# A direct search algorithm for solving the multi-period single-sourcing problem

Miduk Tampubolon,Vita Damayanti,Herman Mawengkang

## Abstract

Generally, problems in logistics faced by a supplier would be the production timing, the location of inventories, and the assignment of customers to warehouses. This paper will consider a dynamic multi-period single-sourcing problem (MPSSP) used to support the corresponding decisions. We propose a direct search algorithm for solving the multi-period single-sourcing problems. In partiular, we generalize the strategy of releasing nonbasic variables from their bounds, combined with the active constraint method that was developed for the Generalized Assignment Problem (GAP) to a class of convex assignment problems. We then identify an important subclass of problems, containing many variants of the multi-period single-sourcing problem (MPSSP), as well as variants of the GAP.

Key words : Logistics, multi period single sourcing, Integer programming

## 1. Introduction

Some of the most important problems in logistics faced by a supplier are the timing of production, the location of inventories, and the assignment of customers to warehouses. In this paper we will study a multi-period single-sourcing problem (MPSSP) that can be used to support the corresponding decisions. The model we propose is dynamic in nature, in contrast to many of the quantitative models proposed in the literature which assume a static environment. The fact that our model is dynamic enables us to handle a dynamic demand pattern of the customers, as well as to support inventory decisions explicitly. Related literature, focusing on static models, can be found in Geoffrion and Graves [10], Benders et al. [2], and Fleischmann [9]. Duran [7] studies a dynamic model for the planning of production, bottling, and distribution of beer, but focuses on the production, instead of the distribution, process. Chan, Muriel and Simchi-Levi [4] study a dynamic, but uncapacitated, distribution problem.

The logistics network we are considering consists of a set of facilities (each of which could be interpreted as a plant with an associated warehouse), and a set of customers. The decisions that need to be made concern (i) the assignment of customers to facilities, and (ii) the location and size of inventories. These two types of decisions can be handled in a nested fashion, where we essentially decide on the assignment of customers to facilities only, and where the location and size of inventories are determined optimally as a function of the customer assignments. Viewed in this way, the multi-period single-sourcing problem is a generalized assignment problem with a convex objective function and possibly additional constraints, representing, for example, throughput or physical inventory capacities, or perishability constraints. To be able to deal with many variants of the multi-period single-sourcing problem using a single solution approach, we will introduce a general class of convex assignment problems, having the property that the objective function and feasible region are convex, and are both separable in the facilities. The class of convex assignment problems clearly contains the well-known Generalized Assignment Problem (GAP), and thus convex assignment problems are NΠ-Hard as well. We will discuss one of the variants of the multi-period single-sourcing problem in detail in this paper. In this variant each plant has known, finite, and possibly time-varying, capacity, each customer needs to be served by (assigned to) a unique facility throughout the planning horizon, and the customer demands exhibit a seasonal pattern.

The outline of the paper is as follows. In Section 2 we will introduce a class of convex assignment problems, CAP, and propose a direct search approach for solving these problems based on the strategy of releasing nonbasic variables from their bounds, combined with the active constraint method. The direct search approach will be given in Section 3. In Section 4 we formulate the variant of the multi-period single-sourcing problem mentioned above. In Section 5 we end the paper with some concluding remarks.

## 2. Convex assignment problems

Consider the following convex assignment problem:

$$\text{minimize} \sum_{i=1}^{m} g_i(x_i)$$

subject to (CAP)

$$\sum_{i=1}^{m} x_{ij} = 1 \qquad j = 1, \ldots, n$$

$$x_{ij} \in \{0, 1\} \qquad i = 1, \ldots, m;\ j = 1, \ldots, n$$

$$x_{i\cdot} \in X_i \qquad i = 1, \ldots, m$$

where the functions gi are convex, as are the sets Xi denoting any additional constraints. Ferland, Hertz and Lavoie [8] introduce an even more general class of assignment problems, and show the applicability of object-oriented programming by developing software containing several heuristics. As mentioned in the introduction, the GAP is an example of a convex assignment problem, where the cost function gi and the additional constraints de_ned by the set $X_i$ associated with agent $i$ are linear in $x_{i\cdot}$. Variants of the MPSSP are examples of convex assignment problems as well, one of which will be discussed in detail in Section 4. In a more general context, all set partitioning models discussed by Barnhart et al. [1] with convex and separable objective function in the index i are examples of convex assignment problems. The CAP can be formulated as a set partitioning problem, in a similar way as was done for the GAP by Cattryse, Salomon, and Van Wassenhove [3]; and Savelsbergh [22]. In particular, a feasible solution for (CAP) can be seen as a partition of the set of *objects* $\{1, \ldots, n\}$ into m subsets. Each element of the partition is associated with one of the m agents.

Now let $L_i$ be the number of subsets of objects that can feasibly be assigned to agent $i$ ($i = 1, \ldots, m$). Let $\alpha^{\ell}_{i\cdot}$ denote the $\ell$-th subset (for fixed $i$), i.e., $\alpha^{\ell}_{ij} = 1$ if object j is an element of subset $\ell$ for agent $i$, and $\alpha^{\ell}_{ij} = 0$ otherwise. We will call $\alpha^{\ell}_{i\cdot}$ the $\ell$-th column for agent $i$. Then, the set partitioning problem can be formulated as follows:

$$\text{minimize} \sum_{i=1}^{m}\sum_{\ell=1}^{L_i} g_i(\alpha^{\ell}_{i\cdot})y^{\ell}_i$$

subject to (MP)

$$\sum_{i=1}^{m}\sum_{\ell=1}^{L_i} \alpha^{\ell}_{ij}y^{\ell}_i = 1 \qquad j = 1, \ldots, n \tag{1}$$

$$\sum_{\ell=1}^{L_i} y^{\ell}_i = 1 \qquad i = 1, \ldots, m \tag{2}$$

$$y^{\ell}_i \in \{0,1\} \qquad \ell = 1, \ldots, L_i;\ i = 1, \ldots, m$$

where $y^{\ell}_i$ is equal to 1 if column $\ell$ is chosen for agent $i$, and 0 otherwise. As mentioned by Barnhart et al. [1], the convexity constraint (2) for agent $i$ ($i = 1, \ldots, m$) can be written as

$$\sum_{\ell=1}^{L_i} y^{\ell}_i \le 1$$

if $\alpha_{ij} = 0$ for each $j = 1, \ldots, n$ is a feasible column for agent i with associated costs $g_i(\alpha_{i\cdot}) = 0$. One of the advantages of (MP) is that its linear relaxation LP(MP) gives a bound on the optimal solution value of (MP) that is at least as tight (and usually tighter) as the one obtained by relaxing the integrality constraints in (CAP), $R$(CAP). Hence, if we let $v$(R(CAP)) and $v$(LP(MP)) denote the optimal objective values of $R$(CAP) and LP(MP), respectively, then the following holds.

Proposition 2.1 *The following inequality holds*:

$$v(R(CAP)) \le v(LP(MP)).$$

Proof : First of all, note that if LP(MP) is infeasible, the inequality follows directly since in that case $v(\text{LP(MP)}) = \infty$. In the more interesting case that LP(MP) is feasible, the desired inequality follows from the convexity of the objective function and the feasible region of (CAP). We may observe that both relaxations can be obtained by relaxing the integrality constraints to nonnegativity constraints. Each feasible solution to LP(MP) can be transformed to a feasible solution to R(CAP) as follows:

$$x_{ij} = \sum_{\ell=1}^{L_i} \alpha^{\ell}_{ij}y^{\ell}_i \qquad i = 1, \ldots, m;\ j = 1, \ldots, n.$$

For each $i = 1, \ldots, m$, vector $x_{i\cdot}$ is a convex combination of vectors $\alpha^{\ell}_{i}$ for $\ell = 1, \ldots, L_i$. Since all constraints in (CAP) are convex x is a feasible solution for (CAP). Moreover, by convexity of the functions $g_i$ we have that

$$\sum_{i=1}^{m} g_i(x_i) = \sum_{i=1}^{m} g_i\left(\sum_{\ell=1}^{L_i} \alpha^{\ell}_{i\cdot}y^{\ell}_i\right) \le \sum_{i=1}^{m}\sum_{\ell=1}^{L_i} g_i(\alpha^{\ell}_{i\cdot})y^{\ell}_i.$$

Thus, the desired inequality follows. □

This result suggests that the formulation (MP) is more promising than (CAP) when solving the convex assignment problem.

2.1 Solving the convex assignment problem

The convex assignment problem is a (non-linear) Integer Programming Problem which can be solved to optimality by using, for example, a Branch and Bound algorithm. One of the factors determining the performance of this algorithm is the quality of the lower bounds used to fathom nodes. Proposition 2.1 shows that the lower bound given by relaxing the integrality constraints in (MP) is at least as good as the one obtained by relaxing the integrality constraints in (CAP). Thus, the set partitioning formulation for the convex assignment problem looks more attractive when choosing a Branch and Bound scheme. There are other reasons to opt for this formulation like the possibility of adding constraints that are difficult to express analytically.

A standard Branch and Bound scheme would require all the columns to be available, but (in the worst case) the number of columns (and thus the number of variables) of (MP) can be exponential in the size of the problem. This makes a standard Branch and Bound scheme quite unattractive for (MP).

3. The Basic Approach for Direct search

Consider a MILP problem with the following form

$$\text{Minimize} \quad P = c^T x \tag{5}$$

$$\text{Subject to} \quad Ax \leq b \tag{6}$$

$$x \geq 0 \tag{7}$$

$$x_j \text{ integer for some } j \in J \tag{8}$$

A component of the optimal basic feasible vector $(x_B)_k$, to MILP solved as continuous can be written as

$$(x_B)_k = \beta_k - \alpha_{k1}(x_N)_1 - \cdots - \alpha_{kj}(x_N)_j - \cdots - \alpha_{kn-m}(x_N)_{n-m} \tag{9}$$

Note that, this expression can be found in the final tableau of Simplex procedure. If $(x_B)_k$ is an integer variable and we assume that $\beta_k$ is not an integer, the partitioning of $\beta_k$ into the integer and fractional components is that given

$$\beta_k = [\beta_k] + f_k,\ 0 \leq f_k \leq 1 \tag{10}$$

suppose we wish to increase $(x_B)_k$ to its nearest integer, $([\beta]+1)$. Based on the idea of suboptimal solutions we may elevate a particular nonbasic variable, say $(x_N)_{j^*}$, above its bound of zero, provided $\alpha_{kj^*}$, as one of the element of the vector $\alpha_{j^*}$, is negative. Let $\Delta_{j^*}$ be amount of movement of the non variable $(x_N)_{j*}$, such that the numerical value of scalar $(x_B)_k$ is integer. Referring to Eqn.(9), $\Delta_{j*}$ can then be expressed as

$$\Delta_{j^*} = \frac{1 - f_k}{-\alpha_{kj*}} \tag{11}$$

while the remaining nonbasic stay at zero. It can be seen that after substituting (10) into (11) for $(x_N)_{j*}$ and taking into account the partitioning of $\beta_k$ given in (10), we obtain

$$(x_B)_k = [\beta] + 1$$

Thus, $(x_B)_k$ is now an integer.

It is now clear that a nonbasic variable plays an important role to integerize the corresponding basic variable. Therefore, the following result is necessary in order to confirm that must be a non-integer variable to work with in integerizing process.

**Theorem 1.** Suppose the MILP problem (5)-(8) has an optimal solution, then some of the nonbasic variables. $(x_N)_j$, $j = 1, \ldots, n$, must be non-integer variables.

**Proof.**

Solving problem as a continuous of slack variables (which are non-integer, except in the case of equality constraint). If we assume that the vector of basic variables $x_B$ consists of all the slack variables then all integer variables would be in the nonbasic vector $x_N$ and therefore integer valued.

3.1 Derivation of the method

It is clear that the other components, $(x_B)_{i \neq k}$, of vector $x_B$ will also be affected as the numerical value of the scalar $(x_N)_{j*}$ increases to $\Delta_{j*}$. Consequently, if some element of vector $\alpha_{j*}$, i.e., $\alpha_{j*}$ for $i \neq k$, are positive, then the corresponding element of $x_B$ will decrease, and eventually may pass through zero. However, any component of vector $x$ must not go below zero due to the non-negativity restriction. Therefore, a formula, called the minimum ratio test is needed in order to see what is the maximum movement of the nonbasic $(x_N)_{j*}$ such that all components of $x$ remain feasible. This ratio test would include two cases.

1. A basic variable, $(x_B)_{i \neq k}$ decreases to zero (lower bound) first.
2. The basic variable, $(x_B)_k$ increases to an integer.

Specifically, corresponding to each of these two cases above, one would compute

$$\theta_1 = \min_{i \neq k | \alpha_{j*} > 0} \left\{ \frac{\beta_i}{\alpha_{j*}} \right\} \quad (12)$$

$$\theta_2 = \Delta_{j*} \quad (13)$$

How far one can release the nonbasic $(x_N)_{j*}$ from its bound of zero, such that vector $x$ remains feasible, will depend on the ratio test $\theta^*$ given below

$$\theta^* = \min(\theta_1, \theta_2) \quad (14)$$

obviously, if $\theta^* = \theta_1$, one of the basic variable $(x_B)_{i \neq k}$ will hit the lower bound before $(x_B)_k$ becomes integer. If $\theta^* = \theta_2$, the numerical value of the basic variable $(x_B)_k$ will be integer and feasibility is still maintained. Analogously, we would be able to reduce the numerical value of the basic variable $(x_B)_k$ to its closest integer $[\beta_k]$. In this case the amount of movement of a particular nonbasic variable, $(x_N)_{j*}$, corresponding to any positive element of vector $\alpha_{j'}$, is given by

$$\Delta_{f'} = \frac{f_k}{\alpha_{kj}} \quad (15)$$

In order to maintain the feasibility, the ratio test $\theta^*$ is still needed. Consider the movement of a particular nonbasic variable, $\Delta$, as expressed in Eqns.(11) and (15). The only factor that one needs to calculate is the corresponding element of vector $\alpha$. A vector $\alpha_j$ can be expressed as

$$\alpha_j = B^{-1} a_j, j = 1, \ldots, n - m \quad (16)$$

Therefore, in order to get a particular element of vector $\alpha_j$ we should be able to distinguish the corresponding column of matrix $[B]^{-1}$. Suppose we need the value of element $\alpha_{kj*}$, letting $v_k^T$ be the k-th column vector of $[B]^{-1}$, we then have

$$v_k^T = e_k^T B^{-1} \quad (17)$$

subsequently, the numerical value of $\alpha_{kj*}$ can be obtained from

$$\alpha_{kj*} = v_k^T a_{j*} \quad (18)$$

in Linear Programming (LP) terminology the operation conducted in Eqns. (17) and (18) is called the pricing operation. The vector of reduced costs $d_j$ ca is used to measure the deterioration of the objective function value caused by releasing a nonbasic variable from its bound. Consequently, in deciding which nonbasic should be released in the integerizing process, the vector $d_j$ must be taken into account, such that deterioration is minimized. Recall that the minimum continuous solution provides a lower bound to any integer-feasible solution. Nevertheless, the amount of movement of particular nonbasic variable as given in Eqns. (11) or (15), depends in some way on the corresponding element of vector $\alpha_j$. Therefore it can be observed that the deterioration of the objective function value due to releasing a nonbasic variable $(x_N)_{j*}$ so as to integerize a basic variable $(x_B)_k$ may be measured by the ration

$$\left|\frac{d_k}{\alpha_{kj*}}\right| \tag{19}$$

where $|a|$ means the absolute value of scalar a.

In order to minimize the deteoration of the optimal continuous solution we then use the following strategy for deciding which nonbasic variable may be increased from its bound of zero, that is,

$$\min_j \left\{ \left|\frac{d_k}{\alpha_{kj*}}\right| \right\}, j = 1, \ldots, n - m \tag{20}$$

From the "active constraint" strategy and the partitioning of the constraints corresponding to basic ($B$), superbasic ($S$) and nonbasic ($N$) variables we can write

$$\begin{bmatrix} B & S & N \\ & & I \end{bmatrix} \begin{bmatrix} x_b \\ x_N \\ x_S \end{bmatrix} = \begin{bmatrix} b \\ b_N \end{bmatrix} \tag{21}$$

or

$$Bx_b + Sx_S + Nx_N = b \tag{22}$$

$$x_N = b_N \tag{23}$$

The basis matrix $B$ is assumed to be square and nonsingular, we get

$$x_B = \beta - Wx_S - \alpha\, x_N \tag{24}$$

where

$$\beta = B^{-1}b \tag{25}$$

$$W = B^{-1}S \tag{26}$$

$$\alpha = B^{-1}N \tag{27}$$

Expression (23) indicates that the nonbasic variables are being held equal to their bound. It is evident through the "nearly" basic expression of Eqn. (24), the integerizing strategy discussed in the previous section, designed for MILP problem can be implemented. Particularly, we would be able to release a nonbasic variable from its bound, Eqn (23) and exchange it with a corresponding basic variable in the integerizing process, although the solution would be degenerate. Furthermore, the Theorem (1) above can also be extended for MINLP problem.

**Theorem 2.** Suppose the MINLP problem has a bounded optimal continuous solution, then we can always get a non-integer $y_j$ in the optimum basic variable vector.

***Proof***

1. If these variables are nonbasic, the they will be at their bound. Therefore they have integer value.
2. If a $y_j$ is superbasic, it is possible to make $y_j$ basic and bring in a nonbasic at its bound to replace it in the superbasic.

However, the ratio test expressed in (14) connot be used as a tool to guarantee that the integer solution optimal found gill remains in the feasible region. Instead, we use the feasibility test from Minos in order to check whether the integer solution is feasible or infeasible.

3.2 Pivoting

Currently, we are in a position where particular basic variable, $(x_B)_k$ is being integerized, thereby a corresponding nonbasic variable, $(c_N)_{j*}$, is being released from its bound of zero. Suppose the maximum movement of $(x_N)_{j*}$ satisfies

$$\theta^* = \Delta_{j*}$$

such that $(x_B)_k$ is integer valued to exploit the manner of changing the basis in linear programming, we would be able to move $(x_N)_{j*}$ into $B$ (to replace $(x_B)_k$) and integer-valued $(x_B)_k$ into $S$ in order to maintain the integer solution. We now have a degenerate solution since a basic variable is at its bound. The integerixing process continues with a new set [$B$, $S$]. In this case, eventually we may end up with all of the integer variables being superbasic.

**Theorem 3**. A suboptimal solution exists to the MILP and MINLP problem in which all of the integer variables are superbasic.

***Proof***

1. If all of the integer variables are in $N$, then they will be a bound.
2. If an integer variable is basic it is possible to either
   - interchange it with a superbasic continuous variable, or
   - make this integer variable superbasic and bring in a nonbasic at its bound to replace it in the basis which gives a degenerate solution.

The other case which can happen is that a different basic variabels $(x_B)_{i\neq k}$ may hit its bound before $(x_B)_k$ becomes integer. Or in other words, we are in a situation where

$$\theta^* = \Delta_1$$

In this case we move the basic variable $(x_B)_i$ into $N$ and its position in the basic variable vector would be replaced by nonbasic $(x_N)_{j^*}$. note $(x_B)_k$ is still a non-integer basic variable with a new value.

4. The Multi-Period Single-Sourcing Problem

In this section we will introduce the notation of the MPSSP and we will show that it is a member of the class of convex assignment problems presented in Section 2. Let $n$ denote the number of customers, $m$ the number of facilities, and $T$ the planning horizon. The total demand of customer $j$ throughout the planning horizon is given by $d_j$. The demand patterns over time of the customers are assumed to exhibit a common seasonality, represented by nonnegative seasonal factors $\sigma_t$ for each $t$ = 1, …, $T$, satisfying $\Sigma_{t=1}^{T}\sigma_t = 1$. Thus, the demand of customer $j$ in period $t$ is equal to $\sigma_t d_j$. Let $b_{it}$ denote the production capacity at facility $i$ in period $t$. The costs of supplying customer $j$ by facility i in period t are equal to $c_{ijt}$. The unit inventory holding costs at facility $i$ in period $t$ are given by $h_{it}$. (All parameters are nonnegative by definition.) For convenience, we assume that each warehouse has essentially unlimited physical and throughput capacity. In other words, we assume that its physical capacity is sufficient to be able to store the cumulative excess production of its corresponding plant, even if this plant produces to full capacity in each period. In addition, the throughput capacity is large enough for the warehouse to be able to supply any combination of customers assigned to it. However, these two types of capacity constraints can be easily added at little expense to the algorithm.

The MPSSP can be formulated as follows:

minimize

$$\text{minimize} \sum_{t=1}^{T}\sum_{i=1}^{m}\sum_{j=1}^{n} c_{ijt}x_{ij} + \sum_{t=1}^{T}\sum_{i=1}^{m} h_{it}I_{it}$$

subject to (P0)

$$\sigma_t \cdot \sum_{j=1}^{n} d_j x_{ij} + I_{it} \le b_{it} + I_{i,t-1} \qquad i = 1, \ldots, m;\ t = 1, \ldots, T \qquad (3)$$

$$\sum_{i=1}^{m} x_{ij} = 1 \qquad j = 1, \ldots, n$$

$$x_{ij} \in \{0, 1\} \qquad i = 1, \ldots, m;\ j = 1, \ldots, n$$

$$I_{i0} = 0 \qquad i = 1, \ldots, m$$

$$I_{it} \ge 0 \qquad i = 1, \ldots, m;\ t = 1, \ldots, T$$

where $x_{ij}$ is equal to 1 if customer $j$ is assigned to facility i and zero otherwise, and $I_{it}$ represents the amount of product in storage at facility $i$ at the end of period $t$. Hereafter $x \in \mathbb{R}^{mn}$ will denote the vector with components $x_{ij}$ and similarly for $I \in \mathbb{R}^{mT}$.

Romeijn and Romero Morales [21] have shown for a variant of the MPSSP that the inventory variables can be eliminated, at the expense of introducing convexity in the objective function, i.e., an equivalent formulation with a convex objective function exists. In our case, this reformulation of the MPSSP yields a Single-Sourcing Problem (hereafter SSP) with convex objective function.

Proposition 4.1 (P0) can be equivalently reformulated as:

$$\text{minimize} \sum_{i=1}^{m}\sum_{j=1}^{n}\left(\sum_{t=1}^{T} c_{ijt}\right)x_{ij} + \sum_{i=1}^{m} H_i\left(\sum_{j=1}^{n} d_j x_{ij}\right)$$

subject to (P)

$$\sum_{j=1}^{n} d_j x_{ij} \le \min_{t=1,\ldots,T}\left(\frac{\sum_{\tau=1}^{t} b_{i\tau}}{\sum_{\tau=1}^{t} \sigma_\tau}\right) \qquad i = 1, \ldots, m$$

$$\sum_{i=1}^{m} x_{ij} = 1 \qquad j = 1, \ldots, n$$

$$x_{ij} \in \{0, 1\} \qquad i = 1, \ldots, m;\ j = 1, \ldots, n$$

where $H_i(u)$ is the convex function given by the optimal value of the following problem

$$\text{minimize} \sum_{t=1}^{T} h_{it} I_t$$

subject to

$$I_t - I_{t-1} \le b_{it} - \sigma_t u \qquad t = 1, \ldots, T$$

$$I_0 = 0$$

$$I_t \ge 0 \qquad t = 1, \ldots, T.$$

Proof: Let $F$ be the feasible region of $(P_0)$. By decomposing $(P_0)$, we obtain the following equality

$$\min_{(x,I)\in F}\left(\sum_{t=1}^{T}\sum_{i=1}^{m}\sum_{j=1}^{n} c_{ijt}x_{ij} + \sum_{t=1}^{T}\sum_{i=1}^{m} h_{it}I_{it}\right) =$$

$$= \min_{x:\exists I'(x,I')\in F}\left(\sum_{i=1}^{m}\sum_{j=1}^{n}\left(\sum_{t=1}^{T} c_{ijt}\right)x_{ij} + \min_{(x,I)\in F}\sum_{t=1}^{T}\sum_{i=1}^{m} h_{it}I_{it}\right)$$

$$= \min_{x:\exists I'(x,I')\in F}\left(\sum_{i=1}^{m}\sum_{j=1}^{n}\left(\sum_{t=1}^{T} c_{ijt}\right)x_{ij} + H(x)\right)$$

where $H(x)$ is equal to

$$\text{minimize } \sum_{i=1}^{m}\sum_{t=1}^{T} h_{it}I_{it}$$

subject to

$$I_{it} - I_{i,t-1} \le b_{it} - \sigma_t \cdot \sum_{j=1}^{n} d_j x_{ij} \qquad i = 1, \ldots, m;\ t = 1, \ldots, T$$

$$I_{i0} = 0 \qquad i = 1, \ldots, m$$

$$I_{it} \ge 0 \qquad i = 1, \ldots, m;\ t = 1, \ldots, T$$

This problem is separable in $i$, and moreover for each $i = 1, \ldots, m$ it only depends on $\sum_{j=1}^{n} d_j x_{ij}$. Thus, $H(x) = \sum_{i=1}^{m} H_i\left(\sum_{j=1}^{n} d_j x_{ij}\right)$. Now we will show that the feasible region of the decomposed problem is equal to the feasible region of ($P$). Consider some $x$ so that there exists a feasible solution $(x, I)$ for ($P_0$). For each facility $i$, we aggregate the capacity constraints over all the periods. Then, we obtain

$$\sum_{t=1}^{T}\left(\sigma_t \cdot \sum_{j=1}^{n} d_j x_{ij} + I_{it}\right) \le \sum_{t=1}^{T}(b_{it} + I_{i,t-1})$$

$$\left(\sum_{t=1}^{T}\sigma_t\right)\cdot\sum_{j=1}^{n} d_j x_{ij} + \sum_{t=1}^{T} I_{it} \le \sum_{t=1}^{T} b_{it} + \sum_{t=1}^{T} I_{i,t-1}$$

$$\left(\sum_{t=1}^{T}\sigma_t\right)\cdot\sum_{j=1}^{n} d_j x_{ij} + I_{iT} \le \sum_{t=1}^{T} b_{it} + I_{i0}$$

which is equivalent to

$$\left(\sum_{t=1}^{T}\sigma_t\right)\cdot\sum_{j=1}^{n} d_j x_{ij} + I_{iT} \le \sum_{t=1}^{T} b_{it}$$

and this implies

$$\left(\sum_{t=1}^{T}\sigma_t\right)\cdot\sum_{j=1}^{n} d_j x_{ij} \le \sum_{t=1}^{T} b_{it}$$

The previous inequality shows that x is feasible for ($P$). Now, consider a feasible solution $x$ to ($P$). Then, we know there exists a vector $y \in \mathbb{R}^{mT}$ so that

$$y_{it} \le b_{it} \qquad i = 1, \ldots, m;\ t = 1, \ldots, T$$

and

$$\sum_{t=1}^{T} y_{it} = \sum_{t=1}^{T}\sigma_t \sum_{j=1}^{n} d_j x_{ij} \qquad i = 1, \ldots, m$$

(Note that y can be interpreted as a set of feasible production levels corresponding to $(x, I)$ in the original three-level formulation of ($P_0$).) Now, define $I_{it}$ as

$$I_{it} = \sum_{t=1}^{T} y_{it} - \left(\sum_{t=1}^{T} \sigma_t\right) \cdot \sum_{j=1}^{n} d_j x_{ij}$$

for each $i = 1, \ldots, m$ and $t = 1, \ldots, T$. It is easy to see that $I_{it}$ is nonnegative, and $(x, I) \in F$.

This means that $x$ is a feasible solution for the decomposed problem. With respect to function $H_i(u)$ it is easy to see that has a finite value and thus by strong LP-duality we obtain

$$H_i(u) = \min\left\{\sum_{t=1}^{T} h_{it} I_t : I_t - I_{t-1} \le b_{it} - \sigma_t u,\ I_0 \ge 0,\ t = 1, \cdots, T\right\}$$

$$= \max\left\{\sum_{t=1}^{T} (\sigma_t u - b_{it}) w_t : w \in W_i\right\}$$

where

$$W_i = \{w \in \mathbb{R}^T : -w_t + w_{t+1} \ge h_{it}, t = 1, \ldots, T-1; w_t \ge 0, t = 1, \ldots, T\}$$

Now let $\mu \in [0,1]$ and fix $u; u' \in \mathbb{R}$. Then

$$\max\left\{\sum_{t=1}^{T} ((\mu u + (1-\mu)u')\sigma_t - b_{it}) w_t : w \in W_i\right\}$$

$$= \max\left\{\mu \sum_{t=1}^{T} (\sigma_t u - b_{it}) w_t + (1-\mu) \sum_{t=1}^{T} (\sigma_t u' - b_{it}) w_t : w \in W_i\right\}$$

$$\le \mu \max\left\{\sum_{t=1}^{T} (\sigma_t u - b_{it}) w_t : w \in W_i\right\} +$$

$$(1-\mu) \max\left\{\sum_{t=1}^{T} (\sigma_t u' - b_{it}) w_t : w \in W_i\right\}$$

which shows the convexity of $H_i(u)$. □

The function $H_i$ calculates the minimal inventory costs at facility $i$ needed to be able to supply the customers assigned to it. We may observe that the value of the inventory costs at each facility only depends on the total demand required by the customers assigned to it. The previous proposition tells us that the MPSSP belongs to the class of convex assignment problems introduced in Section 3.1 by choosing

$$g_i(z) = \sum_{j=1}^{n} \left(\sum_{t=1}^{T} c_{ijt}\right) z_j + H_i\left(\sum_{j=1}^{n} d_j z_i\right) \qquad \text{for each } z \in \mathbb{R}^n$$

$$X_i = \left\{ z \in [0,1]^n : \sum_{j=1}^{n} d_j z_j \le \min_{t=1,\ldots,T} \left( \frac{\sum_{\tau=1}^{t} b_{i\tau}}{\sum_{\tau=1}^{t} \sigma_\tau} \right) \right\}$$

We know that function $H_i$ is convex. In fact, it is easy to show that this function is also piecewise linear. This is illustrated by an example, where we will suppress the index $i$ for convenience. Consider $n = 1$, $T = 3$, and

$\sigma = (1,1,1)^T$

$h = (2,2,2)^T$

$d_1 = 25$

$b = (50,20,10)^T$

In that case, we have that $H(z_1)$ is equal to the optimal value of

$$\text{minimize } 2(I_1 + I_2 + I_3)$$

subject to

$$\begin{aligned} I_1 - I_0 &\leq 50 - 25z_1 \\ I_2 - I_1 &\leq 20 - 25z_1 \\ I_3 - I_2 &\leq 10 - 25z_1 \\ I_0 &= 0 \\ I_t &\geq 0 \qquad t = 1, 2, 3. \end{aligned}$$

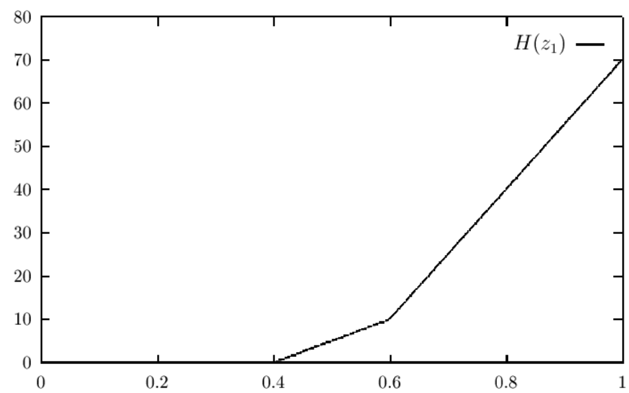

Figure 4: The inventory costs

Figure 4 plots the optimal objective function value of its LP-relaxation as a function of the fraction $z_1$ of the item added to the knapsack. Thus, we observe that it is a piecewise linear function in the fraction $z_1$ added to the knapsack. Note that each breakpoint corresponds to a new inventory variable becoming positive. In this particular case, all inventory variables are equal to zero if the fraction of the demand supplied is below 0.4, i.e., $z_1 \in [0; 0:4]$. If $z_1 \in (0:4; 0:6]$, $I_2$ becomes positive. Finally, if $z_1 \in (0:6; 1]$, $I_1$ also becomes positive.

## 6. Conclusions

In this paper we have presented a direct search approach that was developed for the Generalized Assignment Problem (GAP) to a much richer class of problems, which we have called CAP (Convex Assignment Problems). The viability of this approach depends critically on the possibility of solving the relaxed problem efficiently. We have identified an important subclass of problems, containing many variants of the multi-period single-sourcing problem (MPSSP), as well as some variants of the GAP, for which this is the case. We have applied the method to a particular variant of the MPSSP.

References


[1] C. Barnhart, E.L. Johnson, G.L. Nemhauser, M.W.P. Savelsbergh, and P. Vance. Branch-and-price: column generation for solving huge integer programs. Operations Research, 46(3):316-329, 1998.

[2] J.F. Benders, J.A. Keulemans, J.A.E.E. van Nunen, and G. Stolk. A decision support program for planning locations and allocations with the aid of linear programming. In C.B. Tilanus, O.B. de Gaus, and J.K. Lenstra, editors, Quantitative Methods in Management: cases studies of failures and successes, chapter 4, pages 29-34. John Wiley & Sons, Chichester, 1986.

[3] D.G. Cattrysse, M. Salomon, and L.N. Van Wassenhove. A set partitioning heuristic for the generalized assignment problem. European Journal of Operational Research, 72:167{174, 1994.

[4] L.M.A. Chan, A. Muriel, and D. Simchi-Levi. Supply-Chain Management: Integrating inventory and transportation. Research Report, Department of Industrial Engeneering and Management Sciences, Northwestern University, Evanston, Illinois, 1998.

[5] Z.-L. Chen and W.B. Powell. Solving parallel machine scheduling problems by column generation. INFORMS Journal on Computing, 11(1):78-94, 1999.

[6] CPLEX Reference Manual. ILOG CPLEX 6.5. ILOG, Inc., Incline Village, Nevada, 1999.

[7] F. Duran. A large mixed integer production and distribution program. European Journal of Operational Research, 28:207-217, 1987.

[8] J.A. Ferland, A. Hertz, and A. Lavoie. An object-oriented methodology for solving assignment-type problems with neighborhood search techniques. Operations Research, 44(2):347-359, 1996.

[9] B. Fleischmann. Designing distribution systems with transport economies of scale. European Journal of Operational Research, 70:31- 42, 1993.

[10] A. Geo_rion and G.W. Graves. Multicommodity distribution system design by Benders decomposition. Management Science, 20(5):822-844, 1974.

[11] P.C. Gilmore and R.E. Gomory. A linear programming approach to the cutting-stock problem. Operations Research, 9:849-859, 1961.

[12] N.G. Hall and M.E. Posner. Generating experimental data for scheduling problems. Technical report, The Ohio State University, 1996.

[13] J.B. Hiriart-Urruty and C. Lemar_echal. Convex analysis and minimization algorithms: Fundamentals, volume 1. Springer-Verlag, Berlin, 1993.

[14] S. Martello and P. Toth. An algorithm for the generalized assignment problem. In J.P. Brans, editor, Operational Research, pages 590-603, Amsterdam, 1981. IFORS, North-Holland.

[15] S. Martello and P. Toth. Knapsack problems, algorithms and computer implementations. John Wiley & Sons, New York, 1990.

[16] J.B. Mazzola and A.W. Neebe. Resource-constrained assignment scheduling. Operations Research, 34(4):560-572, 1986.

[17] A.H.G. Rinnooy Kan, L. Stougie, and C. Vercellis. A class of generalized greedy algorithms for the multi-knapsack problem. Discrete Applied Mathematics, 42:279-290, 1993.

[18] H.E. Romeijn and N. Piersma. A probabilistic feasibility and value analysis of the generalized assignment problem. ERASM Management Report Series no. 293, Rotterdam School of Management, Erasmus University Rotterdam, 1996.

[19] H.E. Romeijn and D. Romero Morales. A class of greedy algorithms for the generalized assignment problem. ERASM Management Report Series no. 40(13), Rotterdam School of Management, Erasmus University Rotterdam, 1997. Forthcoming in Discrete Applied Mathematics.

[20] H.E. Romeijn and D. Romero Morales. Generating experimental data for the generalized assignment problem. ERASM Management Report Series no. 1998-9, Rotterdam School of Management, Erasmus University Rotterdam, 1998.

[21] H.E. Romeijn and D. Romero Morales. An asymptotically optimal greedy heuristic for the multi-period single-sourcing problem: the cyclic case. ERASM Management Report Series no. 20-1999, Rotterdam School of Management, Erasmus UniversityRotterdam, 1999.

[22] M.P.W. Savelsbergh. A branch{and{price algorithm for the generalized assignment problem. Operations Research, 45(6):831-841, 1997.

[23] V. Srinivasan and G.L. Thompson. An algorithm for assigning uses to sources in a special class of transportation problems. Operations Research, 21:284-295, 1972.