Vietnam Journal of Mathematics **25**:3 (1997) 203–209

Vietnam Journal
of
MATHEMATICS

On a Non-Convex Optimization Problem in the Inventory Control System*

Nguyen Duc Nghia and Nguyen Trong Toan
Department of Mathematics, Hanoi Polytechnical Institute, Hanoi, Vietnam

Received August 25, 1994
Revised December 5, 1996

Abstract. We investigate the inventory control problem. Based on the specially structured feasible set, we present an algorithm with polynomial complexity to solve it. The result can be also applied to more general problems.

1. Introduction

This paper presents an algorithm for a non-convex optimization problem closely concerning the inventory control system. Consider the dynamic inventory control problem introduced in [1]. Our aim is to make a plan to stock a sort of goods in a time interval partitioned into n subperiods $[t_j, t_{j+1}] (j = 1, 2, \ldots, n)$, so that the buying and carrying costs would be minimum. The mathematical model of the problem is described as follows.

Minimize

$$f(x, y) = \sum_{j=1}^{n} (A_j \operatorname{sign} x_j + c_j x_j + I_j c_j y_{j+1}) \tag{1}$$

subject to

$$y_{j+1} = y_j + x_j - D_j, \quad j = 1, 2, \ldots, n \tag{2}$$

$$x_j \geq 0, \quad j = 1, 2, \ldots, n \tag{3}$$

$$y_{j+1} \geq 0, \quad j = 1, 2, \ldots, n \tag{4}$$

where the constants

(i) A_j $(j = 1, 2, \ldots, n)$ denotes the fixed charge for received order in the period $[t_j, t_{j+1}]$,

* The publication is completed with financial support from the National Basic Research Program in Natural Sciences.

(ii) c_j $(j = 1, 2, \ldots, n)$ the buying price of a unit of the goods in the period $[t_j, t_{j+1}]$,
(iii) I_j $(j = 1, 2, \ldots, n)$, the inventory carrying charge in the period $[t_j, t_{j+1}]$,
(iv) $D_j \geq 0$ $(j = 1, 2, \ldots, n)$, the demand of the goods in the period $[t_j, t_{j+1}]$,
(v) y_1 , the quantity of the goods at the beginning of the planning time,

and the variables

(vi) x_j $(j = 1, 2, \ldots, n)$ denotes the received order of the goods in the period $[t_j, t_{j+1}]$,
(vii) y_{j+1} $(j = 1, 2, \ldots, n)$, the remain quantity at the end of the period $[t_j, t_{j+1}]$.

It is clear that the objective function of the problem (1)–(4) is concave, so we have a concave programming problem with linear constraints. In order to solve the problem, Hadley and Whithin [1] applied a method of the dynamic programming. Based on the study of the structure of the feasible set for the given problem, we present an algorithm with polynomial complexity to solve it.

2. Background of the Algorithm

At first, the problem (1)–(4) is modified into a simpler form, which depends only on the variables x_j.

From condition (2), we obtain

$$y_{j+1} = y_1 + \sum_{i=1}^{j} x_i - \sum_{i=1}^{j} D_i, \quad j = 1, 2, \ldots, n.$$

Let

$$d_j = \sum_{i=1}^{j} D_i - y_1, \tag{5}$$

$$C_j = c_j + \sum_{i=j+1}^{n} I_i c_i,$$

$$f_j(x_j) = A_j \operatorname{sign} x_j + C_j x_j, \quad j = 1, 2, \ldots, n.$$

Then the problem can be rewritten in the following equivalent form:

$$f(x) = \sum_{j=1}^{n} f_j(x_j) \to \min \tag{6}$$

subject to

$$\sum_{j=1}^{i} x_j \geq d_i, \quad i = 1, 2, \ldots, n, \tag{7}$$

$$x_j \geq 0, \quad j = 1, 2, \ldots, n. \tag{8}$$

Denote by Ω the set of all feasible solutions of the problem (6)–(8).

It is clear that the function $f(x)$ is still concave. Denote by $D_k(\lambda), \lambda \geq d_k$ the polytope defined by the following system of inequalities:

$$\begin{gathered}\sum_{j=1}^{i} x_j \geq d_i, \quad i = 1, 2, \ldots, k-1, \\ \sum_{j=1}^{k} x_j \geq \lambda, \\ x_j \geq 0, \quad j = 1, 2, \ldots, k.\end{gathered} \tag{9}$$

From condition (5), it follows that $d_1 \leq d_2 \leq \cdots \leq d_n$. Additionally, we suppose that $d_1 > 0$. Now we study the properties of the polytope $D_k(\lambda)$. It is easy to see that $x^0 = (\lambda, 0, \ldots, 0)$ is a vertex of $D_k(\lambda)$. Further, if $x = (x_1, x_2, \ldots, x_k) \in D_k(\lambda)$, then $x_1 > 0$. The following lemma gives a way to calculate the positive coordinates of any vertex of $D_k(\lambda)$ knowing its index set.

Lemma 1. *Suppose $x^k = (x_1^k, x_2^k, \ldots, x_k^k)$ is a vertex of the polytope $D_k(\lambda)$ and $J^+ = \{1, j_1, j_2, \ldots, j_q \leq k\}$ $(1 < j_1 < \cdots < j_q)$ is the index set of the positive coordinates. Then*

$$\begin{aligned} x_1^k &= d_{j_1-1} \\ x_{j_1}^k &= d_{j_2-1} - d_{j_1-1} \\ &\ldots \\ x_{j_{q-1}}^k &= d_{j_q-1} - d_{j_{q-1}-1} \\ x_{j_q}^k &= \lambda - d_{j_q-1}. \end{aligned} \tag{10}$$

Proof. Since x^k is a vertex of $D_k(\lambda)$ and has $q+1$ positive coordinates at indexes $1, j_1, \ldots, j_q$, it must strictly satisfy $q+1$ inequalities obtained from the system (9) by eliminating the null coordinates of x^k as follows:

$$\begin{aligned} x_1 &\geq d_1 \\ x_1 &\geq d_2 \\ &\ldots \\ x_1 &\geq d_{j_1-1} \\ x_1 + x_{j_1} &\geq d_{j_1} \\ x_1 + x_{j_1} &\geq d_{j_1+1} \\ &\ldots \\ x_1 + x_{j_1} &\geq d_{j_2-1} \\ &\ldots \end{aligned} \tag{11}$$

$$x_1 + x_{j_1} + x_{j_2} + \cdots + x_{j_q} \geq d_{j_q}$$
$$x_1 + x_{j_1} + x_{j_2} + \cdots + x_{j_q} \geq d_{j_q+1}$$
$$\cdots$$
$$x_1 + x_{j_1} + x_{j_2} + \cdots + x_{j_q} \geq \lambda.$$

There are $q+1$ groups of inequalities in the system (11). The inequalities in each group have completely similar left sides. Since $0 < d_1 \leq d_2 \leq \cdots \leq d_k$ and $\lambda \geq d_k$, if x^k satisfies strictly some of the inequalities in a group, then it must also strictly satisfy the final inequality of the group. Since x^k has exactly $q+1$ positive coordinates, x^k must satisfy strictly $q+1$ final inequalities of all groups. It means x^k is the solution of the following system of equations:

$$\begin{aligned} x_1 &= d_{j_1-1} \\ x_1 + x_{j_1} &= d_{j_2-1} \\ &\cdots \\ x_1 + x_{j_1} + x_{j_2} + \cdots + x_{j_q} &= \lambda. \end{aligned} \tag{12}$$

Solving the system (12) gives (10). The lemma is proved. ■

From the proof of Lemma 1, it is easy to see that each vertex of $D_k(\lambda)$ is completely defined by the index set of its positive coordinates. Therefore, the number of vertices of $D_k(\lambda)$ is 2^{k-1} and the one of the polytope Ω is 2^{n-1}.

Corollary 1. *If $x^k = (x_1^k, x_2^k, \ldots, x_k^k)$ is a vertex of $D_k(\lambda)$, then*
(i) $\sum_{j=1}^{k} x_j^k = \lambda$,
(ii) $x_k^k = 0$ *or* $x_k^k = \lambda - d_{k-1}$.

Proof. Indeed, it is clear that (i) follows directly from the final equality of system (12) and (ii) follows from the final equality of (10). ■

Now let

$$P_k(\lambda) = \min\{g_k(x) = f_1(x_1) + f_2(x_2) + \cdots + f_k(x_k) \colon x \in D_k(\lambda)\},$$
$$\lambda \geq d_k; \quad k = 1, 2, \ldots, n.$$

Lemma 2.

$$P_1(\lambda) = f_1(\lambda), \tag{13}$$

$$P_k(\lambda) = \min\{P_{k-1}(\lambda) + f_k(0), P_{k-1}(d_{k-1}) + f_k(\lambda - d_{k-1})\},$$
$$k = 2, 3, \ldots, n. \tag{14}$$

Proof. Equality (13) follows directly from the definition of the function $P_1(\lambda)$. Suppose $k > 1$. Since $g_k(x)$ is concave, its minimum value is attained at some vertex of $D_k(\lambda)$. On the other hand, by Corollary 1, the kth coordinate of whichever vertex of $D_k(\lambda)$ must be 0 or $\lambda - d_{k-1}$. So

$$P_k(\lambda) = \min\{\min\{g_k(x) \colon x \in D_k(\lambda), x_k = 0\}, \min\{g_k(x) \colon x \in D_k(\lambda), x_k = \lambda - d_{k-1}\}\}.$$

Since

$$\begin{aligned}\min\{g_k(x) \colon x \in D_k(\lambda), x_k = 0\} &= \min\{f_1(x_1) + \cdots + f_{k-1}(x_{k-1}) + f_k(0) \colon x \in D_{k-1}(\lambda)\} \\ &= P_{k-1}(\lambda) + f_k(0),\end{aligned}$$

and

$$\begin{aligned}\min\{g_k(x) \colon x \in D_k(\lambda), x_k = \lambda - d_{k-1}\} &= \min\{f_1(x_1) + \cdots + f_{k-1}(x_{k-1}) + f_k(\lambda - d_{k-1}) \colon x \in D_{k-1}(d_{k-1})\} \\ &= P_{k-1}(d_{k-1}) + f_k(\lambda - d_{k-1}),\end{aligned}$$

the equality (14) holds. The proof is complete. ■

Lemma 2 suggests the development of an algorithm for solving the problem (6)–(8). We now turn to the description of the algorithm.

3. Algorithm

It is clear that the problem is equivalent to the following:
Calculate

$$P_n(d_n) = \min\{f(x) \colon x \in \Omega = D_n(d_n)\}.$$

Taking into account Lemma 2, we can rewrite the expression (14) as follows:

$$\begin{aligned}P_k(\lambda) = \min\{&f_1(\lambda) + f_2(0) + f_3(0) + \cdots + f_k(0), \\ &P_1(d_1) + f_2(\lambda - d_1) + f_3(0) + \cdots + f_k(0), \\ &\ldots \\ &P_{i-1}(d_{i-1}) + f_i(\lambda - d_{i-1}) + f_{i+1}(0) + \cdots + f_k(0), \\ &\ldots \\ &P_{k-1}(d_{k-1}) + f_k(\lambda - d_{k-1})\}.\end{aligned}$$

Hence, $P_k(\lambda)$ would be obtained if $P_i(d_i), i = 1, 2, \ldots, k-1$ had already been done. Thus, to calculate $P_n(d_n)$, we should calculate $P_k(d_k)$, $k = 1, 2, \ldots, n$ subsequently. Then, to find the solution x^* of the problem (6)–(8), by using Lemma 1, we only need saving the maximum index j_k of the positive coordinates of the solution for each problem $P_k(d_k)$, respectively.

Algorithm 1.
Initial step. Calculate

$$f_{ii} = f_i(0), i = 2, 3, \ldots, n;$$
$$f_{ij} = f_i(0) + f_{i+1}(0) + \cdots + f_j(0), \quad 2 \le i < j \le n.$$

Step 1. Let $P_1(d_1) = f_1(d_1)$ and $j_1 = 1$.
Step k = 2, 3, ..., n. Calculate

$$F_{1k} = f_1(d_k) + f_{2k},$$
$$F_{2k} = P_1(d_1) + f_2(d_k - d_1) + f_{3k},$$
$$\ldots$$
$$F_{ik} = P_{i-1}(d_{i-1}) + f_i(d_k - d_{i-1}) + f_{i+1,k},$$
$$\ldots$$
$$F_{kk} = P_{k-1}(d_{k-1}) + f_k(d_k - d_{k-1}).$$

Let

$$P_k(d_k) = \min\{F_{ik} : i = 1, 2, \ldots, k\}; \quad j_k = \arg\min\{F_{ik} : i = 1, 2, \ldots, k\}.$$

At the end of the algorithm, we obtain $P_n(d_n)$, the optimal value of the objective function (6), and a sequence of indices $j_1, j_2, \ldots, j_n$. To determine the optimal solution $x^* = (x_1^*, x_2^*, \ldots, x_n^*)$, we use the following:

Procedure φ.
Step 1. Let $k = n, d_0 = 0$.
Step 2. Let

$$x_j^* = 0, \quad j_k + 1 \le j \le k,$$
$$x_{j_k}^* = d_k - d_{j_k - 1}.$$

Step 3. If $k = 1$, then stop. Otherwise, let $k = j_k - 1$ and return to Step 2.

Theorem. *The algorithm provides the exact solution to the problem (6)–(8) and has the computational complexity $O(n^2)$.*

Proof. By Lemma 2, the algorithm must be exact. It is evident that the initial step requires $O(n^2)$ operations. Besides, since every other step requires $O(n)$ operations, the algorithm has the computational complexity $O(n^2)$.

The proof is complete. ■

4. Combinatorial Problem on the Extremal Points of Polytope Ω

Let $F(x)$ be a separable function of the following form:

$$F(x) = f_1(x_1) *_1 f_2(x_2) *_2 \ldots *_{n-1} f_n(x_n), \quad x = (x_1, x_2, \ldots, x_n) \in \Omega, \tag{15}$$

where $*_1, *_2, \ldots, *_{n-1}$, denotes an ordered sequence of additions, multiplications or exponents, and $f_k(x_k) \geq 0, k = 1, \ldots, n$.

Consider the following optimization problem:

$$\text{Minimize } F(x), \quad \text{subject to } x \in V(\Omega), \tag{16}$$

where $V(\Omega)$ denotes the vertex set of Ω.

Let

$$F_k(x) = f_1(x_1) *_1 f_2(x_2) *_2 \ldots *_{k-1} f_k(x_k), \quad k = 1, 2, \ldots, n, \; x \in D_k(\lambda).$$

It is easy to see that $F(x) = F_n(x)$ and the functions $F_k(x)$ satisfy the following property:

$$\begin{aligned} &\min\{F_k(x) \colon x \in D_k(\lambda), x_k = const\} \\ &\quad = \min\{F_{k-1}(x) \colon x \in D_{k-1}(\lambda - x_k)\} *_{k-1} f_k(x_k). \end{aligned}$$

Therefore, based on the structure of the polytope $D_k(\lambda)$, we can modify Algorithm 1 to solve the problem (16).

Algorithm 2.
Step 1. Let $p_1(d_1) = f_1(d_1)$, $j_1 = 1$.
Step $k = 2, \ldots, n$. Calculate

$$\begin{aligned} F_{1k} &= f_1(d_k) *_1 f_2(0) *_2 f_3(0) *_3 \ldots *_{k-1} f_k(0), \\ F_{2k} &= p_1(d_1) *_1 f_2(d_k - d_1) *_2 f_3(0) *_3 \ldots *_{k-1} f_k(0), \\ &\ldots \\ F_{ik} &= p_{i-1}(d_{i-1}) *_{i-1} f_i(d_k - d_{i-1}) *_i f_{i+1}(0) *_{i+1} \ldots *_{k-1} f_k(0), \\ &\ldots \\ F_{kk} &= p_{k-1}(d_{k-1}) *_{k-1} f_k(d_k). \end{aligned}$$

Let

$$p_k(d_k) = \min\{F_{ik} \colon i = 1, 2, \ldots, k\}$$

and

$$j_k = \arg\min\{F_{ik} : i = 1, 2, \ldots, k\}.$$

In order to find the optimal solution x^*, we can use the procedure φ.

Remark. If $F(x)$ is a concave function satisfying (15), then the problem

$$\min\{F(x) \colon x \in \Omega\}$$

can be reduced to the problem (16) and, therefore, solved by Algorithm 2.

Reference

1. G. Hadley and T. M. Whithin, *Analysis of Inventory Systems*, Prentice Hall, Inc., Englewood Cliff, New Jersey, 1967.