{T}_{km(i)ti} = 0 \qquad \text{for } k \in F2(i), t, i \qquad (23')$$

Substituting constraint (11′) and (12′) into the objective function gives new cost parameters.

$b_{kti}=c_{kti}+u_{ks^*(k)ti}$ and $v_{kjti}=u_{kjti}-u_{ks^*(k)ti}$ for k, t, i.

Hence $v_{ks^*(k)ti}=0$, which means that intra-location transfer variables are deleted from the objective function.

A physical interpretation of the above step is as follows: The transfer occurs in two directions: to the next stage or to any other location. Since a product always has to be shipped to the next stage regardless of locations, the intra-location transfer cost to the next stage ($u_{ks^*(k)ti}$) can be considered as a part of the production cost. As a result, the new transfer cost (v_{kjti}) no longer includes the transfer cost to the next stage and represents only the transfer cost to any other location, which leads to the next step.

(2) Notational changes for T_{kjti} where $j\in S'(k)$

T_{kjti} where $j\in S'(k)$ can be replaced by $T_{kj'ti}$, where $j'\in p^*(j)\cap J(k)$, without a loss of generality since (a) there is one-to-one correspondence between them, (b) $v_{kjti}=v_{kj'ti}$, and (c) $v_{jj'ti}=0$ where $j\in s^*(j')$. That is to say, a receiving node can be replaced by its local immediate predecessor that is in the same stage as its sending node. The receiving node is represented by the second subscript in the transfer variable. Hence, the second subscript is modified as follows:

$\sum_{j\in S'(k)} T_{kjti} \rightarrow \sum_{j\in J(k)} T_{kjti}$ in (21′) and (20′),

$\sum_{j\in J(k)} T_{js^*(k)ti} \rightarrow \sum_{j\in J(k)} T_{jkti}$ in (21′),

$\sum_{j\in F2(i)} T_{jm(i)ti} \rightarrow \sum_{j\in F2(i)} T_{jkti} \rightarrow \sum_{j\in J(k)} T_{jkti}$ where $k\in F1(i)$ in (22′),

$T_{km(i)ti} \rightarrow T_{kjti}$ wher $j\in F1(i)$, $k\in F2(i)$ in (23′).

The above steps give Problem (P2).

Problem (P2)

$$\text{MIN} \quad \sum_k\sum_t\sum_i (b_{kti}\cdot X_{kti}+\sum_{j\in J(k)} v_{kjti}\cdot T_{kjti})+\sum_k\sum_t f_{kt}\cdot Z_{kt}$$

s. t.

$$X_{kti} - \sum_{j\in J(k)} T_{kjti} \geq 0 \qquad \text{for } k\notin F,\ t,\ i \qquad (20)$$

$$X_{kti} + \sum_{j\in J(k)} [T_{jkti} - T_{kjti}] = r_{ki}\cdot X_{s^*(k)ti} \qquad \text{for } k\notin F,\ t,\ i \qquad (21)$$

$$X_{kti} + \sum_{j\in J(k)} T_{jkti} = d_{ti} \qquad \text{for } k\in F1(i),\ t,\ i \qquad (22)$$

$$X_{kti} - T_{kjti} = 0 \qquad \text{for } k\in F2(i),\ j\in F1(i) t,\ i, \qquad (23)$$

$$\sum_i X_{kti} \leq q_k\cdot \sum_{1\leq n\leq t} Z_{kn} \qquad \text{for } k,\ t \qquad (24)$$

$$X_{kti} \geq 0,\ T_{kjti} \geq 0 \text{ and } Z_{kt} = \text{GIN} \qquad \text{for } k,\ j,\ t,\ i \qquad (25)$$

where $b_{kti}=c_{kti}+u_{ks^*(k)ti}$ *and* $v_{kjti}=u_{kjti}-u_{ks^*(k)ti}$.

2. 2. 2. Two Assumptions on the Cost Structure and Problem (P2).

Consider the following two assumptions:

(Assumption One) All transfer costs in Problem (P2) are positive ($v_{kjti} > 0$).

(Assumption Two) All transfer costs in Problem (P2) satisfy the law of triangle inequality ($v_{kjti} < v_{klti} + v_{ljti}$).

Proposition 2.1. When all transfer costs in Problem (P2) satisfy both assumptions, constraint (20) is redundant.

〈Proof〉 First consider the following two lemmas.

[Lemma 1] If $t_{kjti} > 0$ for all k, j∈J(k), t, i, then $T^*_{kjti} \cdot T^*_{jkti} = 0$ for k, j∈J(k), t, i, where T^*_{kjti} and T^*_{jkti} are part of the optimal solution of Problem (P2).

〈Proof〉 Suppose that $T^*_{kjti} \cdot T^*_{jkti} > 0$. Let T^*_{kjti} =a and T^*_{jkti} = b and e = Min {a, b}. And let T'_{kjti} = a−e and T'_{jkti} = b−e. Hence $T'_{kjti} \cdot T'_{jkti} = 0$. If we replace T^*_{kjti} and T^*_{jkti} with T'_{kjti} and T'_{jkti}, respectively, these new solutions T'_{kjti} and T'_{jkti} still satisfy the conservation flow equation (21) and hence are feasible. Moreover, it has a smaller objective value by the amount, $e \cdot (v_{kjti}+v_{jkti})$. This contradicts the assumption that T^*_{kjti} and T^*_{jkti} are part of the optimal solution. Hence $T^*_{kjti} \cdot T^*_{jkti} = 0$. Q. E. D.

[Lemma 2] If the law of triangle inequality holds for transfer costs (that is, $v_{kjti} < v_{klti}+v_{ljti}$) then $T^*_{klti} \cdot T^*_{ljti} = 0$ where (l, j)∈J(k).

〈Proof〉 Suppose $T^*_{klti} \cdot T^*_{ljti} > 0$ and $T^*_{klti} \leq T^*_{ljti}$. Let $T'_{ljti} = T^*_{ljti} - T^*_{klti}$ and $T'_{kjti} = T^*_{klti}$, and $T'_{klti} = 0$. The new variables satisfy the conservation flow equation (21) and hence are feasible. The transfer cost of the original solution is $v_{klti} \cdot T^*_{klti} + v_{ljti} \cdot T^*_{ljti} = v_{ljti} \cdot (T^*_{ljti} - T^*_{klti}) + (v_{klti} + v_{ljti}) \cdot T^*_{klti} = v_{ljti} \cdot T'_{ljti} + (v_{klti} + v_{ljti}) \cdot T'_{kjti}$. But, from the law of triangular inequality, $v_{kjti} < v_{klti}+v_{ljti}$. Thus, $v_{ljti} \cdot T'_{ljti}+(v_{klti}+v_{ljti}) \cdot T'_{kjti} > v_{ljti} \cdot T'_{ljti}+v_{kjti} \cdot T'_{kjti}$. This contradicts the fact that T^*_{klti} and T^*_{ljti} are part of the optimal solution. Hence $T^*_{klti} \cdot T^*_{ljti}$ =0. It is also true when $T^*_{ljti} \leq T^*_{klti}$. Q. E. D.

From the above two lemmas, if $T^*_{kjti}>0$ where j∈ J(k), then $T^*_{l'kti}= 0$

where $l' \in J(k)$ and $T^*_{jl''ti} = 0$ where $l'' \in J(j)$. In words, for given (t, i), a sender (k) cannot receive from any plant ($T^*_{l'kti} = 0$ where $l' \in J(k)$) and a receiver (j) cannot send to any plant ($T^*_{jl''ti} = 0$ where $l'' \in J(j)$. Hence, in the optimal solution, a plant can be either a sender or a receiver, but not both at the same time for a given (t,i).

Now let us prove the proposition. Constraint (20) and constraint (21) are,

$X^*_{kti} - \sum_{j \in J(k)} T^*_{kjti} \geq 0$ for $k \notin F$, t, i (20)

$X^*_{kti} + \sum_{j \in J(k)} [T^*_{jkti} - T^*_{kjti}] = r_{ki} \cdot X^*_{s^*(k)ti}$ for $k \notin F$, t, i (21)

where X^*_{kti} and T^*_{kjti} are part of the optimal solution of Problem (P2).

We will prove that constraint (20) is implied by constraint (21) and the non-negativity constraint of X_{kti}. From the above lemmas, plant k can be categorized as either a sender or a receiver. When plant k is a sender, $\sum_{j \in J(k)} T^*_{jkti} = 0$ in constraint (21). Hence constraint (21) becomes $X^*_{kti} - \sum_{j \in (k)} T^*_{kjti} = r_{ki} \cdot X^*_{s^*(k)ti} \geq 0$, which is constraint (20). When plant k is a receiver, $\sum_{j \in J(k)} T^*_{kjti} = 0$ in constraint (20). Hence constraint (20) becomes, $X^*_{kti} \geq 0$, which is its non-negativity constraint. Therefore, for the optimal solution, constraint (20) is implied by constraint (21) and the non-negativity constraint and hence is redundant.

[Note] The assumption of $v_{kjti} > 0$ in Problem (P2) is equivalent to the assumption of $u_{ks^*(k)ti} < u_{kjti}$ whenever $j \in S'(k)$ in Problem (P1), i.e., transfer costs within a location must be less than the transfer costs between locations. Henceforth, Problem (P2) without constraint (20) is denoted as Problem (P3).

Problem (P3)

$\text{MIN} \sum_k \sum_t \sum_i (b_{kti} \cdot X_{kti} + \sum_{j \in J(k)} v_{kjti} \cdot T_{kjti}) + \sum_k \sum_t f_{kt} \cdot Z_{kt}$

s.t.

$X_{kti} + \sum_{j \in J(k)} [T_{jkti} - T_{kjti}] = r_{ki} \cdot X_{s^*(k)ti}$ for $k \notin F$,t,i (21)

$X_{kti} + \sum_{j \in J(k)} T_{jkti} = d_{ti}$ for $k \in F1(i)$, t, i (22)

$X_{kti} - T_{kjti} = 0$ for $k \in F2(i), j \in F1(i)$,t,i (23)

$\sum_i X_{kti} \leq q_k \cdot \sum_{1 \leq n \leq t} Z_{kn}$ for k,t (24)

$X_{kti} \geq 0$, $T_{kjti} \geq 0$ and $Z_{kt} = \text{GIN}$ for k,j,t,i (25)

Proposition 2.2. Problem (P3) still maintains the assumption of PSO by the two assumptions.

⟨Proof⟩ The optimal solution of Problem (P3) has only direct transfers (no transsshipment nodes) due to the two assumptions. This is a property of the assumption of PSO (Property 2.1) Q. E. D.

2.3. An Echelon Model : Problem (P4)

Variable T_{kjti} in Problem (P3) still implicity links every stage to all its previous stages. A new variable, "location transfer," is introduced to remove the implicit linkage.

2.3.1. Location Transfer, LT_{kjti}.

Location transfer is defined as the total amount of a product transferred between two locations regardless of the stages (as product itself or as part of an assembly). This is basically the same concept as the echelon inventory.

Formally, LT_{kjti} can be defined as follows. Let R(k) be a set of plants that are located in the same location as plant k and form a path from plant k to the last plant, or $R(k)=\{k, k^1, k^2, \cdots k^{f-1}, k^f\}$ where $k^1=s^*(k)$, $k^2=s^*(k^1)$, $\cdots$, $k^f=s^*(k^{f-1})$ and the last plant. And let $R(k,u)=R(k)-R(u)=(k, k^1, k^2, \cdots u-1\}$ where $k^1=s^*(k)$,$k^2=s^*(k^1)$, $\cdots$, $u-1=p^*(u)$ and $u \in R(k)$. Let $\alpha_{kui}= \prod_{l \in R(k,u)} r_{li}$ where $\alpha_{kki}=1$. In words, α_{kui} is the number of units of the product produced at plant k that are required for one unit of product produced at plant u for item i.

Then $LT_{kjti} = \sum_{u \in R(k),\, v \in R(j) \cap J(u)} \alpha_{kui} \cdot T_{uvti}$ (21a)

And its cost, $w_{kjti}=v_{kjti}-\sum_{p \in p^*(k), q \in p^*(j) \cap J(p)} r_{pi} \cdot v_{pqti}$ (22a)

Or from (21a) and (22a)

$T_{kjti}=LT_{kjti}-r_{ki} \cdot LT_{s^*(k)s^*(j)ti}$ (23a)

$v_{kjti}=w_{kjti}+\sum_{p \in (k),\, q \in p(j) \cap J(p)} \alpha_{pki} \cdot w_{pqti}$ (24a)

Equation (21a) and (23a) show that T_{kjti} and LT_{kjti} have a one-to-one linear transformation relationship.

2.3.2. Problem (P4)

Substituting equation (23a) into Problem (P3) and solving for X_{kti} in the conservation flow equations give the following Problem (P4)

Problem (P4)

$$\text{MIN} \sum_k\sum_t\sum_i (b_{kti} \cdot X_{kti}+\sum_{j\in J(k)} w_{kjti}\, LT_{kjti})+\sum_k\sum_t f_{kt} \cdot Z_{kt}$$

s.t.

$$r_{ki} \cdot LT_{s^*(k)s^*(j)ti}-LT_{kjti} \le 0 \qquad \text{for } k, t, i \qquad (30)$$

$$X_{kti}+\sum_{j\in J(k)} [LT_{jkti}-LT_{kjti}] = r^*_{ki}d_{ti} \qquad \text{for } k\in K(i), t, i \qquad (31)$$

$$X_{kti}+ \sum_{j\in J(k)} [LT_{jkti}-LT_{kjti}]= 0 \qquad \text{for } k \notin K(i), t, i \qquad (32)$$

$$\sum_i X_{kti} \le q_k \cdot \sum_{1\le n\le t} Z_{kn} \qquad \text{for } k, t \qquad (33)$$

$$X_{kjti} \ge 0, LT_{kjti} \ge 0 \text{ and } Z_{kt} = \text{GIN} \qquad \text{for } k, j, t,i \qquad (34)$$

Proposition 2.3. Problem (P3) and Problem (P4) are equivalent.

〈Proof〉

Since the location transfer is basically the same concept as the echelon inventory, the proof by Afentakis et al. [1] can be used to prove this proposition. It is thus not repeated here. Constraint (30) comes from constraint (23a) and the non-negativity constraint. Q. E. D.

LT_{kjti} is basically an echelon concept. However, unlike the echelon inventory, it does not require the assumption that the (transfer) cost at each stage has to be greater than the sum of the costs of all the predecessors. It is therefore possible to have negative w_{kjti}. However, negative w_{kjti} do not hurt the solution procedure since they are eliminated in the subproblem as shown later.

Problem (P4) still maintains the assumption of PSO by having $[w_{kjti}+\sum_{p\in P(k),q\in P(j)\cap J(p)} \alpha_{pki} \cdot w_{pqti}] <$

$$[w_{klti} + \sum_{p\in P(k),q'\in P(l)\cap J(p)} \alpha_{pki} \cdot w_{pq'ti}]+[w_{ljti}+\sum_{p'\in P(l),q\in P(j)\cap J(p')} \alpha_{p'ki} \cdot w_{p'qti}] \cdots\cdots (25a)$$

where j, $l\in J(k)$ and P(·) is a set of all local predecessors of plant(·) from the assumption of the law of triangle inequality over v_{kjti} in Problem (P3) and constraint (24a). Not that this expression links a stage to its predecessors.

3. Subproblems

3.1 The Subproblem for Problem (P4) :Problem (SP4)

It can be easily seen that Problem (P4) consists of several single-stage problems held together by coupling constraint (30). Thus, Problem (P4) can be decomposed into single-stage problems by relaxing constraint (30). The resulting subproblem (SP4) is:

Let s = index of stages and L(s)=a set of plants in stage s.

Problem (SP4) For each stage s,

$$\text{MIN} \sum_{k\in L(s)}\sum_{t}\sum_{i} [b_{kti} \cdot X_{kti} + \sum_{j\in J(k)} w_{kjti} \cdot LT_{kjti}] + \sum_{k\in L(s)}\sum_{t} f_{kt} \cdot Z_{kt}$$

s.t

$$X_{kti} + \sum_{j\in J(k)} [LT_{jkti} - LT_{kjti}] = r^{*}_{ki}\, d_{ti} \qquad \text{for } k\in L(s)\cap K(i), t, i$$

$$X_{kti} + \sum_{j\in J(k)} [LT_{jkti} - LT_{kjti}] = 0 \qquad \text{for } k\in L(s) \text{ but} \notin K(i), t, i$$

$$\sum_{i} X_{kti} \le q_{k} \cdot \sum_{1\le n\le t} Z_{kn} \qquad \text{for } k \in L(s), t$$

$$X_{kti}, LT_{kjti} \ge 0 \text{ and } Z_{kt} = \text{GIN} \qquad \text{for } k \in L(s), t, i$$

A drawback of Problem (SP4) is that it no longer maintains the assumption of PSO. From constraint (25a) in Section 2.3, we know that Problem (P4) maintains the assumption of PSO by forcing the law of triangle inequality over its transfer costs.

$$[w_{kjti}+\sum_{p\in P(k),q\in P(j)\cap J(p)} \alpha_{pki} \cdot w_{pqti}] < [w_{klti}+ \sum_{p\in P(k),q'\in P(l)\cap J(p)} \alpha_{pki} \cdot w_{pq'ti}]$$
$$+[w_{ljti}+\sum_{p'\in P(l),q\in P(j)\cap J(p')} \alpha_{p'qki} \cdot w_{p'qti}] \cdots\cdots (25a)$$

where $j,l\in J(k)$ and $P(\cdot)$ is a set of all local predecessors of plant $(\cdot)$.

Note that the expression in the constraint links a stage to its predecessors. Decomposing Problem (P4) by the stages destroys the law of triangle inequality since each single-stage subproblem considers only its own transfer costs. In other words, the assumption of PSO is relaxed in the process of the decomposition. In this case, the solutions of subproblems can still be infeasible to Problem (P4) even when they satisfy the coupling constraint (30). Morever, the sum of subproblems gives a very poor lower bound for Problem (P4) when a subproblem(s) has a negative cycle(s) of transfer costs, w_{kjti}. Hence we do need a transformation from Problem (SP4) to a subproblem that maintains the assumption of PSO.

3.2. A Path-Based Formulation: Problem(SP).

In order to have the subproblem maintain the assumption of PSO, we transform Problem (SP4), which is an arc-based formulation, into a path-based formulation. This transformation is possible because of the following two properties of Problem (SP4).

(Property 3.1) A set of transfers (arcs) in a feasible solution of Problem (SP4) always forms a path from the source to the destination where a path is defined as a sequene of locations and the source is the location where a product is produced and the destination is the location where the product is consumed.

This property is based on the fact that a set of transfers that does not form a path violates the conservation flow equations in Problem (SP4). This property is also true for Problem (P4).

(Property 3.2) The entire set of transfer variabls can be deleted form Problem (SP4).

This property is based on the fact that Problem (SP4) has no constraints over its transfer variables. Hence, with property 3.1, we are free to choose a set of transfers that consists of the best path from the source to the destination for each (t,i). Transfer variables can be deleted (a) by adding the costs of the transfer variables consisting of the best paths to the production cost and (b) by ignoring all other transfer variables for each (t,i). Hence the transfer costs of the best path is considered as a part of the production cost. This property is not true for Problem (P4) since constraint (30) is a capacity constraint over transfer variables as well as a coupling constraint over the stages.

The resulting path-based formulation, Problem (SP), is:

Subproblem(SP)

$$\text{MIN} \sum_{k\in L(s)}\sum_t\sum_i a_{kti}\cdot X_{kti} + \sum_{k\in L(s)}\sum_t f_{kt}\cdot Z_{kt}$$

s.t

$\sum_{k\in L(s)} X_{kti}=D_{ti}$ for t, i

$\sum_i X_{kti} \leq q_k \cdot \sum_{1\leq n\leq t} Z_{kn}$ for $k \in L(s)$, t

$X_{kti} \geq 0$, $Z_{kt} = \text{GIN}$ for $k\in L(s)$, t, i

where $a_{kti}=b_{kti}$+transfer costs for the best path from source k to the destination.

$D_{ti} = r^*_{ki} \cdot d_{ti}$.

Finding the best paths during the transformation is equivalent to solving the shortest path problem. We use the Bellman-Ford algorithm, which solves the following recursive equation: $U^k(\eta,\delta)_{ti}= \min_j\{U^j(\eta-1,\delta)_{ti}+w_{jkti}\}$ where η is the bound on the number of arcs in the path and δ is the destination and $U^k(\eta,\delta)_{ti}=$ the cost of the shortest path from node k to the destination (δ) for (t, i) subject to the condition that the path contains no more than η arcs.

The following proposition says that the assumption of PSO determines the value of η. Hence Problem (SP) can maintain the assumption of PSO in solving the shortest path problem with the proper value of η. Without the assumption of PSO, η is set to ∞, which produces a negatively infinite lower bound with negative cycles.

First, the proposition needs the following notation.

〈Notation〉

s=the current stage and $s \in S$;

a(b)=a solved predecessor (successor) of stage s;

$P(\cdot)_{ti}$=a path at stage(·) for (t,i), which is a sequence of locations visited by one unit of the product manufactured at stage (·);

$NP(\cdot)_{ti}= |P(\cdot)_{ti}|-1$, which is the number of arcs in $P(\cdot)_{ti}$ where an arc represents a transfer between two locations;

$\mu(\cdot)$ = the number of successors of stage (·) plus one, $|B(\cdot)+1|$;

<u>Proposition 3.1.</u>

The assumption of PSO is satisfied if and only if following (31a) is satisfied:

$\max\{0, NP(a)_{ti}-\mu(a)+\mu(s), NP(b)_{ti}\} \le NP(s)_{ti} \le \min\{\mu(s), NP(a)_{ti}, NP(b)_{ti} -\mu(b)+\mu(s)\}$ for t, i, and $s \in S$ ······(31a)

〈Proof〉

An arc is defined as a flow between two locations and plants can be located in the same location. Hence, from property 2.2 and property 2.3, the assumption of PSO has $NP(s)_{ti}-NP(s')_{ti} \le 1$ ······(31b)

where s′ is the immediate successor of stage s and $NP(s')_{ti}=0$ when stage s is the last stage. The proposition is proved by using constraint (31b).

(i) First, we prove that constraint (31b) implies constraint (31a). Consider a set of stages S′ such that $S'=\{s^1, s^2, \cdots, s^{j-1}, s^j, \cdots ,s^J\}$ where stage s^1 is the last stage and stage s^j is an immediate predecessor of stage s^{j-1}. From constraint (31b), we have

$NP(s^j)_{ti} \leq NP(S^{j-1})_{ti} + 1 \leq NP(s^{j-2})_{ti} + 2 \leq \cdots \leq NP(s^{j-r})_{ti} + r = NP(s^{j-r})_{ti} + \mu(s^j) - \mu(s^{j-r})$, which is, $0 \leq NP(s^j)_{ti} - NP(s^{j-r})_{ti} \leq \mu(s^j) - \mu(s^{j-r})$ where $1 \leq r \leq j-1$ (32b)

With the original notation {s,a,b}, constraint (32b) becomes:

$0 \leq NP(s)_{ti} - NP(b)_{ti} \leq \mu(s) - \mu(b)$ (33b)

$0 \leq NP(s)_{ti} \leq \mu(s)$ when b=the market (34b)

$0 \leq NP(a)_{ti} - NP(s)_{ti} \leq \mu(a) - \mu(s)$ (35b)

With $NP(s)_{ti} \geq 0$, constraint (33b), (34b) and (35b) reduce to constraint (31a).

(ii) Next, we prove that constraint (31a) implies constraint (31b). We only need to prove that constraint (32b) implies constraint (31b). By setting s^j = s and s^{j-1}=s′, $NP(s)_{ti} - NP(s')_{ti} \leq \mu(s) - \mu(s') = 1$ Q.E.D.

[Note] If a = ϕ (b=ϕ) then, simply discard the expressions containing stage a (stage b) in constraint (31a).

For each independent stage s (subproblem), we have a=b=ϕ and hence $0 \leq NP(s)_{ti} \leq \mu(s)$. Hence Problem (SP) for stage s maintains the assumption of PSO by setting $\eta = \mu(s)$ in its shortest paths. Naturally, it also gives a better lower bound even with negative cycles.

4. Conclusions and Extensions

We suggets a procedure of decomposing a multi-stage dynamic location problem with respect to stages. It turns out that the dimensions of item and time period have no role in the process of decomposition. It also turns out that the property of the integer variables Z_{kt} has no effect on the decomposition procedure. The resulting subproblem (SP) is a typical single-stage dynamic location problem. Many efficient solution methods for this problem have been developed, which satisfies a critical condition for a successful decomposition, that is: the subprobem should be easy to solve.

In order to finish the job, we have to develop a procedure that integrates the solutions of the subproblems to find a final of problem (P) and to do some empirical work that proves its efficiency.

Referenes

[1] Afentakis, P., B. Gavish and U. Karmarkar, "Computational Efficient Optimal Solutions to the Lot-sizing Problem in Multistage Assembly System," *Management Science*, Vol. 30, No. 2 (1984), pp. 222-239.

[2] Clark, A. and H. Scarf, "Optimal Policies for Multi-Echelon Inventory Problems," *Management Science*, Vol. 23 (1960), pp. 475-490

[3] Erlenkotter, D., "A comparative Study of Approaches to Dynamic Location Problems," *European Journal of Operational Research*, Vol. 6 (1981), pp. 133-143.

[4] Fong, C. O., and V. Srinivasan, "The Multiregion Dynamic Capacity Expansion Problem, Part 1 and 2," *Operations Research*, Vol. 29 (1981), pp. 787-816.

[5] Fong, C. O., and V. Srinivasan, "The Multiregion Dynamic Capacity Expansion Problem, Improved Heuristic," *Management Science*, Vol. 32 (1986), pp. 1140-1152.

[6] Geoffrion, A. M. and G. W. Graves, "Multicommodity Distribution System Design by Benders Decomposition," *Management Science*, Vol. 20, No. 5 (1974), pp. 822-844.

[7] Schrage, L., *LINDO, 3RD ED.*, The Scientific Press, New York, 1986.

[8] Shulman, A., "An Algorithm for Solving Dynamic Capacitated Plant Location Problems with Discrete Expansion Sizes," *Operations Research*, Vol. 39, No. 3 (1991), pp. 423-436.

[9] Van Roy, T. J., and D. Erlenkotter, "A Dual Based Procedure for Dynamic Facility Location," *Management Science*, Vol. 28 (1982), pp. 1091-1105.