Cowles Foundation Paper 222
Reprinted from Cahiers, 6(4), 1964

A DUAL DECOMPOSITION ALGORITHM FOR QUADRATIC PROGRAMMING[1]

Andrew WHINSTON
University of Virginia

INTRODUCTION

In this paper we present a decomposition type algorithm for a convex quadratic programming problem. The method presented here will be based on a quadratic programming algorithm developed by van de Panne and Whinston [4]. This method can be considered as a direct extension of the dual method for linear programming and for this reason we refer to the decomposition algorithm as a dual method.

In an earlier paper A. Whinston [5] presented a decomposition algorithm for quadratic programming which can be viewed as an extension of a method for linear programming by Dantzig and Wolfe [1]. These methods could be referred to as " primal" methods since the constraints of the problem are always satisfied during the search for an optimal solution. On the other hand, the method to be discussed in this paper maintains at every iteration, an optimal solution which however, may violate the constraints. Thus the algorithm searches for a feasible solution while preserving optimality at every step[2].

Decomposition algorithms are useful when the number of constraints of a problem is large, but possessing a special block diagonal form. With a large number of constraints it may be that a general purpose algorithm could not be used.

THE ALGORITHM

In presenting the algorithm we shall attempt to illustrate the

---

[1] This paper is an outgrowth of some joint work with Mr. C. van de Panne as reported in [4]. I am, of course, very much indebted to him. My colleague, Menahem Yaari and Bela Martos of the Hungarian Academy of Sciences made several very helpful suggestions. I remain responsible for all possible errors.

[2] In a recent paper by Abadie and Williams [6], a dual decomposition algorithm for linear programming was developed. The method developed here can be viewed as an extension of their approach to quadratic programming.

basic differences between this case and the linear programming problem. With this in mind, we consider only two subsectors in the following problem

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

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

$$A_1 x_1 + A_2 x_2 = b_0 , \tag{1}$$

$$B_1 x_1 \leqslant b_1 , \tag{2}$$

$$B_2 x_2 \leqslant b_2 , \tag{3}$$

$$x_1 \geqslant 0, \qquad x_2 \geqslant 0;$$

$$x_1' = (\overset{1}{x}_1 \dots \overset{q_1}{x}_1), \qquad x_2' = (\overset{1}{x}_2 \dots \overset{q_2}{x}_2), \qquad b_0' = (b_{01} \dots b_{0l}),$$

$$p_1' = (\overset{1}{p}_1 \dots \overset{q_1}{p}_1), \qquad p_2' = (\overset{1}{p}_2 \dots \overset{q_2}{p}_2).$$

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[1].

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)

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

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

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

---

[1] See Abadie and Williams [6] where the more general argument for linear programming is developed. Exactly the same approach would be applicable in this case.

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

$$\operatorname*{Max}_{\rho_{ki}} p_1' \Sigma \rho_{k1} x_{k1} + p_2' \Sigma \rho_{k2} x_{k2} - 1/2 \Sigma \rho_{k1} x_{k1}' C_{11} \Sigma \rho_{k1} x_{k1} -$$

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

(6) $$A_1 \Sigma \rho_{k1} x_{k1} + A_2 \Sigma \rho_{k2} x_{k2} + I y_1 = b_0,$$

(7) $$\Sigma \rho_{k1} + y_2 = 1,\ \Sigma \rho_{k2} + y_3 = 1, \text{ where } y_1 = \begin{pmatrix} y_{11} \\ \vdots \\ y_{1l} \end{pmatrix},$$

(8) $$\rho_{ki} \geqslant 0,\quad y_{1j} = 0,\quad y_2 = 0,\quad y_3 = 0, \text{ all } k, i = 1, 2,\ j = 1, \ldots l.$$

The Kuhn-Tucker [2] conditions for this problem are (6), (7), (8) and

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

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

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

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

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

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

Note that η_1 is associated with y_2, and η_2 with y_3.

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

$$f(\lambda(\hat{x}_1, \hat{x}_2) + (1-\lambda)(\tilde{x}_1, \tilde{x}_2)) \geqq \lambda f(\hat{x}_1, \hat{x}_2) + (1-\lambda) f(\tilde{x}_1, \tilde{x}_2),$$

where $(\hat{x}_1, \hat{x}_2)$ and $(\tilde{x}_1, \tilde{x}_2)$ are any two points satisfying the constraints of the problem. We are given that

$$(x_1, x_2) = \mathrm{T}(\rho_1, \rho_2),$$

where T is a linear transformation.

Define $g(\rho) = f(\mathrm{T}\rho)$. Then

$$\begin{aligned} g(\lambda\hat{\rho} + (1-\lambda)\tilde{\rho}) &= f(\mathrm{T}(\lambda\hat{\rho} + (1-\lambda)\tilde{\rho})) \\ &= f(\lambda\mathrm{T}\hat{\rho} + (1-\lambda)\mathrm{T}\tilde{\rho}) \\ &\geqq \lambda f(\mathrm{T}\hat{\rho}) + (1-\lambda) f(\mathrm{T}\tilde{\rho}) \\ &= \lambda g(\hat{\rho}) + (1-\lambda) g(\tilde{\rho}). \end{aligned}$$

q.e.d.

DESCRIPTION OF THE ALGORITHM

An important aspect of any decomposition algorithm, as pointed out by Dantzig and Wolfe, is the sequential generation of the extreme points as they are required in the algorithm. We shall indicate specifically how the approach, discussed in this paper, accomplishes this : First we give a general description of the algorithm and its rules.

To initiate the algorithm we set $y_1 = b_0$, $y_2 = 1$, $y_3 = 1$, $\rho_{kt} = 0$, $v = 0$, $\eta_1, \eta_2 = 0$. Thus all u_{kt} are in the basis, set equal to $-p_i x_{ki}$. Of course the $\{x_{ki}\}$ are not known but we must insure that conditions (11) and (14) are satisfied. This will evidently be so if all components of the vectors p_1 and p_2 are nonpositive.

However if the latter is not the case then we follow a special procedure in order to arrive at an initial solution satisfying conditions (11) and (14). We introduce into the set of constraints (6) the added equation

$$\sum_{k,t} \rho_{kt} \leqq \mathrm{M},$$

where M is chosen large enough so that no possible solution value is excluded. Associated with this constraint is a dual variable v_m. The introduction of the extra equation adds one more row and one more column to the quadratic tableau. Next we determine the largest possible value among the components of P_1 and P_2 where $P_1 = (p'_1 x_{11} \dots p'_2 x_{k_1 1})$.

*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	$-\mathbf{P}'_1$	I	0	$-\bar{A}'_1$	$-\bar{C}_{11}$	$-\bar{C}_{12}$	$-l'_1$	0	0	0	0
u'_2	$-\mathbf{P}'_2$	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

$u_1 = (u_{k1} \dots u_{k_1 1})$,
$u_2 = (u_{k2} \dots u_{k_2 2})$,
$v = (v_1 \dots v_l)$,
$\bar{A}_1 = (A_1 x_{11} \dots A_1 x_{k_1 1})$,
$\bar{A}_2 = (A_2 x_{12} \dots A_2 x_{k_2 2})$,
$l_1 = (1 \dots 1_{k_1})$,
$l_2 = (1 \dots 1_{k_2})$.

$\rho_1 = (\rho_{11} \dots \rho_{k_1 1})$,
$\rho_2 = (\rho_{12} \dots \rho_{k_2 2})$,

$P_1 = (p'_1 x_{11} \dots p'_1 x_{k_1 1})$,
$P_2 = (p'_2 x_{21} \dots p'_2 x_{k_2 2})$,

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

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

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

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

---

* 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.

and $P_2 = (p'_2 x_{21} \ldots p'_2 x_{k_2 2})$. This can be accomplished by solving the following linear programming problems

$$\begin{array}{rlcrl} \operatorname{Max} p'_1 x_1, & & \qquad & \operatorname{Max} p'_2 x_2, & \\ B_1 x_1 & \leqq b_1, & & B_2 x_2 & \leqq b_2, \\ x_1 & \geqq 0; & & x_2 & \geqq 0. \end{array}$$

Of the two solutions we choose that one which has the largest value (if they both have the same value either one can be chosen). The solution is attained at an extreme point which we designate as $x_{k^*i^*}$. Associated with the extreme point $x_{k^*i^*}$ is the variable $\rho_{k^*i^*}$ and its complimentary variable $u_{k^*i^*}$. We introduce v_m into the basis and remove $u_{k^*i^*}$. The new solution will be a dual feasible one since all $u_{ki} \geqq 0$ [1].

In general, the initial solution will not satisfy the conditions in (8) but will do so in (10), (11), (13) and (14). (Naturally conditions (6), (7), (9) and (12) will always be satisfied). When conditions (11) and (14) are satisfied then we say the solution is dual feasible. When conditions (10) and (13) are satisfied we have a standard form tableau; otherwise the tableau is referred to as nonstandard. The initial solution constitutes a dual feasible standard form tableau. The algorithm moves from a dual feasible solution, which is either standard or nonstandard form, to a primal feasible solution, i.e., a solution satisfying the conditions in (8).

Before giving the rules of the algorithm we wish to make certain preliminary remarks. The pairs (ρ_{ki}, u_{ki}), (v_i, y_{1i}), (η_1, y_2) and (η_2, y_3) are referred to as complimentary variables. Let $\{z_i\}$ be the set of variables which are permitted to leave the basis in a particular iteration and $\{l_{ij}\}$ be the elements in the j-th column at the tableau where we assume that this is the column of the variable to come into the basis. Then the variable to leave the basis is that z_i for which

$$\operatorname*{Min}_{i} \left\{ k\frac{z_i}{l_{ij}} \,\middle|\, k\frac{z_i}{l_{ij}} \geqq 0, l_{ij} \neq 0 \right\}$$

is achieved where $k = 1$ if the new variable is introduced in a positive amount and $k = -1$ if it is introduced in a negative amount. The algorithm proceeds in the following manner [2] :

---

[1] See [4] for a complete discussion of this type of starting solution.

[2] These rules are based on [4].

1. Among the set of ρ_{ki} and $-|y|$ variables we choose the most negative variable. If there are none the algorithm is terminated.
2. We introduce into the basis the complimentary u_{ki}, v_i or η_i variable to the one chosen in step 1 with opposite sign of its compliment. For the variable to be eliminated from the basis we choose from among the u_{ki} variables in the basis and the variable designated in step 1. If the variable designated in step 1 is removed we return to step 1, if not we go to step 3.
3. We introduce the ρ_{ki} variable corresponding to the u_{ki} variable which has just left the basis. The variable to be eliminated is chosen from the u_{ki} variables in the basis and the variable designated in step 1.

According to the rules of the algorithm let us assume that we have chosen the variable to come into the basis. Next we consider the question of determining which variable should leave the basis and in particular which, if any, of the u_{ki} variables in the basis should be removed.

Let u^*_{ki} indicate the current values of the basic u_{ki} variables. Then the relevant information needed is an expression for

$$\frac{u^*_{ki}}{l_{ki,j}},$$

where $l_{ki,j}$ is the tableau element in the j-th column, the column associated with the vector designated to come into the basis, opposite the u_{ki} variable.

At any iteration the u_{ki} variable is a linear function of the extreme point x_{ki}. Consider then the determination of an expression for $l_{ki,j}$. We designate B as the matrix of coefficients for the variables in the current basis and π_j as the column of coefficients of the variable that is to enter the basis. Then we have

$$\mathrm{B}l = \pi_j$$

to determine the vector l of tableau elements. We partition $\mathrm{B} = (\mathrm{B}^1, \mathrm{B}^2)$ so that B^1 consists of the coefficients associated with the u_{ki} variables in the basis, while B^2 contains the coefficients for the variables ρ_{ki}, v_i, η_i and y_i which are in the basis. Then we write

$$(\mathrm{B}^1, \mathrm{B}^2)\begin{pmatrix} l_1 \\ l_2 \end{pmatrix} = \pi_j,$$

where l_1 is the vector containing the $l_{ki,j}$ elements. By permuting the rows of B we have

$$\begin{pmatrix} \mathrm{I} & \mathrm{B}_{12} \\ 0 & \mathrm{B}_{22} \end{pmatrix} \begin{pmatrix} l_1 \\ l_2 \end{pmatrix} = \begin{pmatrix} \pi_{j1} \\ \pi_{j2} \end{pmatrix},$$

where $$\mathrm{B}^1 = \begin{pmatrix} \mathrm{I} \\ 0 \end{pmatrix} \text{ and } \mathrm{B}^2 = \begin{pmatrix} \mathrm{B}_{12} \\ \mathrm{B}_{22} \end{pmatrix}.$$

The elements in B_{22} are all known. Accordingly, since B is nonsingular, B_{22} must also be a nonsingular matrix. Then we have

$$l_2 = \mathrm{B}_{22}^{-1}\pi_{j2}.$$

The vector l_2 gives the tableau elements opposite the ρ_{ki}, v_i, y_i and η_i variables in the basis. For the vector l_1 we have the expression

$$l_1 = \pi_{j1} - \mathrm{B}_{12} l_2.$$

Let us consider a particular u_{ki}, say u_{k1}, which is in the basis. Then we have

$$l_{k1,j} = \pi_{j1,j} - (-\tilde{v}'\mathrm{A}x_{k1} - x_{k1}\mathrm{C}_{11}\Sigma\tilde{\rho}_{k1}x_{k1} - x_{k2}\mathrm{C}_{12}\Sigma\tilde{\rho}_{k2}x_{k2} - \tilde{\eta}_1),$$

where $\tilde{v}$, $\tilde{\rho}_{k1}$, $\tilde{\rho}_{k2}$, and $\tilde{\eta}_1$ compose the vector of tableau elements in l_2. The expression for $l_{k2,j}$ is computed in a similar fashion. The component $\pi_{j1,j}$ is either a constant or a linear function of the extreme point x_{k1}, depending on which vector has been chosen to enter the basis. With this information we must compute

$$\left\{\min_k k\frac{u_{k1}}{l_{k1,j}} \,\middle|\, k\frac{u_{k1}}{l_{k1,j}} \geqq 0,\ l_{k1,j} \neq 0\right\},$$

and

$$\left\{\min_k k\frac{u_{k2}}{l_{k2,j}} \,\middle|\, k\frac{u_{k2}}{l_{k2,j}} \geqq 0,\ l_{k2,j} \neq 0\right\}.$$

Each problem determines the minimum of the ratio of two linear functions of the extreme points. If we can show that the minimum of such a linear fractional function, if it is achieved, will occur at an extreme point, then our problem will be immensely simplified. For example consider the minimum of

$$\frac{-p_1'x_1 + x_1\mathrm{C}_{11}\Sigma\rho_{k1}x_{k1} + x_1\mathrm{C}_{12}\Sigma\rho_{k2}x_{k2} + v'\mathrm{A}_1x_1 + \eta_1}{\pi_{j1,j} + \tilde{v}'\mathrm{A}x_1 + x_1\mathrm{C}_{11}\Sigma\tilde{\rho}_{k1}x_{k1} + x_1\mathrm{C}_{12}\Sigma\tilde{\rho}_{k2}x_{k2} + \tilde{\eta}_1},$$

$$\text{s.t.} \quad \mathrm{B}_1x_1 \leqq b_1,$$

$$x_1 \geqq 0.$$

where we assume that $\bar{k} = +1$. Note that the numerator is always nonnegative, and we require that the denominator be also nonnegative. Methods for finding the minimum of a linear fractional function, if a minimum exists under the condition that the denominator be nonnegative, have been presented by B. Martos [3] [1]. He has also shown that the minimum will occur at an extreme point. Thus we see that by solving two fractional programming problems we may determine the vector to leave the basis and its associated extreme point.

With the determination of a u_{ki} to leave the basis we obtain the value of the associated extreme point x_{ki}. This provides us with the value of the simplex pivot value. All other steps and procedures of the algorithm can be carried out without further discussion. Since there are only a finite number of extreme points in each of the subproblems the argument in [4] can be used to prove convergence.

EXAMPLE

We present in this section an example in order to illustrate the way in which the algorithm works. For this purpose several iterations will be carried out.

Consider the problem :

$$\text{Max } -6x_1-2x_1^2+2x_1x_2-2x_2^2,$$

$$x_1+x_2 = 2, \tag{15}$$

$$0 \leqq x_1 \leqq 2, \tag{16}$$

$$0 \leqq x_2 \leqq 1. \tag{17}$$

where (15) is the interconnecting constraint, and (16) and (17) represent 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 (16) and (17) contains two extreme points. The method presented here, however, does not depend on this information being available.

---

[1] Abadie and Williams [6], have also discussed this problem and have presented a method of solution closely related to [3].

In order to insure that the denominator is nonnegative we may introduce this condition as an explicit constraint. In the case where the constraint is binding the fraction is $+\infty$. When the result is finite the constraint will not be binding, so that an extreme point relative to the original system of constraint is obtained.

Let

$$\{x_{11}, x_{21}\} \tag{18}$$

be the set of extreme points for the convex set represented by (16) where

$$\begin{aligned} x_1 &= \rho_{11} x_{11} + \rho_{21} x_{21}, \\ \rho_{11} + \rho_{21} &= 1, \\ \rho_{11}, \rho_{21} &\geqq 0. \end{aligned} \tag{19}$$

and for (17) we have

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

and

$$\begin{aligned} x_2 &= \rho_{12} x_{12} + \rho_{22} x_{22}, \\ \rho_{12} + \rho_{22} &= 1, \\ \rho_{12} \geqq 0, &\ \rho_{22} \geqq 0. \end{aligned} \tag{20}$$

We have the transformed problem

$$\begin{aligned} \text{Max } & -6(\rho_{11} x_{11} + \rho_{21} x_{21}) - 2(\rho_{11} x_{11} + \rho_{21} x_{21})^2 \\ & + 2(\rho_{11} x_{11} + \rho_{21} x_{21})(\rho_{12} x_{12} + \rho_{22} x_{22}) \\ & - 2(\rho_{12} x_{12} + \rho_{22} x_{22})^2, \end{aligned}$$

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

$$\rho_{11} + \rho_{21} = 1, \tag{22}$$

$$\rho_{12} + \rho_{22} = 1, \tag{23}$$

$$\begin{aligned} \rho_{11}, \rho_{21}, \rho_{12}, \rho_{22} &\geqq 0, \\ y &= 0. \end{aligned}$$

Note that we have introduced the artificial variable "y" into (21).

The Kuhn-Tucker conditions are

$$\begin{aligned} & -4 x_{k1}(\rho_{11} x_{11} + \rho_{21} x_{21}) + 2 x_{k1}(\rho_{12} x_{12} + \rho_{22} x_{22}) \\ & - v x_{k1} - \eta_1 + u_{k1} = 6 x_{k1}, \end{aligned} \tag{24}$$

for $k = 1, 2$.

$$\begin{aligned} & +2 x_{k2}(\rho_{11} x_{11} + \rho_{21} x_{21}) - 4 x_{k2}(\rho_{12} x_{12} + \rho_{22} x_{22}) \\ & - v x_{k2} - \eta_2 + u_{k2} = 0, \end{aligned} \tag{25}$$

for $k = 1, 2$.

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 an initial solution to equations (21), (22) and (23). Since we set $u_{11} = 0$ and $u_{12} = 0$ we obtain from equations (24) and (25) two conditions to determine η_1 and η_2. We thus have the following system of equations :

$$(26)\qquad \begin{cases} 0\rho_{11}+0\rho_{12}-\ \eta_1+0\eta_2+0y=0, \\ 0\rho_{11}+0\rho_{12}+0\eta_1-\ \eta_2+0y=0, \\ 0\rho_{11}+0\rho_{12}+0\eta_1+0\eta_2+\ y=2, \\ 1\rho_{11}+0\rho_{12}+0\eta_1+0\eta_2+0y=1, \\ 0\rho_{11}+1\rho_{12}+0\eta_1+0\eta_2+0y=1. \end{cases}$$

The solution is

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

According to step two of the algorithm we introduce the variable v to the basis in a negative amount since its compliment y is positive.

From (26) we have the following matrix of coefficients for the current basis

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

The inverse is

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

To determine the tableau elements for the "v" variable entering the basis we have

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

In order to determine which variable will leave the basis among $u_{ki}>0$ we solve the following fractional programming problems as stated in section three :

$$(27)\qquad \begin{aligned} &\text{Min}\ \frac{6x_1}{x_1}, \\ &0\leq x_1\leq 2; \end{aligned}$$

and

$$(28) \qquad \operatorname{Min} \frac{0}{x_2}, \qquad 0 \leqq x_2 \leqq 1.$$

and choose the minimum solution of the two problems. This is achieved at $x_2 = 0$ or $x_2 = 1$. We choose $x_2 = 1$ and designate this extreme point as x_{22}.

In order to determine the new basis matrix the variable u_{22} must be explicitly introduced into the present contracted basis B^0. Thus we adjoin the equation

$$u_{22} + 0\rho_{11} + 0\rho_{12} - 0\eta_1 - \eta_2 + 0y = 0$$

to the system (26). 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] represents a vector of zeros and

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

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

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

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

In order to continue the algorithm we must determine the inverse of the new basis where the activity v replaces u_{22}. First we must compute the tableau values for v in the terms of the current enlarged basis $\hat{B}^0$. We have

$$(\hat{B}_0)^{-1} \begin{pmatrix} -1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} -1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Next use the theory of basis transformation and construct the matrix E_1^0. E_1^0 is an identity matrix except that the first column consists of elements derived from the tableau representation of v. The matrix E_1^0 is :

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

We have

$$E_1^0(\hat{B}^0)^{-1} = \begin{bmatrix} -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} = (B^1)^{-1},$$

According to rule three of the algorithm the variable ρ_{22} comes into the basis. To obtain the current tableau values for the activity ρ_{22} we have

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

Based on the current information we may construct the following abbreviated tableau :

Basic variables	Value of Basic variables	Tableau values for ρ_{22}
v	0	+4
ρ_{11}	1	0
ρ_{12}	1	1
η_1	0	0
η_2	0	0
y	2	1

To carry out step three of the algorithm we must determine the u_{ki} variable, if any, to leave the basis. We obtain the following two programming problems :

$$\text{(29)} \qquad \text{Min } \frac{6x_1}{2x_1+4x_1}, \quad 0 \leqq x_1 \leqq 2;$$

and

$$\text{(30)} \qquad \text{Min } \frac{0}{4x_1-4x_2} = \frac{0}{0}, \quad 0 \leqq x_2 \leqq 1.$$

Problem (30) has no solution indicating that there are no u_{k2} variables in the basis. Problem (29) achieves a minimum at the extreme point $x_1 = 2$ and we designate it as x_{21}. We have as the ratio 1/1 which compared with 2/1 for the y variable indicates that u_{21} will leave the basis. Again employing rule three ρ_{21} will be introduced into the basis.

We continue in the above fashion until a dual feasible solution is attained. The final solution is $x_1 = 1$, $x_2 = 1$.

REFERENCES

[1] Dantzig, G.B. and P. Wolfe, "The Decomposition Algorithm for Linear Programs", *Econometrica*, October 1961.

[2] 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.

[3] Martos, B., "Hyperbolic Programming", *Publication of the Mathematical Institute of the Hungarian Academy of Science*, 1960.

[4] 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 Quarterly.*

[5] Whinston, A., "A Decomposition Algorithm for Quadratic Programming", Cowles Foundation Discussion Paper No. 172.

[6] Abadie, J. and A.C. Williams, "Dual and Parametric Methods in Decomposition", in *Recent Advances in Mathematical Programming*, Graves and Wolfe ed., McGraw-Hill 1963