Cowles Foundation Paper 239

Cahiers du Centre d'Etudes de Recherche Opérationnelle
267, avenue Molière Bruxelles 6
Vol. 8, nº 2, 1966

A DECOMPOSITION ALGORITHM FOR QUADRATIC PROGRAMMING**

ANDREW WHINSTON
University of Virginia

INTRODUCTION

In this paper we consider a direct extension of the Dantzig and Wolfe [4] decomposition algorithm for linear programming to the case of a quadratic criterion function. First presented by Dantzig [2] and further elaborated by van de Panne and Whinston [6], the quadratic algorithm we shall use is itself a direct extension of the simplex method. Thus, if the quadratic part of the criterion function is zero, i.e., if we, in fact, have a linear criterion function, the decomposition algorithm is reduced to the one given by Dantzig and Wolfe.

In their paper, Dantzig and Wolfe [3] suggested an extension of their algorithm to the case of a nonlinear convex criterion function (1). This is discussed in more detail in Dantzig [1]. Their suggestion differs from the method being presented here in several ways. First, in the quadratic case our method leads to a finite procedure while their method yields an infinite convergent algorithm. Second, in the solution of the subproblems we must solve linear programming problems while their method requires the solution of the same number of quadratic programming problems. Finally, the method discussed here does not require that the criterion function be separable. Since extension beyond the quadratic case would not possess the characteristics indicated above, we present the argument for the quadratic case only.

Decomposition algorithms arise naturally where the constraint matrix has a block diagonal structure. In this case a large problem may be broken down into a collection of smaller ones which are then tied

---

* This paper is an outgrowth of some joint work with Mr. C. van de Panne as reported in [6]. I am, of course, very much indebted to him. My colleague Menahem Yaari made several very helpful suggestions. I remain responsible for all possible errors.

**Research undertaken by the Cowles Commission for Research in Economics under Task NR 047-006 with the Office of Naval Research.

(1) J.B. ROSEN has presented a decomposition algorithm for nonlinear programming in [7].

together by a series of interconnecting constraints. This type of approach may have some utility where the number of constraints in the original problem is very large.

One may also interpret the decomposition algorithm as a decentralized approach to solving a large decision making problem. Each of the smaller subproblems can be thought of as a division of a firm,while a central coordinating staff is responsible for the coupling conditions. The type of information exchange suggested by the algorithm could be considered as one method of organizing a decentralized decision making system ([2]).

The Algorithm

In presenting the algorithm we shall attempt to illustrate the basic differences between this case and the linear programming problem. With this in mind, we consider only two subsectors in the following problem

$$\operatorname*{Max}_{x_1,\ x_2} f(x_1, x_2) = p_1'x_1 + p_2'x_2 - \tfrac{1}{2}[x_1'x_2'] \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} =$$

$$= p_1'x_1 + p_2'x_2 - \tfrac{1}{2}x_1'C_{11}x_1 - x_1'C_{12}x_2 - \tfrac{1}{2}x_2'C_{22}x_2$$

$$A_1x_1 + A_2x_2 = b_0 \tag{1}$$

$$B_1x_1 \leqslant b_1 \tag{2}$$

$$B_2x_2 \leqslant b_2 \tag{3}$$

$$x_1 \geqslant 0 \qquad x_2 \geqslant 0$$

$$x_1' = (x_1^1 \ldots x_1^{q_1}) \qquad x_2' = (x_2^1 \ldots x_2^{q_2}) \qquad b_0' = (b_{01} \ldots b_{0l})$$

$$p_1' = (p_1^1 \ldots p_1^{q_1}) \qquad p_2' = (p_2^1 \ldots p_2^{q_2})$$

([2]) For an interpretation of the present algorithm in the context of organizational decision making see Whinston [9] and for a general discussion of such applications see Whinston [8].

The matrix $\begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}$ is symmetric positive semidefinite matrix of dimension (q_1+q_2). We shall assume that the convex sets described by the inequalities (2) and (3) are each bounded, in order to simplify the exposition (3).

Let $\{x_{k1}\}$ be the set of extreme points for the constraints of (2), and $\{x_{k2}\}$ the set of extreme points for the constraints of (3). Then any point x_1 satisfying the constraints of (2) can be written as

$$x_1 = \sum_{k=1}^{k_1} \rho_{k1}\, x_{k1} \tag{4}$$

$$\Sigma\rho_{k1} = 1$$

$$\rho_{k1} \geqslant 0$$

Correspondingly we have for (3)

$$x_2 = \sum_{k=1}^{k_2} \rho_{k2}\, x_{k2} \tag{5}$$

$$\Sigma\rho_{k2} = 1$$

$$\rho_{k2} \geqslant 0$$

Substituting (4) and (5) into the programming problem we have the equivalent problem

$$\max_{\rho_{ki}} p_1' \Sigma\rho_{k1} x_{k1} + p_2' \Sigma\rho_{k2} x_{k2} - \tfrac{1}{2}\Sigma\rho_{k1} x_{k1}' C_{11} \Sigma\rho_{k1} x_{k1}$$

$$- \Sigma\rho_{k1} x_{k1}' C_{12} \Sigma\rho_{k2} x_{k2} - \tfrac{1}{2}\Sigma\rho_{k2} x_{k2}' C_{22} \Sigma\rho_{k2} x_{k2}$$

$$A_1 \Sigma\rho_{k1} x_{k1} + A_2 \Sigma\rho_{k2} x_{k2} = b_0 \tag{6}$$

$$\Sigma\rho_{k1} = 1 \qquad \Sigma\rho_{k2} = 1 \tag{7}$$

$$\rho_{ki} \geqslant 0 \qquad i = 1, 2 \qquad \text{for all } k. \tag{8}$$

(3) Without this assumption we would follow the argument as developed by Dantzig and Wolfe in [4] as it pertains to this point.

The Kuhn-Tucker [5] conditions for this problem are (6), (7), (8) and

$$p_1' x_{k1} - x_{k1}' C_{11} \Sigma \rho_{k1} x_{k1} - x_{k1}' C_{12} \Sigma \rho_{k2} x_{k2} - v' A_1 x_{k1} - \eta_1 + u_{k1} = 0 \quad (9)$$

$$u_{k1}' \rho_{k1} = 0 \quad (10)$$

$$u_{k1} \geqslant 0, \quad (11)$$

$$p_2' x_{k2} - x_{k2} C_{22} \Sigma \rho_{k2} x_{k2} - x_{k2}' C_{21} \Sigma \rho_{k1} x_{k1} - v' A_2 x_{k2} - \eta_2 + u_{k2} = 0 \quad (12)$$

$$u_{k2}' \rho_{k2} = 0 \quad (13)$$

$$u_{k2} \geqslant 0 \qquad \text{for all } k. \quad (14)$$

We must show that the transformed problem is still a concave programming problem. We write

$$f(x_1, x_2) = p_1' x_1 + p_2' x_2 - [x_1', x_2'] \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

and show the following:

Theorem:

The criterion function $f(\Sigma \rho_{k1} x_{k1}, \Sigma \rho_{k2} x_{k2})$ is a concave function of $(\rho_1, \rho_2) = (\rho_{11} \ldots \rho_{k_1 1}, \rho_{12} \ldots \rho_{k_2 2})$.

Proof

Since $f(x_1, x_2)$ is concave we have

$$f(\lambda \tilde{x} + (1-\lambda)\hat{x}) \geqq \lambda f(\tilde{x}) + (1-\lambda) f(\hat{x}) \quad (15)$$

where $\tilde{x} = (\tilde{x}_1, \tilde{x}_2)$ and $\hat{x} = (\hat{x}_1, \hat{x}_2)$ are given points. We have

$$\tilde{x} = T\tilde{\rho}$$

$$\hat{x} = T\hat{\rho}$$

where T is a linear transformation. Define

$$g(\rho) = f(T\rho) \quad (16)$$

Then

$$g(\lambda\tilde{\rho}+(1-\lambda)\hat{\rho}) = \tag{17}$$

$$f(\mathrm{T}(\lambda\tilde{\rho}+(1-\lambda)\hat{\rho})) = \tag{18}$$

$$f(\lambda \mathrm{T}\tilde{\rho}+(1-\lambda)\mathrm{T}\hat{\rho}) \geqq \tag{19}$$

$$\lambda f(\mathrm{T}\tilde{\rho})+(1-\lambda)f(\mathrm{T}\hat{\rho}) = \lambda g(\tilde{\rho})+(1-\lambda)g(\hat{\rho}) \tag{20}$$

q.e.d.

Description of the Algorithm

In this section we shall describe in some detail the working of the algorithm. After certain preliminary definitions we shall give a description of the algorithm in terms of the quadratic programming tableau. Then various aspects of the algorithm will be explored in some detail.

Dantzig and Wolfe have pointed out that an important aspect of the algorithm is the sequential generation of the extreme points and resulting columns of the tableau as they are required. To initiate the algorithm, we must have $l+2$ "ρ_{ki}" variables, where l is the dimension of b_0, which are positive and satisfy (6), (7) and (8)([4]). We also have l "v" variables and η_1 and η_2 in the basis. Note that these variables appear in every tableau since their values are unconstrained. Finally, we will have $m+n-l-2$ "u_{ki}" in the basis corresponding to ρ_{ki} not in the basis. In general, the initial set of u_{ki} in the basis will not satisfy (11) and (14), but will satisfy (10) and (13). When conditions (10) and (13) are satisfied, then we say that the tableau is in standard form; otherwise, it is a nonstandard tableau. The initial solution constitutes a feasible, standard form tableau which is not optimal i.e., satisfies conditions (6), (7), (8), (9), (10), (12) and (13) but not necessarily (11) and (14). A feasible solution is one that satisfies (6), (7), and (8). The algorithm proceeds by moving from a feasible solution which is either in standard or nonstandard form to an optimal solution, i.e., a tableau also satisfying (11) and (14) besides the conditions (6), (7), (8), (9), (10), (12) and (13).

([4]) We assume that the constraint matrix has rank $l+2$ and that the initial set of ρ_{ki} variables are associated with an independent set of columns. It should be observed that degeneracy can be handled exactly as in the case of linear programming.

*Set-Up Tableau for the Algorithm**

Basic variables	Value of basic variables	u_1	u_2	v	ρ_1	ρ_2	η_1	η_2	y_1	y_2	y_3
u_1'	$-P_1'$	I	0	$-\bar{A}_1'$	$-\bar{C}_{11}$	$-\bar{C}_{12}$	$-l_1'$	0	0	0	0
u_2'	$-P_2'$	0	I	$-\bar{A}_2'$	$-\bar{C}_{21}$	$-\bar{C}_{22}$	0	$-l_2'$	0	0	0
y_1'	b_0'	0	0	0	$\bar{A}_1$	$\bar{A}_2$	0	0	I	0	0
y_2'	1	0	0	0	l_1	0	0	0	0	1	0
y_3'	1	0	0	0	0	l_2	0	0	0	0	1

$u_1 = (u_{k1} \cdots u_{k_1 1})$ $\qquad$ $\rho_1 = (\rho_{11} \cdots \rho_{k_1 1})$

$u_2 = (u_{k2} \cdots u_{k_2 2})$ $\qquad$ $\rho_2 = (\rho_{12} \cdots \rho_{k_2 2})$

$v = (v_1 \cdots v_l)$

$\bar{A}_1 = (A_1 x_{11} \cdots A_1 x_{k_1 1})$ $\qquad$ $P_1 = (p_1' x_{11} \cdots p_1' x_{k_1 1})$

$\bar{A}_2 = (A_2 x_{12} \cdots A_2 x_{k_2 2})$ $\qquad$ $P_2 = (p_2' x_{21} \cdots p_2' x_{k_2 2})$

$l_1 = (1 \cdots 1_{k_1})$

$l_2 = (1 \cdots 1_{k_2})$

$$\bar{C}_{11} = \begin{bmatrix} x_{11}' C_{11} x_{11} & \cdots\cdots & x_{11}' C_{11} x_{k_1 1} \\ \vdots & & \\ \vdots & & \\ x_{k_1 1}' C_{11} x_{11} & \cdots\cdots & x_{k_1 1}' C_{11} x_{k_1 1} \end{bmatrix}$$

* We add the artificial variables $y = (y_1, y_2, y_3)$ to the constraints in order to write the tableau in this form. I refers to an identity matrix of appropriate dimension.

$$\bar{C}_{12} = \begin{bmatrix} x'_{11} C_{12} x_{12} & \cdots\cdots & x'_{11} C_{12} x_{k_2 2} \\ \vdots & & \\ \vdots & & \\ x'_{k_1 1} C_{12} x_{12} & \cdots\cdots & x'_{k_1 1} C_{12} x_{k_2 2} \end{bmatrix}$$

$$\bar{C}_{21} = \begin{bmatrix} x'_{12} C_{21} x_{11} & \cdots\cdots & x'_{12} C_{21} x_{k_1 1} \\ \vdots & & \\ \vdots & & \\ x'_{k_2 2} C_{21} x_{11} & \cdots\cdots & x'_{k_2 2} C_{21} x_{k_1 1} \end{bmatrix}$$

$$\bar{C}_{22} = \begin{bmatrix} x'_{12} C_{22} x_{12} & \cdots\cdots & x'_{12} C_{22} x_{k_2 2} \\ \vdots & & \\ \vdots & & \\ x'_{k_2 2} C_{22} x_{12} & \cdots\cdots & x'_{k_2 2} C_{22} x_{k_2 2} \end{bmatrix}$$

Let $\{z_i\}$ be the values of the set of basic variables which are permitted to leave the basis in a particular iteration and let $\{l_{ij}\}$ be the elements in the j^{th} column of the tableau which is the column of the variable to come in the basis. Then the variable to leave the basis is the one whose value z_i is such that

$$\operatorname*{Min}_{i} \left\{ \frac{z_i}{l_{ij}} \mid \frac{z_i}{l_{ij}} \geqslant 0 \qquad l_{ij} \neq 0 \right\}$$

Starting from a basic feasible solution which is not necessarily optimal, the algorithm proceeds in the following manner ([5]):

1. Determine the most negative u_{ki} variable. If there are none the algorithm is terminated.

2. Introduce into the basis the complimetary ρ_{ki} to the u_{ki} chosen in Step 1. The variable to be removed from the basis is chosen from among the "ρ_{ki}" variables in the basis and the "u_{ki}" variable designated in Step one. If the variable designated in Step 1 is removed return to Step 1; if not, go to Step 3.

3. Introduce the "u_{ki}" variable corresponding to the "ρ_{ki}" variable which was just taken out of the basis. The variable to be eliminated from the basis is chosen from the ρ_{ki} in the basis and the variable designated in Step 1. If a ρ_{ki} is eliminated repeat Step 3, if not go back to Step 1.

We now wish to show that given the values of the $l+2$ "ρ_{ki}" variables

([5]) These rules are based on [6].

in the initial basis we may determine $v = (v_1 ... v_l)$ and η_1 and η_2. From the tableau we have the following $l+2$ equations involving v, η_1, η_2 and ρ_{ki} which are in the basis:

$$p_1' x_{k1} = v' A_1 x_{k1} + \sum_{j \varepsilon J_1} \rho_{j1} x_{j1}' C_{11} x_{k1} + \sum_{j \varepsilon J_2} x_{k1}' C_{12} \rho_{j2} x_{j2} + \eta_1 \tag{21}$$

$$p_2' x_{k2} = v' A_2 x_{k2} + \sum_{j \varepsilon J_1} x_{k2}' C_{21} \rho_{j1} x_{j1} + \sum_{j \varepsilon J_2} x_{k2}' C_{22} \rho_{j2} x_{j2} + \eta_2 \tag{22}$$

where

$$J_1 = \{j | \rho_{j1} > 0\} \qquad \text{and} \qquad J_2 = \{j | \rho_{j2} > 0\}$$

Each of the equations corresponds to a particular $\rho_{ki} > 0$ and consequently there are $l+2$ equations. The set of extreme points indexed $\{x_{k1}\}$ which appears in these equations is associated with $\rho_{k1} > 0$. As a result these points are known. The extreme points indexed $\{x_{k2}\}$ are associated in the tableau with columns of $\rho_{k2} > 0$ and again are known. Thus we have $l+2$ equations in $l+2$ unknowns $(v_1 ... v_l)$, η_1 and η_2. In principle, we may solve for v, η_1 and η_2 (6).

We now wish to discuss how Step 1 in the algorithm is carried out. At some stage in order to determine the most negative u_{ki} we have the following equations for u_{k1} and u_{k2} :

$$u_{k1} = -p_1' x_{k1} + x_{k1}' C_{11} \Sigma \rho_{k1} x_{k1} + x_{k1}' C_{12} \Sigma \rho_{k2} x_{k2} x_{k2} + v_1' A_1 x_{k1} + \eta_1 \tag{23}$$

$$u_{k2} = -p_2' x_{k2} + x_{k2}' C_{22} \Sigma \rho_{k2} x_{k2} + x_{k2}' C_{21} \Sigma \rho_{k1} x_{k1} + v_1' A_2 x_{k2} + \eta_2 \tag{24}$$

According to Step 1 we are to determine that particular u_{ki} which is minimum. This is equivalent to minimizing the expression for u_{ki} over all extreme points $\{x_{ki}\}$. In order to express the problem in a more interpretable fashion we may take the maximization of the negative expression for u_{ki}. Thus, we have

$$\begin{gathered} \operatorname*{Max}_{x_1} p_1' x_1 - x_1' C_{11} \Sigma \rho_{k1} x_{k1} - x_1' C_{12} \Sigma \rho_{k2} x_{k2} - v' A_1 x_1 - \eta_1 \\ \text{s.t. } B_1 x_1 \leqslant b_1 \\ x_1 \geqslant 0 \end{gathered} \tag{25}$$

$$\begin{gathered} \operatorname*{Max}_{x_2} p_2' x_2 - x_2' C_{22} \Sigma \rho_{k2} x_{k2} - x_2' C_{21} \Sigma \rho_{k1} x_{k1} - v' A_2 x_2 - \eta_2 \\ \text{s.t. } B_2 x_2 \leqslant b_2 \\ x_2 \geqslant 0 \end{gathered} \tag{26}$$

(6) This point will be discussed below in more detail.

Both maximization problems are linear programming problems and therefore the solution is at an extreme point. From these solutions we obtain both the minimum u_{ki} value and its associated extreme point.

In the course of the algorithm we must be able to determine the current basis values of ρ_{ki}, v_i, η_i and the particular u_{ki} currently designated by Step 1. In a certain iteration we refer to the u_{ki} designated in the preceding Step 1 as u_{ki}^*. Of course as the algorithm proceeds the particular variable designated as u_{ki}^* will vary.

At any iteration we are either introducing a ρ_{ki}^* variable complimentary to u_{ki}^* designated by Step 1 or some u_{ki} variable. If we can show that at any iteration we can determine the tableau elements of the particular incoming column associated with ρ_{ki}, v_i, η_i and u_{ki}^* in the basis, we then will know the values of these basic variables throughout the algorithm.

Consider the system of equations (6), (7), (9) and (12). Associated with each variable is a column of coefficients. However, during the course of the algorithm not all coefficients will be known. The algorithm, as represented by the rules 1-3, can be viewed as a procedure for moving from one basic solution to (6), (7), (9) and (12) to another one via a simplex-type pivot. Each basic solution is successively an improvement in terms of certain criteria. What we wish to show is, that, in general, we need only be concerned with a smaller basis matrix—a submatrix of the larger basis matrix where all the coefficients are known.

We first consider the case where we are introducing ρ_{ki}^* into the basis. By determining the appropriate tableau elements for ρ_{ki}^* we may determine which variable will leave the basis among the $\rho_{ki} > 0$ and the appropriate $u_{ki}^* < 0$. Let p_{ki} be the column associated with ρ_{ki}^*. Let B be the basis matrix where we partition $B = (B^1, B^2)$. B^1 contains the columns of u_{ki} in the basis except u_{ki}^* while B^2 consists of the columns associated with ρ_{ki}, v_i, η_i and u_{ki}^* in the basis. Thus we have

$$(B^1, B^2) \begin{pmatrix} \lambda_1 \\ \lambda_2 \end{pmatrix} = p_{ki}$$

where $\lambda' = (\lambda_1, \lambda_2)$ is the vector of tableau elements. By permutation of the rows of $(B^1, B^2) = B$ and p_{ki} we write the problem as

$$\begin{pmatrix} B_{11} & B_{12} \\ 0 & B_{22} \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \lambda_2 \end{pmatrix} = \begin{pmatrix} p_{ki1} \\ p_{ki2} \end{pmatrix}$$

where $B_{11} = I$. The dimension of I is equal to the number of columns in B^1. Note also that B_{22} is a square submatrix. Next we wish to show that all the coefficients in B_{22} are known. The last $l+2$ rows of B_{22} correspond to the constraints (6) and (7) and since the columns correspond to "ρ_{ki}" variables in the basis the coefficients are known. The remaining rows of B_{22} consist of equations drawn from (9) and (12) where the "u_{ki}" variable associated with each constraint has been removed from the basis. Therefore the extreme point x_{ki} associated with the complimentary variable ρ_{ki} has been determined. This, combined with the fact that the columns are associated with variables in the basis, establishes that all coefficients are known. A similar argument holds for p_{ki2}. The matrix B must be nonsingular since it is obtainable by a series of simplex pivot operations, therefore the matrix B_{22} is also nonsingular. The system of equations

$$B_{22}\lambda_2 = p_{ki2}$$

is solvable for $\lambda_2 = B_{22}^{-1}\, p_{ki2}$. The vector λ_2 contains the needed tableau elements.

With this result we may determine the tableau elements associated with a ρ^*_{ki} which is to come into the basis. If some u_{ki} is to be introduced into the basis according to Step 3 then with a simular argument we may show that the required tableau elements can be generated.

We have shown that the various steps of the algorithm can be carried out even though not all extreme points are known at every step. Each time a new ρ_{ki} variable is brought into the basis, as a result of Step 2, the criterion function increases. Steps 2 and 3, which involve nonstandard tableaux, do not decrease the value of the criterion function and after a finite number of iterations lead back to a standard tableau where Step 1 is again applied. We then have a series of standard tableaux each one associated with a higher value of the criterion function. Therefore, no standard tableau can ever be repeated. Since the number of extreme points is finite there can be only a finite number of standard tableaux and consequently the algorithm is finite (7).

Example

With the presentation of a specific example we will illustrate the workings of the algorithm. We will also indicate the approach needed in the computational aspects of the method.

(7) For a discussion of these points see [6].

Consider the following problem:

$$\max_{x_1,\, x_2} 6x_1 - 2x_1^2 + 2x_1 x_2 - 2x_2^2$$

$$\text{s.t. } x_1 + x_2 \leqslant 2 \tag{27}$$

$$0 \leqq x_1 \leqq 1 \tag{28}$$

$$0 \leqq x_2 \leqq 1 \tag{29}$$

where (27) is the interconnecting constraint, and (28) and (29) are the sub-problems. To simplify the writing and presentation of the example we will take into consideration the fact that each of the sets represented by (28) and (29) contains two extreme points. The method developed above, however, does not depend on this information being available.

Let

$$\{x_{11}, x_{21}\}$$

be the set of extreme points for the convex set represented by (28) where

$$x_1 = \rho_{11} x_{11} + \rho_{21} x_{21} \tag{30}$$

$$\rho_{11} + \rho_{21} = 1$$

$$\rho_{11}, \rho_{21} \geqq 0$$

and for (29) we have

$$\{x_{12}, x_{22}\} \text{ and}$$

$$x_2 = \rho_{12} x_{12} + \rho_{22} x_{22}$$

$$\rho_{12} + \rho_{22} = 1 \tag{31}$$

$$\rho_{12} \geqq 0, \rho_{22} \geqq 0$$

We have the transformed problem

$$\text{Max } 6(\rho_{11} x_{11} + \rho_{21} x_{21}) - 2(\rho_{11} x_{11} + \rho_{21} x_{21})^2$$

$$+ 2(\rho_{11} x_{11} + \rho_{21} x_{21})(\rho_{12} x_{12} + \rho_{22} x_{22})$$

$$- 2(\rho_{12} x_{12} + \rho_{22} x_{22})^2$$

$$\rho_{11} x_{11} + \rho_{21} x_{21} + \rho_{12} x_{12} + \rho_{22} x_{22} \leqq 2 \tag{32}$$

$$\rho_{11}+\rho_{21} = 1 \tag{33}$$

$$\rho_{12}+\rho_{22}= 1 \tag{34}$$

$$\rho_{11}, \rho_{21}, \rho_{12}, \rho_{22} \geqq 0$$

The Kuhn-Tucker conditions are

$$6x_{k1} = x_{k1}\,4\,\rho_{11}\,x_{11}+x_{k1}\,4\,\rho_{21}\,x_{21}+x_{k1}\,-2\,\rho_{12}\,x_{12} \tag{35}$$

$$+\ x_{k1}\ -2\,\rho_{22}\,x_{22}+v\,x_{k1}+\eta_1-u_{k1}$$

$$\text{for each } k = 1, 2$$

$$0 = x_{k2}+4\rho_{12}\,x_{12}+x_{k2}+4\rho_{22}\,x_{22} \tag{36}$$

$$+\ x_{k2}-2\rho_{11}\,x_{11}+x_{k2}-2\rho_{21}\,x_{21}+v\,x_{k2}+\eta_2-u_{k2}$$

$$\text{for } k = 1, 2$$

$$\rho_{11}\,x_{11}+\rho_{21}\,x_{21}+\rho_{12}\,\rho_{12}+\rho_{22}\,x_{22}+y = 2 \tag{37}$$

$$\rho_{11}+\rho_{21} = 1 \tag{38}$$

$$\rho_{12}+\rho_{22} = 1 \tag{39}$$

$$\rho_{11}, \rho_{21}, \rho_{12}, \rho_{22} \geqq 0$$

$$y \geqq 0,\ v \geqq 0$$

$$u_{ki}\,\rho_{ki} = 0,\ y\,v = 0$$

$$u_{ki} \geqq 0 \qquad k = 1, 2$$

$$i = 1, 2$$

where y is a slack variable ([8]).

([8]) The slack variable y, which is required to be nonnegative, should be distinguished from an artificial variable, which must be zero during iterations of the algorithm. A slack variable is treated in the same fashion as any other primal variable.

To initiate the algorithm we assume that the extreme points $x_{11} = 0$ and $x_{12} = 0$ are known. Setting $\rho_{11} = 1$, $\rho_{12} = 1$ and $y = 2$ gives us a basic feasible solution to equations (32), (33) and (34). Since we set $u_{11} = 0$ and $u_{12} = 0$ we obtain from equations (35) and (36) two conditions to determine η_1 and η_2. We thus have the following system of equations:

$$
\begin{aligned}
0\rho_{11}+0\rho_{12}-\eta_1+0\eta_2+0y &= 0\\
0\rho_{11}+0\rho_{12}+0\eta_1-1\eta_2+0y &= 0\\
0\rho_{11}+0\rho_{12}+0\eta_1+0\eta_2+1y &= 2\\
1\rho_{11}+0\rho_{12}+0\eta_1+0\eta_2+0y &= 1\\
0\rho_{11}+1\rho_{12}+0\eta_1+0\eta_2+0y &= 1
\end{aligned}
\tag{40}
$$

The matrix of coefficients is

$$
B^0 = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}
$$

and the inverse is

$$
(B^0)^{-1} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}
$$

We denote by $b_0 = \begin{pmatrix} 0 \\ 0 \\ 2 \\ 1 \\ 1 \end{pmatrix}$

as the column of constants associated with the initial system of equations. $(B^0)^{-1} b_0$ gives us the values of the current set of basic variables. In the present case we have

$$
\rho_{11} = 1,\ \rho_{12} = 1,\ \eta_1 = 0,\ \eta_2 = 0,\ y = 2\ .
$$

These values are used in problems (25) and (26) to determine the next ρ_{ki} to enter the basis. Inserting the present solution, we have the following, in problem (25)

$$\text{Max } 6\,x_1$$

$$x_1 \leqq 1 \tag{41}$$

and (26)

$$\text{Max } 0\,x_2$$

$$x_2 \leqq 1 \tag{42}$$

Solving both programming problems the overall maximum is attained by problem (41) at the extreme point $x_1^* = 1$. We designate the latter point as x_{21}. Accordingly ρ_{21} enters the basis. Before introducing ρ_{21} into the basis the condition from (35) containing u_{21} must be adjoined to (40). The added equation is:

$$u_{21} + 0\rho_{11} + 0\rho_{12} - \eta_1 + 0\eta_2 + 0y = -6$$

Denote by $\hat{B}^0$ as the augmented matrix of coefficients for the larger collection of constraints. We have

$$\hat{B}^0 = \begin{bmatrix} 1 & h \\ [0] & B^0 \end{bmatrix}$$

where [0] is a vector of zeros and

$$h = [0\;0\;-1\;0\;0]$$

To obtain $(\hat{B}^0)^{-1}$ note that

$$(\hat{B}^0)^{-1} = \begin{bmatrix} 1 & -h\,B^{0\;-1} \\ [0] & (B^0)^{-1} \end{bmatrix} =$$

$$\begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

In order to determine the relevant tableau elements associated with the activity ρ_{21} we have

$$(\hat{B}^0)^{-1}\begin{pmatrix}-4\\0\\0\\1\\1\\0\end{pmatrix}=\begin{pmatrix}-4\\1\\0\\0\\0\\1\end{pmatrix}$$

This may be summarized in the following abbreviated tableau:

Active Basic Variables	Value of Active Basic variables	Tableau values for ρ_{21}
u_{21}	-6	-4
ρ_{11}	1	1
ρ_{12}	1	0
η_1	0	0
η_2	0	0
y_1	2	1

Taking $\underset{i}{\text{Min}}\left\{\frac{z_i}{l_{ij}} \mid \frac{z_i}{l_{ij}} \geqq 0 \quad l_{ij} \neq 0\right\}$ we have the minimum of $\left\{\frac{-6}{-4}, \frac{1}{1}, \frac{2}{1}\right\}$

Since the minimum is achieved at 1/1 we choose ρ_{11} to leave the basis.

In order to continue the algorithm we must determine the inverse of the new basis where the activity ρ_{21} replaces ρ_{11}. To do this we use the well known theory of basis transformation and construct the matrix E_2^0 (9). E_2^0 is an identity matrix except that the second column consists of elements derived from the tableau representation of ρ_{21}. The matrix E_2^0 is:

$$E_2^0=\begin{bmatrix}1&+4&0&0&0&0\\0&1&0&0&0&0\\0&0&1&0&0&0\\0&0&0&1&6&0\\0&0&0&0&1&0\\0&-1&0&0&0&1\end{bmatrix}$$

(9) See Dantzig [2] for a discussion of this topic.

$$E_2^0(\hat{B}^0)^{-1} = \begin{bmatrix} 1 & -1 & 0 & 0 & +4 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & +1 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & +1 & -1 & 0 \end{bmatrix} = (B^1)^{-1}$$

To determine the values of the current active basic variables we have

$$(B^1)^{-1}\begin{pmatrix} -6 \\ 0 \\ 0 \\ 2 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} -2 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Thus we have $u_{21} = -2$, $\rho_{21} = 1$, $\rho_{12} = 1$, $\eta_1 = 0$, $\eta_2 = 0$ and $y = 1$. According to Step 3, u_{11} is introduced into the basis to replace ρ_{11}. The tableau elements for u_{11} are determined from

$$(B^1)^{-1}\begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} -1 \\ 0 \\ 0 \\ -1 \\ 0 \\ 0 \end{pmatrix}$$

Using the earlier representation we have:

Active Basic Variables	Value of Active Basic variables	Tableau values for u_{11}
u_{21}	-2	-1
ρ_{21}	1	0
ρ_{12}	1	0
η_1	0	-1
η_2	0	0
y	1	0

It can be seen immediately that u_{11} replaces u_{21} in the basis, and we return to Step 1 in the algorithm. Next we compute the elementary matrix

$$E_1^1 = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$(B^2)^{-1} = E_1^2(B^1)^{-1}$$

and

$$(B^2)^{-1} = \begin{bmatrix} -1 & +1 & 0 & 0 & -4 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 & -4 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & +1 & -1 & 0 \end{bmatrix}$$

To compute the current values of the active basic variables we form the expression

$$(B^2)^{-1}\, b^0$$

where

$$b^0 = \begin{pmatrix} -6 \\ 0 \\ 0 \\ 2 \\ 1 \\ 1 \end{pmatrix}$$

This gives us $u_{11} = +2$, $\rho_{21} = +1$, $\rho_{12} = +1$, $\eta_1 = +2$, $\eta_2 = 0$ and $y = 1$. Substituting there values into problem (25) we have

$$\text{Max } 2x_1 - 2 \tag{43}$$

$$x_1 \leqq 2$$

$$x_1 \geqq 0$$

and into problem (26) we have

$$\text{Max } 2x_2 \tag{44}$$

$$x_2 \leqq 1$$

$$x_2 \geqq 0$$

The overall maximum is achieved by problem (44) at $x_2 = 1$. We let $x_{22} = 1$ be that extreme point associated with the variable ρ_{22}. According to Step 2 the variable ρ_{22} is chosen to come into the basis. Next we adjoin to the set of constraints the equation from (36) containing u_{22}. This constraint is of the form

$$u_{22}+2\rho_{21}+0\rho_{12}+0\eta_1-\eta_2+0y = 0$$

Therefore

$$h = [0 \ +2 \ 0 \ 0 \ -1 \ 0]$$

and

$$-h(B^2)^{-1} = (0 \ 0 \ -1 \ 0 \ -2 \ 0)$$

The current augmented inverse is

$$(\hat{B}^2)^{-1} = \begin{bmatrix} 1 & 0 & 0 & -1 & 0 & -2 & 0 \\ 0 & & & & & & \\ 0 & & & & & & \\ 0 & & & (B^2)^{-1} & & & \\ 0 & & & & & & \\ 0 & & & & & & \\ 0 & & & & & & \end{bmatrix}$$

To determine the tableau elements for ρ_{22} we have

$$(\hat{B}^2)^{-1}\begin{Bmatrix} -4 \\ +2 \\ 0 \\ 0 \\ 1 \\ 0 \\ 1 \end{Bmatrix} = \begin{Bmatrix} -4 \\ -2 \\ 0 \\ 1 \\ -2 \\ 0 \\ +1 \end{Bmatrix}$$

and to determine the values of the current solution we have

$$(\hat{B}^2)^{-1}\begin{bmatrix} 0 \\ -6 \\ 0 \\ 0 \\ 2 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} -2 \\ +2 \\ 1 \\ 1 \\ 2 \\ 0 \\ 1 \end{bmatrix}$$

Sumarizing the information we have the following tableau:

Active Basic Variables	Value of Active Basic variables	Tableau values for ρ_{22}
u_{22}	-2	-4
u_{11}	$+2$	-2
ρ_{21}	1	0
ρ_{12}	1	1
η_1	2	-2
η_2	0	0
y	1	$+1$

Clearly, u_{22} leaves the basis and again we return to Step 1. We obtain a new inverse which contains the column of coefficients from ρ_{22} replacing the coefficients associated with u_{22}. This new solution is

$$\rho_{22} = \tfrac{1}{2},\ \rho_{21} = 1,\ \rho_{12} = \tfrac{1}{2},\ y = \tfrac{1}{2},\ \eta_1 = 3 \text{ and } \eta_2 = 0.$$

Substituting these values into problems (25) and (26) we have

$$\text{Max } 2x_1 - 3$$
$$x_1 \leqq 1$$
$$x_1 \geqq 0$$

and

$$\text{Max } 0$$
$$x_2 \leqq 1$$
$$x_2 \geqq 0$$

Both problems have nonpositive solutions which indicate that the minimum over all u_{ki} variables is nonnegative. Therefore we are at an optimal solution. In terms of the original variables the solution is

$$x_1 = 1(1) = 1$$

$$x_2 = \tfrac{1}{2}(0)+\tfrac{1}{2}(1) = \tfrac{1}{2}$$

CONCLUDING REMARKS

This paper has presented one possible approach to solving a convex quadratic programming problem where the number of constraints is large, but of a special structure. In a problem with a large number of constraints a direct approach may be impossible since the size of the initial basis matrix would cause difficulty in determining its inverse. While in the present algorithm the dimensionality may increase one may always obtain, at worst, approximate solutions.

REFERENCES

[1] DANTZIG, G.B., "General Convex Objective Forms", *Mathematical Methods in the Social Sciences 1959*, ed., Arrow, Karlin and Suppes, Stanford University Press, 1960.

[2] DANTZIG, G.B., *Linear Programming and Extensions*, Princeton University Press, Princeton, New Jersey.

[3] DANTZIG, G.B. and P. WOLFE, "The decomposition Algorithm for Linear Programs", *Econometrica*, October 1961.

[4] DANTZIG, G.B. and P. WOLFE, "Decomposition Principle for Linear Programs", *Operations Research*, Vol. 8, 1960.

[5] KUHN, H.W. and A.W. TUCKER, "Nonlinear Programming", *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, University of California, 1950 pp. 481-492.

[6] VAN DE PANNE, C., and A. WHINSTON, "The Simplex and the Dual Method for Quadratic Programming", Econometric Institute, Rotterdam, April 1963 and forthcoming in *Operational Research Quaterly.*

[7] ROSEN, J.B., "Convex Partition Programming", *Recent Advances in Mathematical Programming*, ed., Graves and Wolfe, McGraw-Hill, New York, 1963.

[8] WHINSTON, A., "Price Guides in Decentralized Organizations," *New Perspectives in Organization Research*, ed., Cooper, Leavitt and Shelly, John Wiley and Sons, New York, forthcoming.

[9] WHINSTON, A., "Theoretical and Computational Problems in Organizational Decision Making", forhtcoming in the proceedings of the conference on Operations Research and the Social Sciences.