1968

Solution of linear programs- an algorithm

Raymond L. Somers
Lehigh University

SOLUTION OF LINEAR PROGRAMS - AN ALGORITHM

by
Raymond L. Somers

A THESIS

Presented to the Graduate Committee

of Lehigh University

in Candidacy for the Degree of

Master of Science

in

Mathematics

Lehigh University

1968

This thesis is accepted and approved in partial fulfillment of the requirements for the degree of Master of Science.

May 15, 1968
(date)

Gerhard Rayna
Professor in charge

Everett Fitcher
Head of the Department

ACKNOWLEDGEMENT

I wish to thank Dr. Gerhard Rayna for his valuable assistance in the preparation of this thesis.

Table of Contents

ABSTRACT

A linear programming problem, of minimizing a linear combination of variables subject to a linear set of inequalities, is presented. A closely related maximization problem is then given and the relationship between the two problems is investigated. The pair of linear programs is re-expressed as a pair of linear systems of equations which are jointly represented in the form of a schema of numbers. A pivot transformation which results in a re-expression of the pair of linear systems in terms of a different set of independent and dependent variables is defined and the resulting change in the schema of numbers is given.

An algorithm is given for the solution of the pair of linear programs. This is a variation of a method presented by Balinski and Gomory. A hierarchy of levels of subschemata for the schema is defined and with each level subschema a corresponding goal is associated. The level 1 subschema is examined to see if goal 1 can be accomplished. If it can, a pivot transformation is performed, accomplishing this goal. The hierarchy of levels of subschemata is then redetermined and the new level 1 subschema is examined to see if goal 1 can be accomplished. If at any point goal 1 cannot be accomplished, the lowest k for which goal k can be accomplished is determined. When that k is determined, a pivot transformation is performed, accomplishing goal k.

After the transformation has been performed, the hierarchy of subschemata is redetermined and the level 1 subschema is examined as before to see if goal 1 can be accomplished. In a finite number of steps the algorithm results in a schema from which the solutions to our pair of linear programming problems, if they exist, can be easily read, or from which it can be determined that no solutions exist.

FORMULATION OF PROBLEM

Consider the linear programming problem:

$$\text{Minimize } \sum_{j=1}^{n} c_j \cdot y_j + d$$

$$\text{subject to } \sum_{j=1}^{n} a_{ij} \cdot y_j + b_i \leq 0, \quad i = 1, \ldots, m$$

$$\text{and} \qquad y_j \geq 0, \quad j = 1, \ldots, n$$

where the a_{ij}, c_j, b_i, and d are known constants.

This is very closely related to the following problem:

$$\text{Maximize } \sum_{i=1}^{m} b_i \cdot x_{n+i} + d$$

$$\text{subject to } \sum_{i=1}^{m} a_{ij} \cdot x_{n+i} + c_j \geq 0, \quad j = 1, \ldots, n$$

$$\text{and} \qquad x_{n+i} \geq 0, \quad i = 1, \ldots, m$$

where the a_{ij}, c_j, b_i, and d are the same constants as above.

We shall consider these problems simultaneously and investigate the relationship between the two.

The two problems are conveniently represented by the following schema. **(The labels are not considered part of the schema.)**

$$\begin{array}{c|ccc|c|l}
 & y_1 & \cdots & y_n & 1 & \\
\hline
x_{n+1} & a_{11} & \cdots & a_{1n} & b_1 & = -y_{n+1} \\
\vdots & \vdots & & \vdots & \vdots & \vdots \\
x_{n+m} & a_{m1} & \cdots & a_{mn} & b_m & = -y_{n+m} \\
\hline
1 & c_1 & \cdots & c_n & d & = v \\
\hline
 & =x_1 & \cdots & =x_n & =u &
\end{array} \qquad (\text{I})$$

where we require all y_i and $x_i \geq 0$, $i = 1, \ldots, m+n$, and we are to minimize v and maximize u.

PIVOT TRANSFORMATIONS

As indicated above, the schema

$$\begin{array}{c|ccccc|c}
 & \cdots & y^* & \cdots & y & \cdots & \\
\hline
\vdots & & \vdots & & \vdots & & \\
x^* & \cdots & a & \cdots & b & \cdots & = -w^* \\
\vdots & & \vdots & & \vdots & & \\
x & \cdots & c & \cdots & d & \cdots & = -w \\
\vdots & & \vdots & & \vdots & & \\
\hline
 & \cdots & = z^* & \cdots & = z & \cdots &
\end{array}$$

conveniently exhibits two systems of linear equations, a row system:

$$\begin{array}{ccccccccc}
 & & \vdots & & & & \vdots & & \\
\cdots & + & ay^* & + & \cdots & + & by & + \cdots & = -w^* \\
 & & \vdots & & & & \vdots & & \\
\cdots & + & cy^* & + & \cdots & + & dy & + \cdots & = -w \\
 & & \vdots & & & & \vdots & &
\end{array}$$

and a column system:

$$\begin{array}{ccccccccc}
 & & \vdots & & & & \vdots & & \\
\cdots & + & ax^* & + & \cdots & + & cx & + \cdots & = z^* \\
 & & \vdots & & & & \vdots & & \\
\cdots & + & bx^* & + & \cdots & + & dx & + \cdots & = z \\
 & & \vdots & & & & \vdots & &
\end{array}$$

If $a \neq 0$, we can re-express each of the systems by solving the $-w^*$ equation in the row system for y^* and the z^* equation in the column system for x^*, and then use these new equations to eliminate y^* from the other row equations and x^* from the other column equations.

Solving for y^* in the row system, we have

$$\cdots + y^* + \cdots + a^{-1}by + \cdots = -a^{-1}w^*$$

or

$$\cdots + a^{-1}w^* + \cdots + a^{-1}by + \cdots = -y^*$$

Substituting we have

$$\ldots\ c(\ldots -a^{-1}w^{*} + \ldots -a^{-1}by + \ldots) + \ldots + dy + \ldots = -w$$

or

$$\ldots\ -ca^{-1}w^{*} + \ldots (d - ca^{-1}b)y + \ldots = -w.$$

If we solve for x^{*} in the column system we have

$$\ldots + a^{-1}z^{*} + \ldots - a^{-1}cx + \ldots = x^{*}.$$

Substitution then gives

$$\ldots\ +ba^{-1}z^{*} + \ldots (d - ba^{-1}c)x + \ldots = z.$$

This result can be expressed in the following schema:

	...w^{*}	...	y	...	
z^{*}	a^{-1}		ba^{-1}	...	$= -y^{*}$
x	$-ca^{-1}$	...	$d-ca^{-1}b$	...	$= -w$
	$=x^{*}$		$=z$		

We call "a" a "pivot" and the operation of interchanging dependent and independent variables as described above a "pivot transformation". Clearly, since the only result of performing a pivot transformation is to re-express a pair of linear systems in terms of different sets of independent and dependent variables, the resulting schema is equivalent to the original.

Our plan is to define a finite sequence of pivot transformations beginning with schema (I) and terminating with a schema from which the solution to our linear programming problem, if it exists, can be easily read, or from which we can tell that a solution does not exist.

PRELIMINARIES

Definition: A set $\{y_1, \ldots, y_{m+n}\}$ which satisfies the row equations in schema (I) above with all $y_i \geq 0$ is said to be a row feasible set. Similarly, $\{x_1, \ldots, x_{m+n}\}$ which satisfies the column equations with all $x_i \geq 0$ is said to be a column feasible set.

Theorem 1: If $\{y_1, \ldots, y_{m+n}\}$ (and the associated value of v) is a row feasible set and if $\{x_1, \ldots, x_{m+n}\}$ (and the associated value of u) is a column feasible set, then $\sum_{i=1}^{m+n} x_i y_i = v - u$.

Proof:
$$\sum_{i=1}^{m+n} x_i y_i = \sum_{i=1}^{n} x_i y_i + \sum_{i=n+1}^{m+n} x_i y_i$$
$$= \sum_{i=1}^{n} y_i \Big(\sum_{j=1}^{m} a_{ji} x_{n+j} + c_i \Big) - \sum_{i=n+1}^{m+n} x_i \Big(\sum_{j=1}^{n} a_{i-n,j} y_j + b_{i-n} \Big)$$
$$= v + \sum_{i=1}^{n} y_i \sum_{j=1}^{m} a_{ji} x_{n+j} - u - \sum_{i=n+1}^{m+n} x_i \sum_{j=1}^{n} a_{i-n,j} y_j$$
$$= v - u$$

Corollary: If $\{y_1, \ldots, y_{m+n}\}$ (and the associated value of v) is a row feasible set and if $\{x_1, \ldots, x_{m+n}\}$ (and the associated value of u) is a column feasible set, then $v \geq u$.

Corollary: If row feasible y_i and column feasible x_i can be found such that $v = u$, then they constitute solutions to both the row and column equations; i.e., we have solved both the maximization and the minimization problems given above. We have such a solution only if $x_i y_i = 0$ for all i.

Proof: For row feasible y_i and column feasible x_i, by the previous corollary, we always have $v \geq u$. If, in fact, we have v = u, then v cannot become any smaller, and we have found the minimum. Similarly, u is a maximum. Also, $0 = v - u = \sum_{i=1}^{m+n} x_i y_i$ only if $x_i y_i = 0$ for all i.

We note that the converse of this second corollary is also true: namely, that if either the maximization or the minimization problem has a solution, then so does the other, and v = u for the two solutions. This converse will be stated and proved as a corollary of the main theorem of this paper.

To obtain a solution to our linear programming problem we shall define a finite sequence of pivot transformations beginning with our original schema and ending with a tableau from which the solution, if it exists, is easily obtained. At any step in this sequence we would have a tableau such as the following:

	y'_1	. . .	y'_n	1	
x'_{n+1}	a'_{11}	. . .	a'_{1n}	b'_1	$= -y'_{n+1}$
.	.		.	.	
.	.		.	.	
.	.		.	.	
x'_{n+m}	a'_{m1}	. . .	a'_{mn}	b'_m	$= y'_{m+n}$
1	c'_1	. . .	c'_n	d'	$= v$
	$=x'_1$	. . .	$=x'_n$	$=u$	

where the primed variables are some permutation of the original variables and the a'_{ij}, b'_i, c'_j, and d' are determined by the preceding pivot transformations.

In order to obtain a solution to our linear programming problem, we shall let the independent variables have the value zero, thereby determining values for the dependent variables:

$$y'_{n+1} = -b'_1 , \ldots , y'_{n+m} = -b'_m$$

$$x'_1 = c'_1 , \ldots , x'_n = c'_n$$

If $b'_1 \leq 0, \ldots, b'_m \leq 0$, we have a row feasible solution (see previous definition), and if $c'_1 \geq 0, \ldots , c'_n \geq 0$, we have a column feasible solution. This prompts the following definition.

Definition: A row feasible schema (column feasible schema) is one in which all $b'_i \leq 0$ (all $c'_j \geq 0$).

If in our sequence of pivot transformations we reach a schema which is both row and column feasible and we let all independent variables have the value zero, thus determining the values of the dependent variables, we then have $v = d' = u$, and by a previous corollary these values of the y'_i and x'_i constitute solutions to our linear programming problem. If, however, at some step we reach a schema in which for some i $b'_i > 0$ and $a'_{ij} \geq 0$ for $j = 1, \ldots , n$, then this schema exhibits the condition $-y'_{i+n} = \sum_{j=1}^{n} a'_{ij} \cdot y'_j + b'_i > 0$, which violates the requirement that all variables be non-negative. Thus the row program has no solution. If this same schema is a column feasible schema, then it is clear that if we allow x'_{n+i} to take on any value, no matter how large, while letting all other independent x' variables be zero, all the column constraints will be satisfied and u can be made as large as we like. Similarly, if we arrive at a schema in which for some j, $c'_j < 0$ and $a_{ij} \leq 0$ for $i = 1, \ldots, m$, there is no solution to the column program; and if this same schema is a row feasible schema, there is no lower bound for v.

This provides the motivation for the following definition.

<u>Definition</u>: <u>A simplex method</u> for solving a pair of linear programs is a finite seuence of schemata obtained from the schema of the given pair of programs by a sequence of successive pivot transformations which results in a schema exhibiting either optimal solutions to both programs or the non-compatibility of the row and/or the column constraints.

Before stating and proving the main theorem of this paper we give a definition.

<u>Definition</u>: Our <u>row</u> (<u>column</u>) <u>pivot</u> <u>choice</u> <u>rule</u> is applied to a row (column) feasible schema and is as follows: If the schema is row (column) feasible but not column (row) feasible, there must exist a $c'_j < 0$ for some j (a $b'_i > 0$ for some i). We sucessively examine from left to right (top to bottom) each column with $c'_j < 0$ (row with $b'_i > 0$) for a pivot choice as follows. For each such c'_j (b'_i) we have two possible cases:

1. Every entry in the column of this c'_j is nonpositive. (Every entry in the row of this b'_i is nonnegative.)
2. There exist positive (negative) entries.

If (1) holds, we examine the next column with $c'_j < 0$ (row with $b'_i > 0$) for a pivot choice. If (2) holds, choose as a pivot $a'_{kj} > 0$ ($a'_{i\ell} < 0$) satisfying

$$\frac{b'_k}{a'_{kj}} = \max_{a'_{sj} > 0} \frac{b'_s}{a'_{sj}} \qquad \left(\frac{c'_\ell}{a'_{i\ell}} = \max_{a'_{is} < 0} \frac{c'_s}{a'_{is}} \right)$$

if $b'_k \neq 0$ ($c'_\ell \neq 0$). All b'_s (c'_s) in the ratio above are less than or equal to zero (greater than or equal to zero) so the ratio is less than or equal to zero. If any b'_s (with $a'_{sj} > 0$) (c'_s (with $a'_{is} < 0$)) is zero, then the maximum above will be zero and we examine the next column with $c'_j < 0$ (row with $b'_i > 0$) for a pivot choice. If there is a tie for the choice of b'_k (c'_ℓ), we choose the top-most one (left-most one). If all $c'_j \geq 0$ (all $b'_i \leq 0$), i.e., if the schema is both row and column feasible, or if for each $c'_j < 0$ ($b'_i > 0$) either all column entries are nonpositive (all row entries are nonnegative) or the maximum of the ratio in (2) is zero, there is no pivot choice under this rule.

We note that with the choice of a'_{kj} as a row pivot by this rule row feasibility is retained after the pivot transformation has been performed:

1. $b''_k = a'^{-1}_{kj} \cdot b'_k < 0.$

2. $b''_i = b'_i - a'^{-1}_{kj} \cdot a'_{ij} \cdot b'_j \leq 0$, if $a'_{ij} \leq 0$, $i \neq k$.

3. $b''_i = b'_i - a'^{-1}_{kj} \cdot a'_{ij} \cdot b'_j \leq 0$, if $a'_{ij} > 0$, $i \neq k$.

because a'_{kj} was chosen to maximize the ratio. (The double-primed variables are the ones obtained after the pivot transformation has been performed.) Similarly a column pivot choice retains column feasibility after the pivot transformation has been performed.

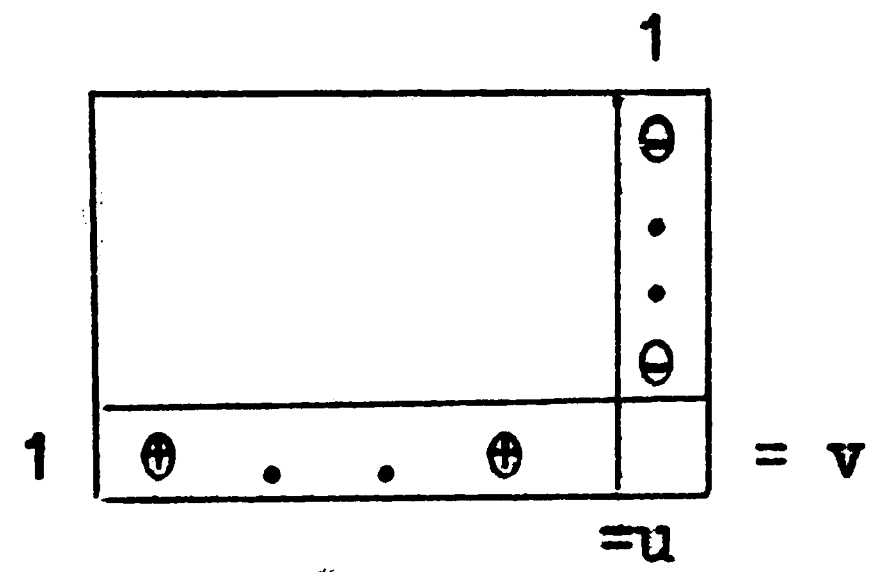
We note that $d'' = d' - b'_k . c'_j . a'_{kj} < d'$ after the pivot transformation has been performed with a'_{kj} as a row pivot choice under our rule. Thus a new row feasible solution has been obtained which gives a value to v which is strictly less than the previous exhibited value of v. Similarly a pivot transformation performed with $a'_{i\ell}$ as a column pivot choice results in a strict increase in the value of u.

If the initial schema does not exhibit either a row or a column feasible solution, the proof of our theorem gives a constructive means of producing one. The proof specifies pivot choices for any schema, whether row or column feasible or not, which lead to a schema exhibiting optimal solutions to both programs or the noncompatibility of the row and/or column constraints.

MAIN THEOREM:

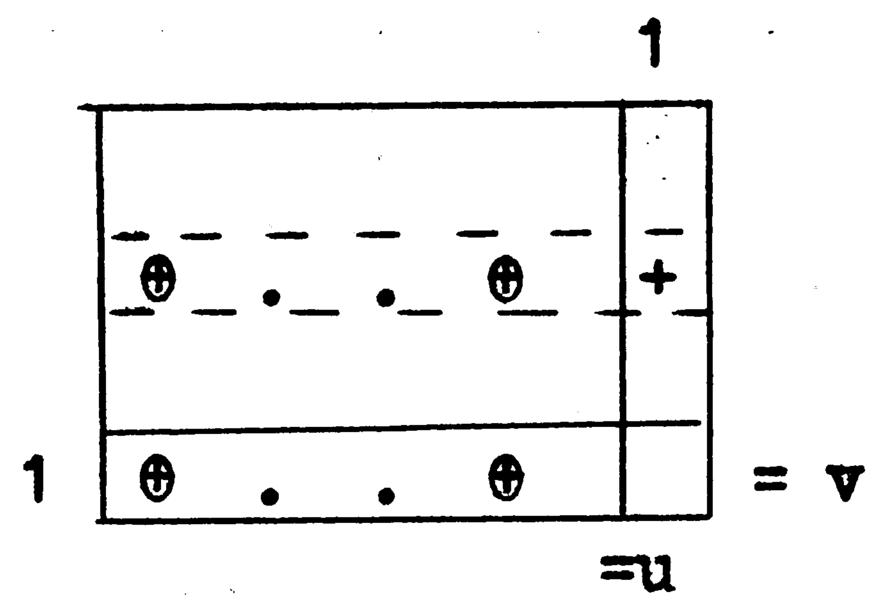
Theorem 2: Given a pair of linear programs as exhibited in schema I, there exists a simplex method which begins with the given schema and ends with one which has one of the following forms (where ⊕ and ⊖ denote nonnegative and non-positive entries respectively):

A



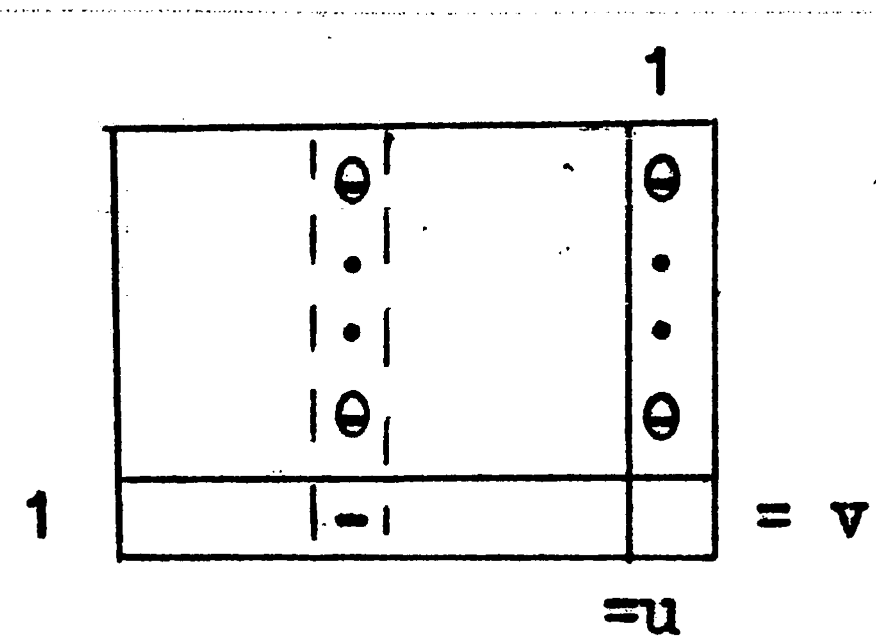
(which exhibits optimal solutions to both programs)

B



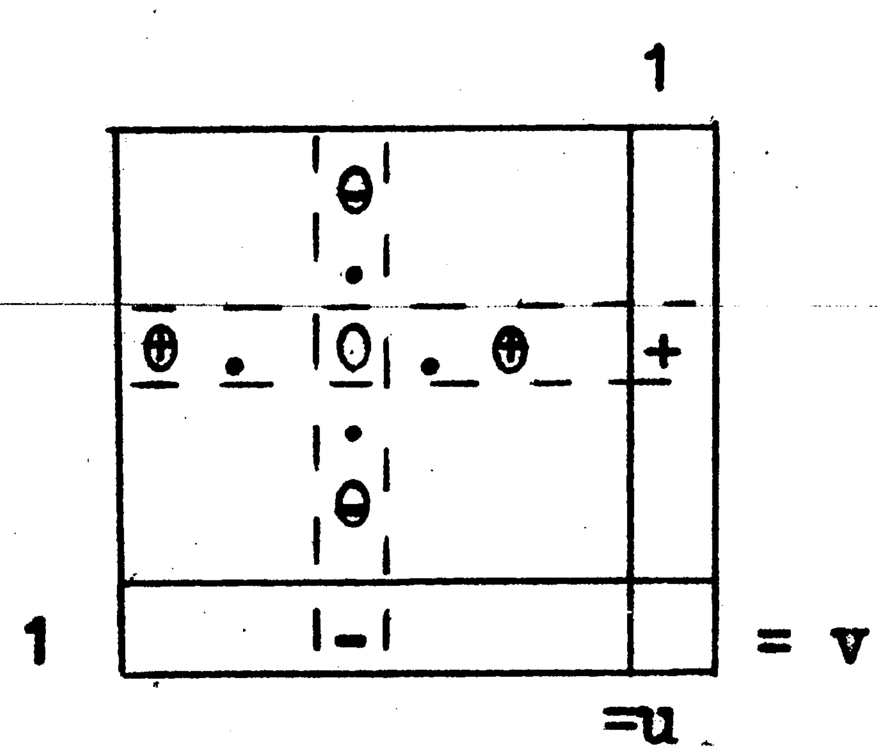
(which exhibits the non-compatibility of the row constraints and the unboundedness from above of u)

C



(which exhibits the non-compatibility of the column constraints and the unboundedness from below of v)

D



(which exhibits the non-compatibility of both the row and column constraints).

Before proving this theorem, we state and prove an important corollary:

Corollary: (Fundamental duality theorem of linear programming) If there exist feasible solutions to both row and column programs then there must exist optimal solutions to both.

Proof: Representations B, C, and D cannot occur, so A must.

Corollary: If either the row or column program has a solution, then so does the other and v = u for the two solutions.

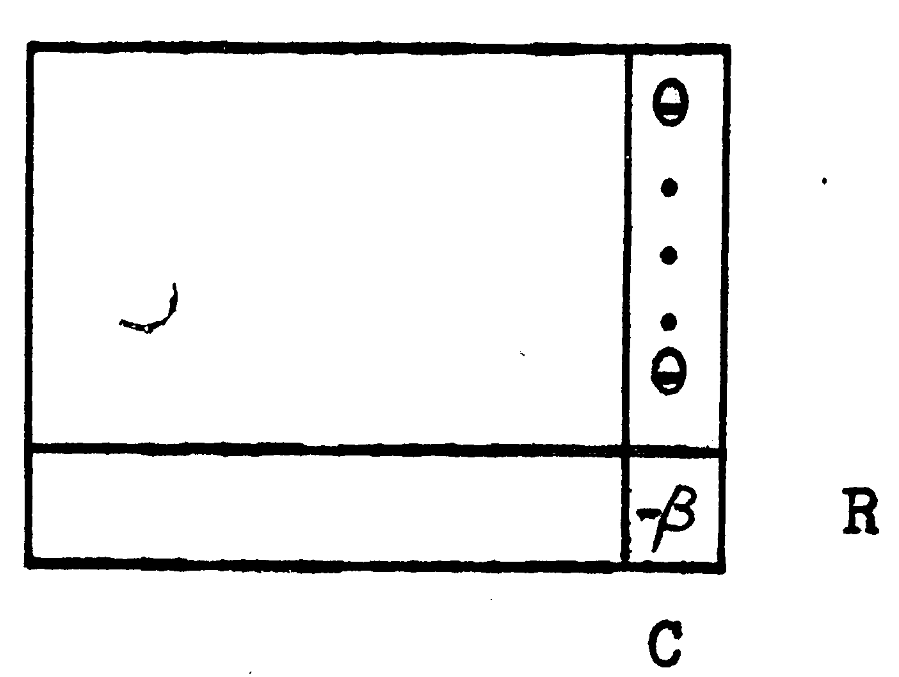
Proof: By the theorem there exists a simplex method which ends with Representation A. If in this final representation we let all independent variables have the value zero, we have a solution to both the row and column programs for which v = d' = u (where d' is the value of the entry in the lower right-hand corner). Since by a previous corollary $v \geq u$, any other solution to both the row and column programs must also have v = u.

Proof of Theorem:[1] The general idea of the proof is as follows. We define a hierarchy of levels of subschemata for our schema, each subschema contained in the previous one, and associate with each level k subschema a goal k. We examine the level 1 subschema to see if goal 1 can be accomplished by the method to be described. If it can, we perform a pivot transformation on the entire schema (and its hierarchy of levels of subschemata), accomplishing this goal. We then redefine the hierarchy of levels of subschemata and examine the new level 1 subschema to see if goal 1 can be accomplished. If at any

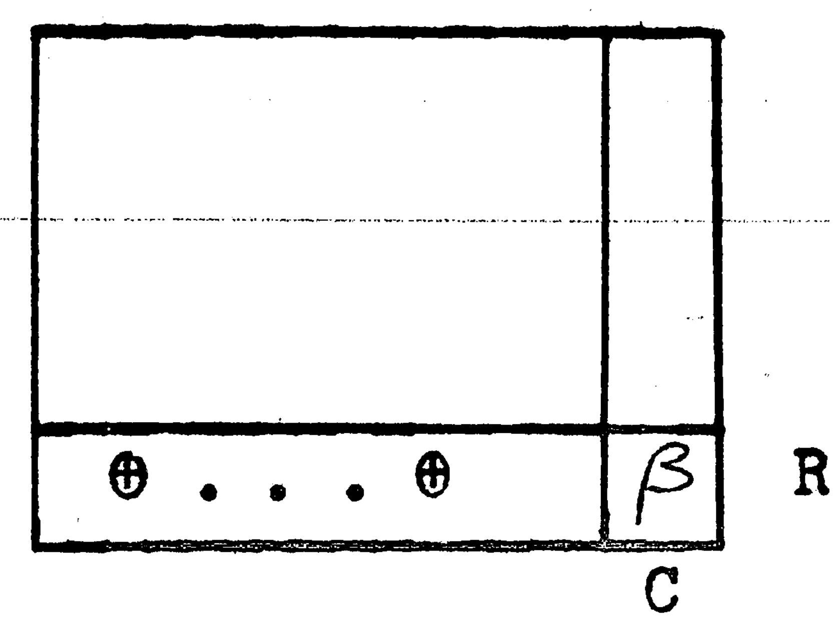
1 The following is a modification of the proof by Balinski and Gomory appearing in (1).

point goal 1 cannot be accomplished, we search for the lowest k for which goal k can be accomplished by the method to be described. When that k is found we perform a pivot transformation on the entire schema accomplishing goal k. After the transformation has been performed, we redetermine the hierarchy of subschemata and then examine the level 1 subschema to see if goal 1 can be accomplished and continue as before. In a finite number of steps we shall reach either Representation A of the theorem or a schema which exhibits the non-compatibility of the row and/or column constraints and from which we can obtain either Representation B, C, or D.

For any schema we choose according to certain rules a hierarchy of numbered levels of subschemata, there being one subschema associated with each level, each with a distinguished entry (and hence row and column). Level k subschemata for k odd are row feasible subschemata and have the form:

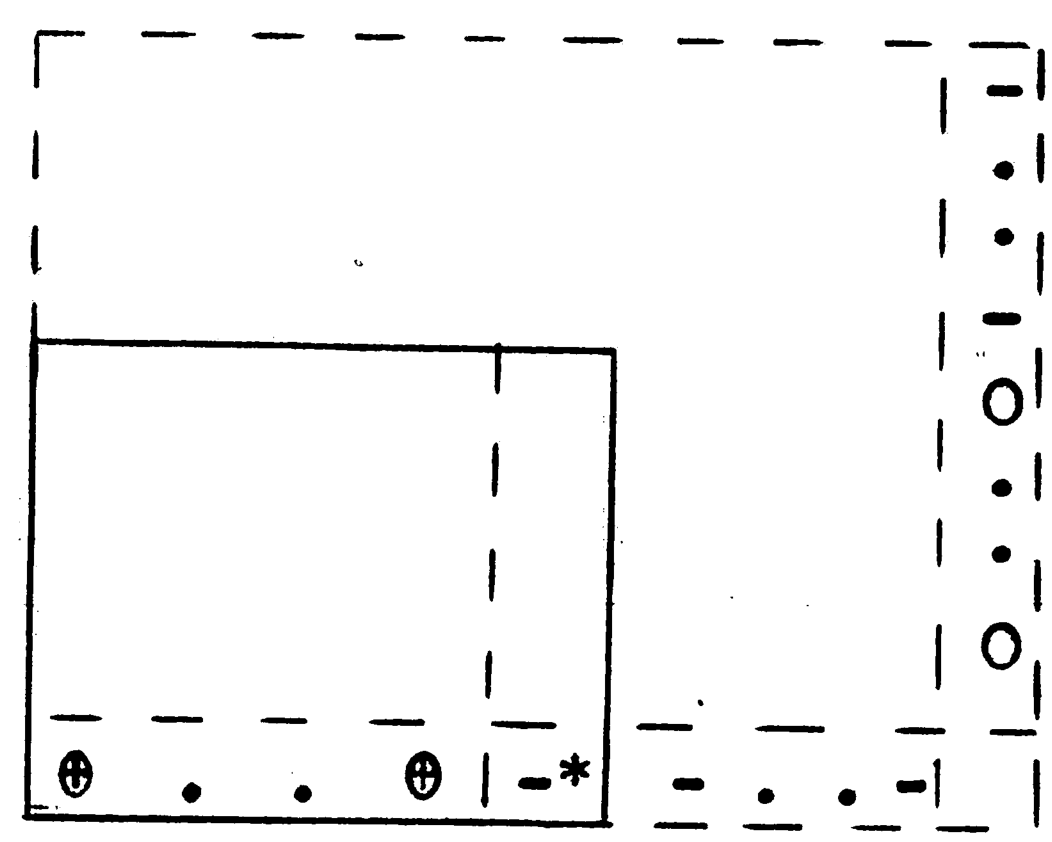


with the number of rows (including the distinguished row) designated by $\alpha(k)$ ($\alpha(k)$ may be 1 - the subschema then consists of only one row, the distinguished one) and $-\beta(k)$ the value of the distinguished entry. (For convenience of representation we rearrange rows and columns if necessary to have the distinguished entry in the bottom right-hand corner.) Even numbered level subschemata are column-feasible and have the form:



with $\alpha(k)$ the number of columns (including the distinguished column) and $\beta(k)$ the value of the distinguished entry.

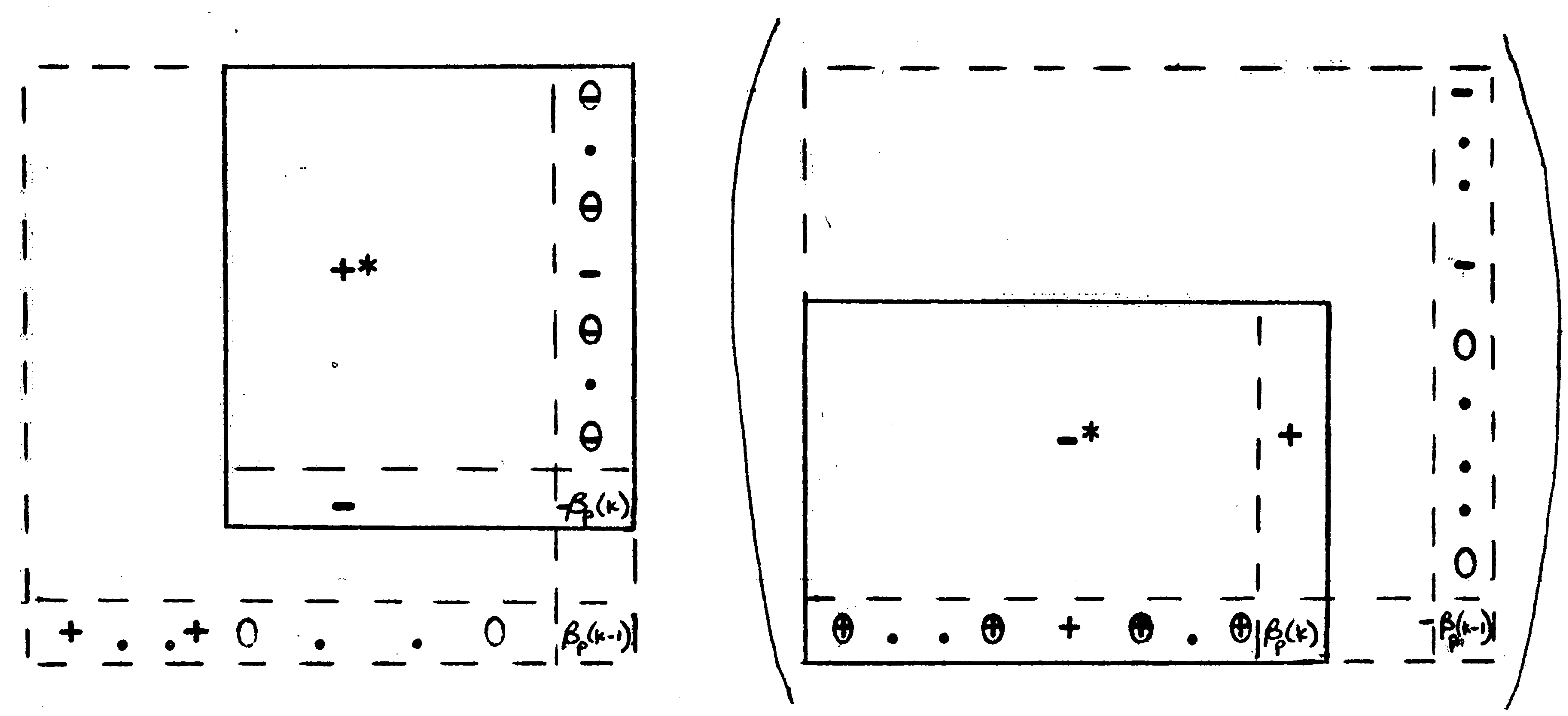
We define the hierarchy of subschemata associated with any schema inductively as follows. If not all $b'_i \leq 0$, the level 1 subschema consists of all columns of the schema (including the column of b'_is) and all rows in which $b'_i \leq 0$ together with the top-most row with $b'_i > 0$. (The row containing d' is not included.) This $b'_i > 0$ is made the distinguished entry. If all $b'_i \leq 0$, we take the whole schema as the level 1 subschema and d' (whether positive or not) as distinguished entry. Suppose now that the level k row feasible (column feasible) subschema has been defined with distinguished row R and distinguished column C. If C (R) contains zeros and R is not all nonnegative (C is not all nonpositive), we define the level k+1 column feasible (row feasible) subschema to consist of those rows (columns) for which the entry in C (R) is zero and those columns (rows) for which the entry in R is nonnegative (entry in C is nonpositive) together with the column (row) which contains the left-most negative entry of R (top-most positive entry of C). This left-most negative (top-most positive) entry is made the distinguished entry. The following is a schematic illustration of the construction of our hierarchy in which the k^{th} level subschema is row feasible and the $k+1^{th}$ level subschema is column feasible. The distinguished entry is marked with a star. (as before, "⊕" indicates a nonnegative quantity, "-" a negative quantity, and "0" a zero quantity.)



We associate with any schema and its subschemata a hierarchy of goals, with goal k being to perform a pivot transformation to obtain a new schema for which a hierarchy exists whose new level k subschema (if it exists) has $\alpha(k)$ larger or has $\alpha(k)$ unchanged but $\beta(k)$ larger, while $\alpha(i)$, $\beta(i)$ for $i < k$ remain unchanged.

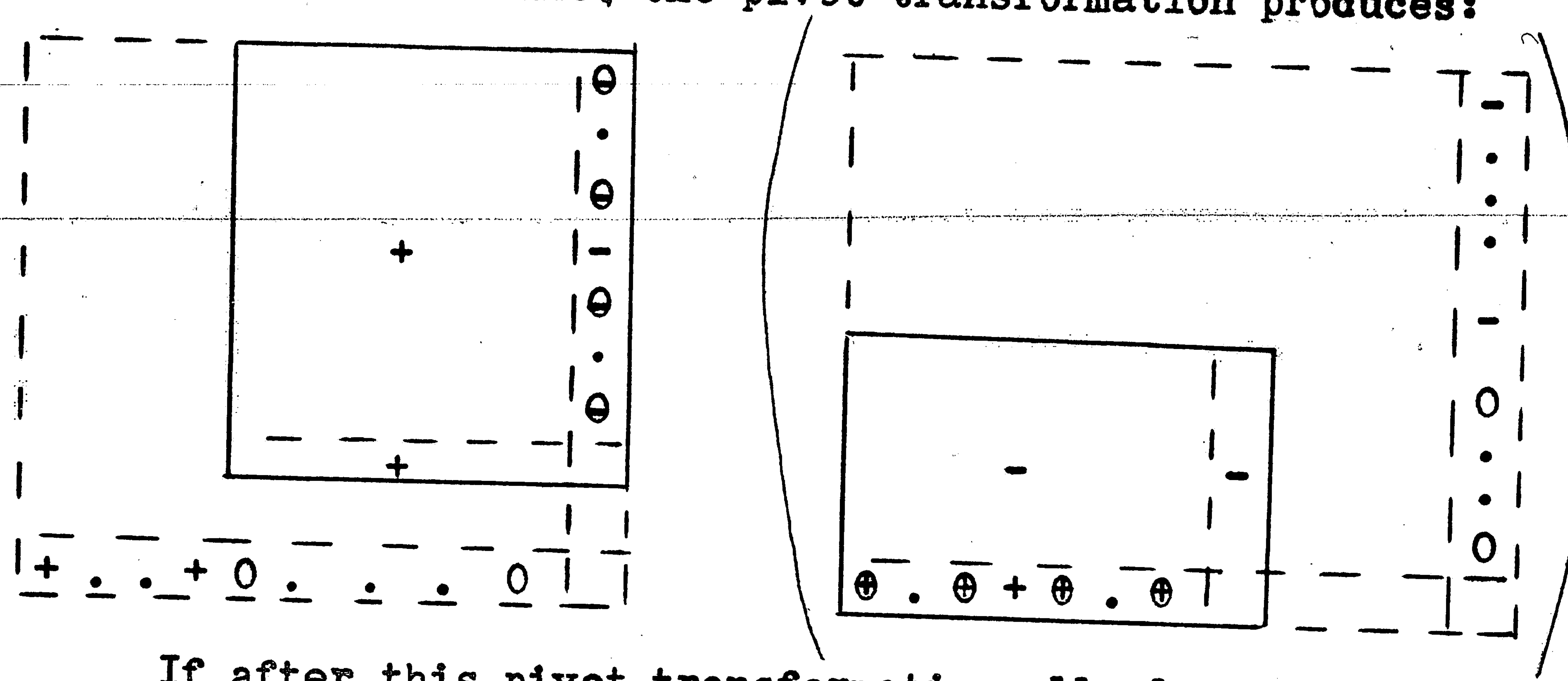
We initiate our sequence of pivot transformations by determining the level 1 subschema from our original schema. We proceed inductively as follows. Suppose we have performed the p^{th} pivot transformation in our sequence and have just determined the level k row (column) feasible subschema with distinguished row $R_p(k)$ and column $C_p(k)$ and with $-\beta_p(k)$ ($\beta_p(k)$) the value of the distinguished entry and $\alpha_p(k)$

the number of rows (columns). We first apply our row (column) pivot choice rule (see above) to this subschema. If there exists a pivot choice under this rule, we have (the level k subschema is enclosed in solid lines, the level k-1 one in dashed lines, and the pivot choice is starred):



(Of course if k = 1 there is no k-1 level subschema.)

We perform the pivot transformation on the entire schema and its intermediate level subschema. Because of the zeros in the distinguished row (column) of the level k-1 subschema above, none of the entries in that row (column) are changed after the pivot transformation has been performed and similarly for all lower level subschemata. Thus after the transformation the i^{th} level subschemata for $i < k$ consist of exactly the same selection of rows and columns of the entire schema as before (although the values of their entries may have changed), since it is precisely the unchanged distinguished rows and columns which determine the selection of the rows and columns of these subschemata. Since, as noted after the definition, row feasibility (column feasibility) is retained after the pivot transformation has been performed using our pivot choice under this rule, the pivot transformation produces:

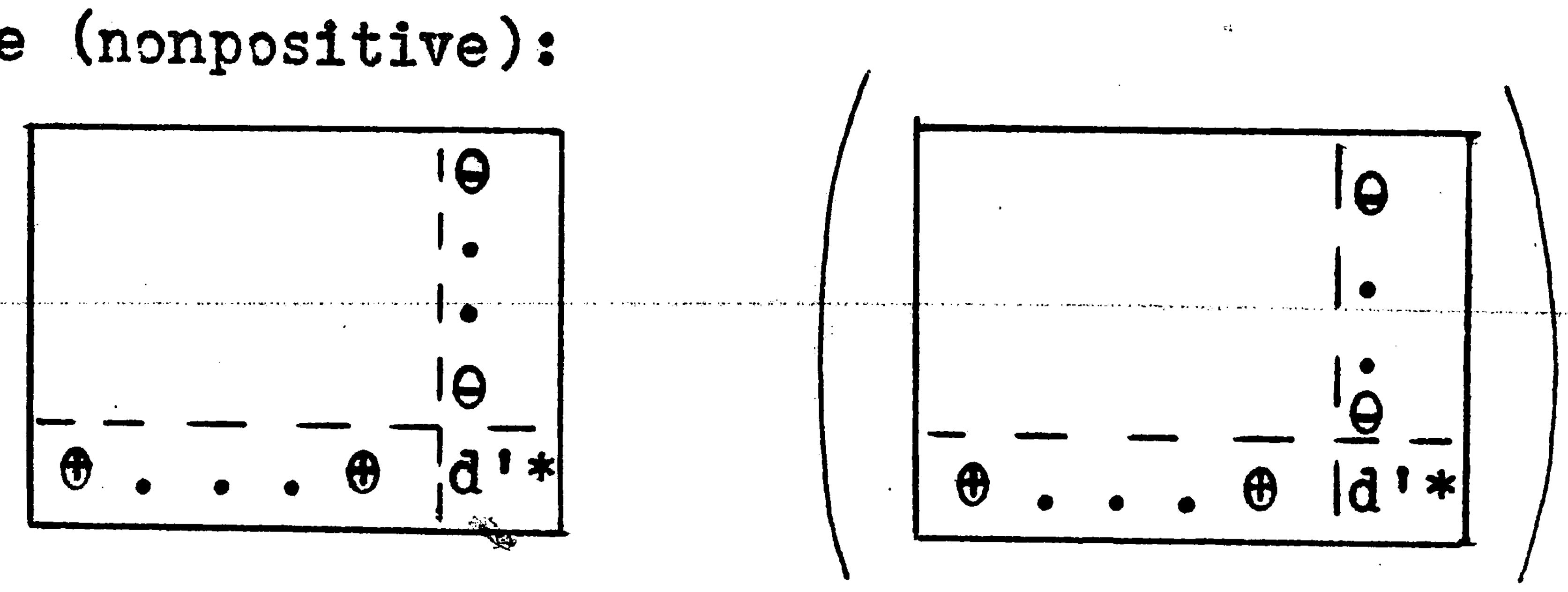


If after this pivot transformation all elements in $C_{p+1}(k-1)$ are nonpositive (in $R_{p+1}(k-1)$ are nonnegative), then there is no longer a k^{th} level subschema. (This of course does not apply for $k = 1$; there is no 0^{th} level subschema.) If there still is a k^{th} level schema, then $\alpha_{p+1}(k) \geq \alpha_p(k)$ because a pivot transformation performed with a pivot chosen under our row (column) pivot choice rule preserves row (column) feasibility, and if $\alpha_{p+1}(k) = \alpha_p(k)$, then $\beta_{p+1}(k) > \beta_p(k)$ since, as noted after the definition, the value of the distinguished entry is strictly decreased (increased) after the pivot transformation has been performed. As noted above $\alpha_{p+1}(i) = \alpha_p(k)$ and $\beta_{p+1}(i) = \beta_p(i)$ for $i < k$ since the pivot entry has zeros in rows and columns that could affect these values.

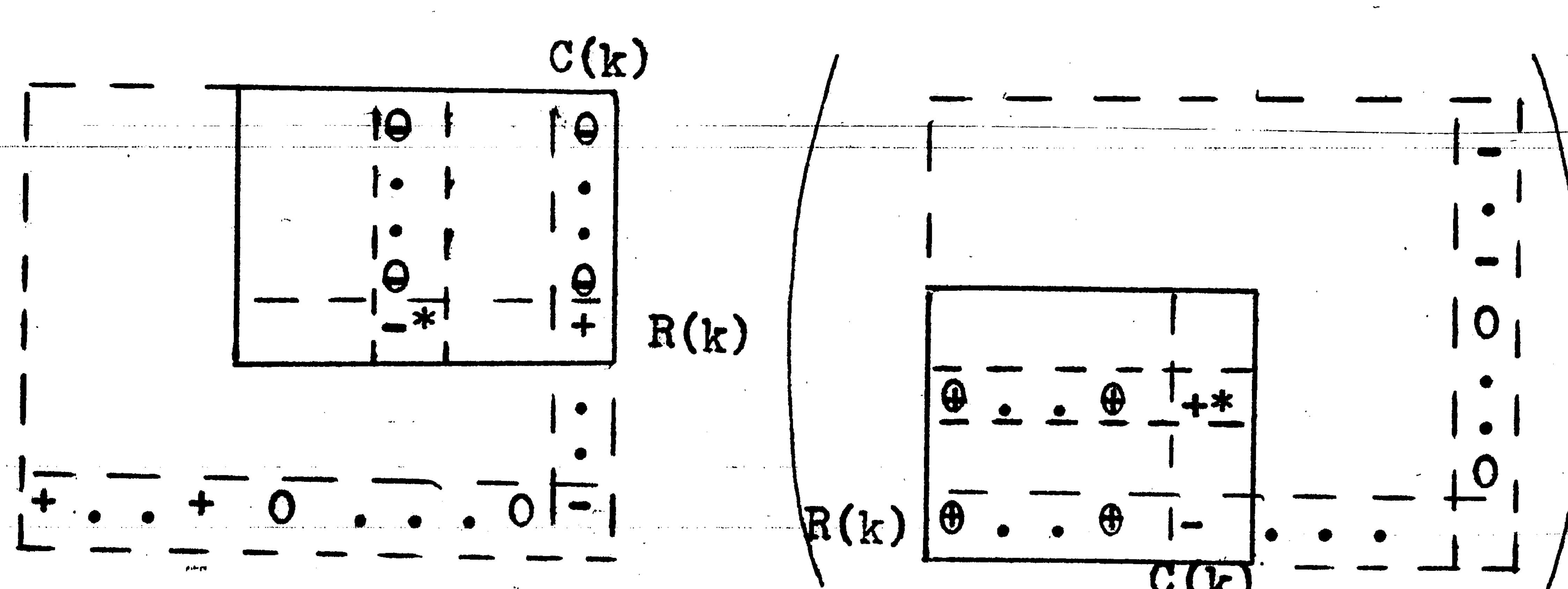
Thus after the pivot transformation has been performed, the i^{th} level subschemata for $i < k$ consist of the same selection of rows and columns of the entire schema as before (although the values of their entries may have changed). If a k^{th} level subschema still exists and $\alpha_{p+1}(k) = \alpha_p(k)$, the k^{th} level subschema will also consist of the same selection of rows and columns as before. In this case the higher level subschemata (if they exist) may consist of a different selection of rows and columns and we redetermine them where possible. If $\alpha_{p+1}(k) > \alpha_p(k)$, the level k subschema now contains additional rows (columns). We redefine it and any subsequent level subschemata if possible. We then return to the level 1 subschema and repeat our process from the beginning by applying our row pivot choice rule to that subschema.

If no pivot choice exists when our row (column) pivot choice rule is applied to the subschema, at least one of the following three cases must occur:

(1) Every entry in the distinguished row (column) of the subschema (except possibly the distinguished one) is nonnegative (nonpositive):



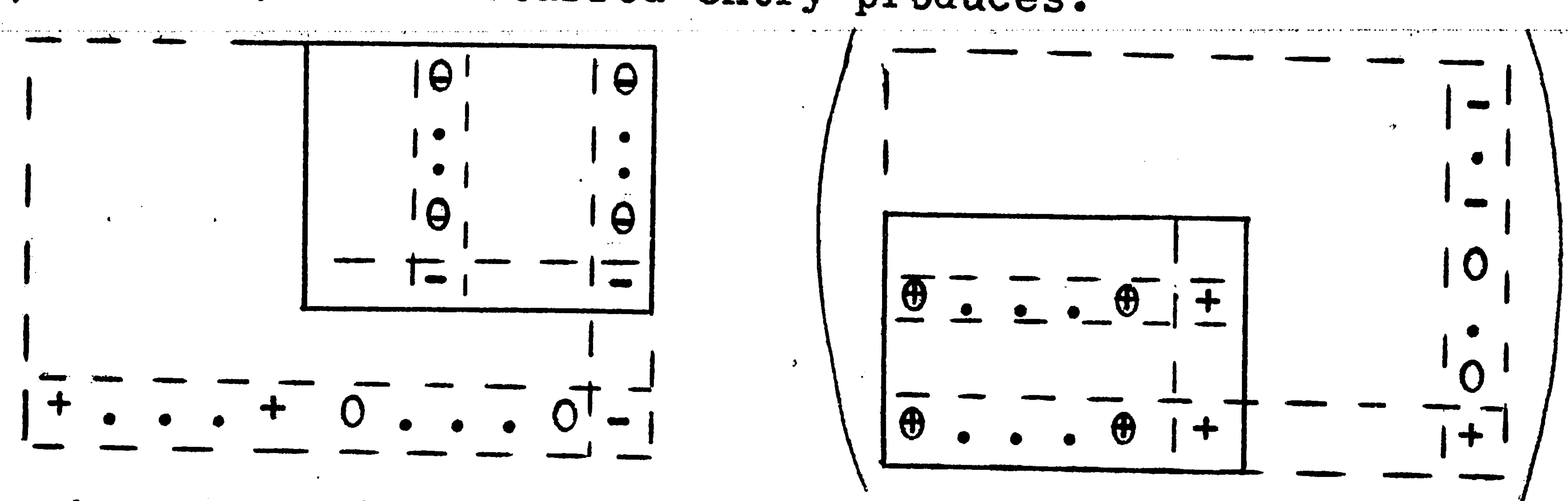
(2) The distinguished row (column) excluding the distinguished entry contains a negative (positive) element, but every entry in the column (row) of this element is nonpositive (nonnegative):



(3) The distinguished row (column) excluding the distinguished entry contains a negative (positive) element and at least one entry, call it "a", in the column (row of this element is positive (negative), but the distinguished column (row) entry which is in the same row (column) as "a" is zero. (Remember that in the definition of our row (column) pivot choice rule there was no pivot choice if the maximum of the ratio was zero.)

We first determine whether case 1 holds. If it does, we choose as a pivot the distinguished entry (starred). However, we do not perform the pivot transformation because, as we shall show below, case 1 can occur only if k = 1, and we stop when the pivot choice is an element of the last row or column of the total schema.

If case 1 does not hold, we determine if case 2 does. If so, pivoting on the starred entry produces:

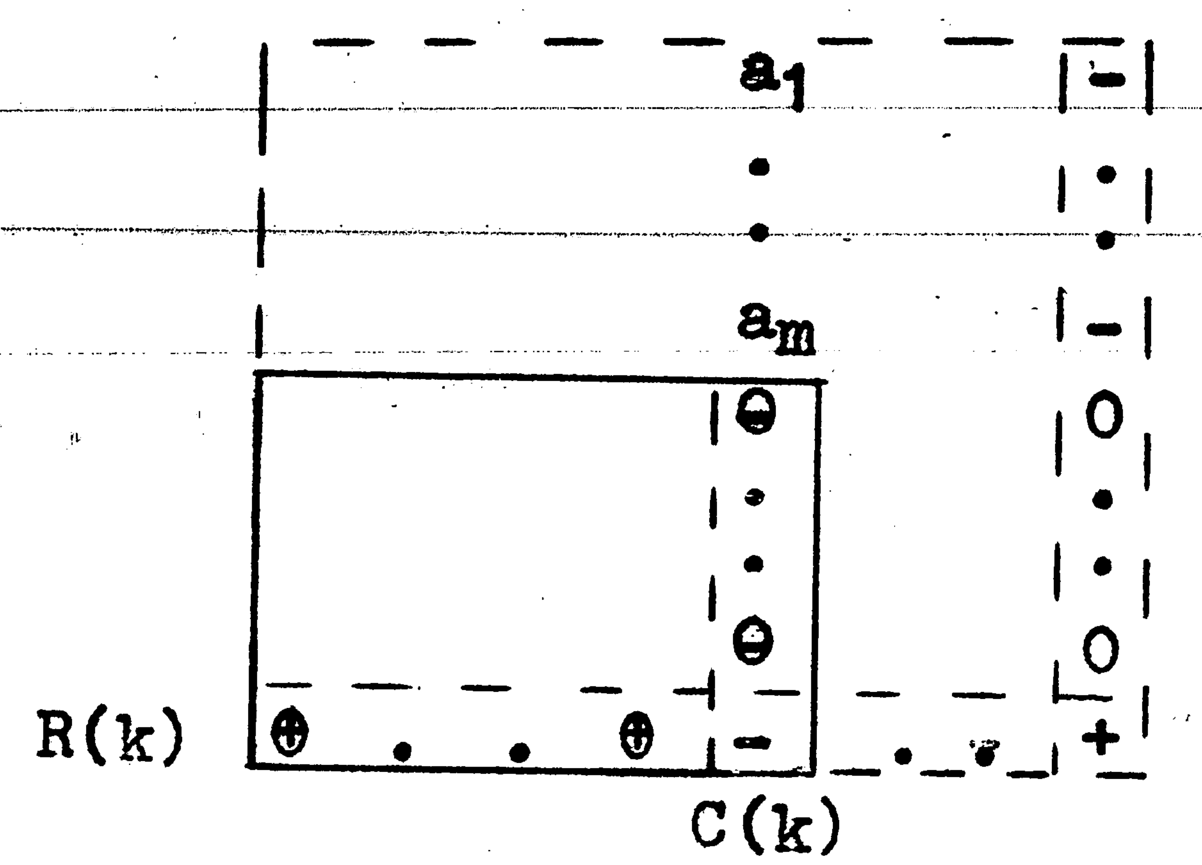


As before, because of the zeros in the distinguished rows and columns, the subschemata up to and including the $(k-1)^{th}$ one will consist of exactly the same selection of rows and columns of the entire schema as before (although the value of their entries may have changed). If all entries in $C_{p+1}(k-1)$ $(k>1)$ are nonpositive (in $R_{p+1}(k-1)$ are nonnegative), then there no longer is a k^{th} level subschema. Otherwise, since the distinguished entry in the k^{th} level subschema has become negative (positive), we have $\alpha_{p+1}(k) > \alpha(k)$, while $\alpha_{p+1}(i) = \alpha_p(i)$ and $\beta_{p+1}(i) = \beta_p(i)$ for $i < k$. After the pivot transformation has been completed, we redetermine the rows and columns of the k^{th} level subschema (if it exists) and any subsequent ones if possible. We then return to the level 1 subschema and repeat the process by applying our row pivot choice rule to that subschema.

If cases 1 and 2 do not occur, 3 must. Since in this case $R_p(k)$ contains at least one negative entry ($C_p(k)$ contains at least one positive entry) and $C_p(k)$ ($R_p(k)$) contains at least one zero, there exists a level k+1 column (row) feasible subschema. We then repeat our inductive step on the $(k+1)^{th}$ level subschema, applying our column (row) pivot choice rule to the subschema, and if this yields no pivot choice sucessively determining if we have case 1, 2, or 3.

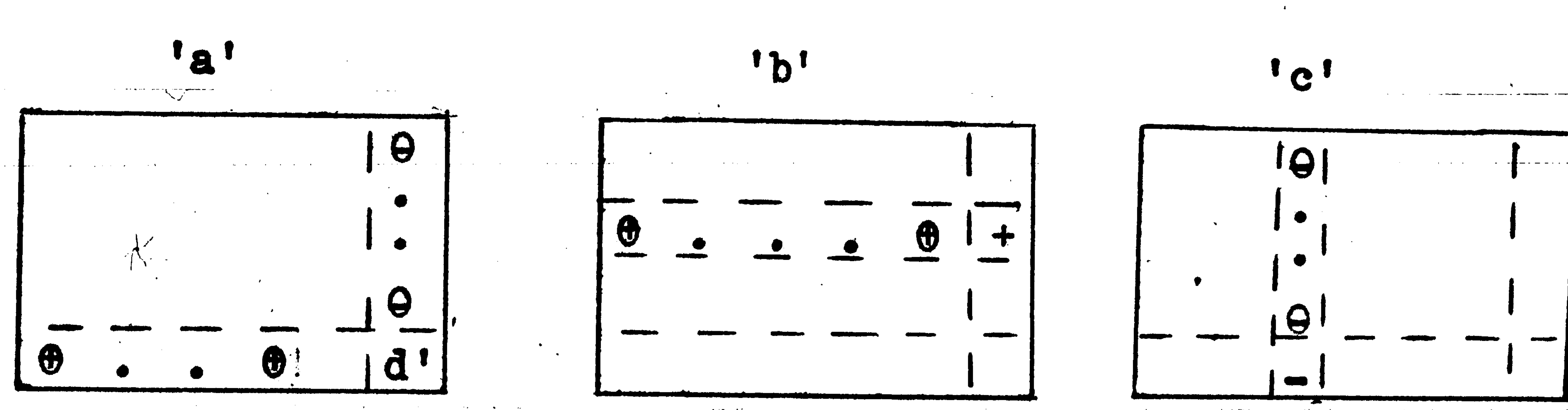
We repeat the above until at some point the choice of pivot entry is an element of the last row or column of the total schema (some b_i', c_j', or d'). This occurs only if we have either case 1 or case 2. It must occur in a finite number of steps since our hierarchy of goals of increasing $\alpha(k)$ and/or $\beta(k)$ at each step insures our not repeating any schema and there exist at most $\binom{n+m}{n}$ possible equivalent schemata.

Before we continue, it still must be shown that with our simplex method case 1 can occur only if $k = 1$. Suppose $k > 1$. There then exists a level k-1 subschema and we have (we illustrate with a column feasible k level subschema; the argument is analogous if the k level subschema is row feasible):



If there exist any a_i, none of them can be greater than zero because then a row pivot choice would have existed when our row pivot choice rule was applied to the level k-1 subschema and we would have returned to the level 1 subschema instead of determining the k^{th} level one. If no a_i exists (because all of the entries in C(k-1), except possibly the distinguished entry, are zero) or if all $a_i \leq 0$, we have an impossibility because we then would have had case 2 for the level k-1 subschema and again we would have returned to the level 1 one. Thus we must have k = 1.

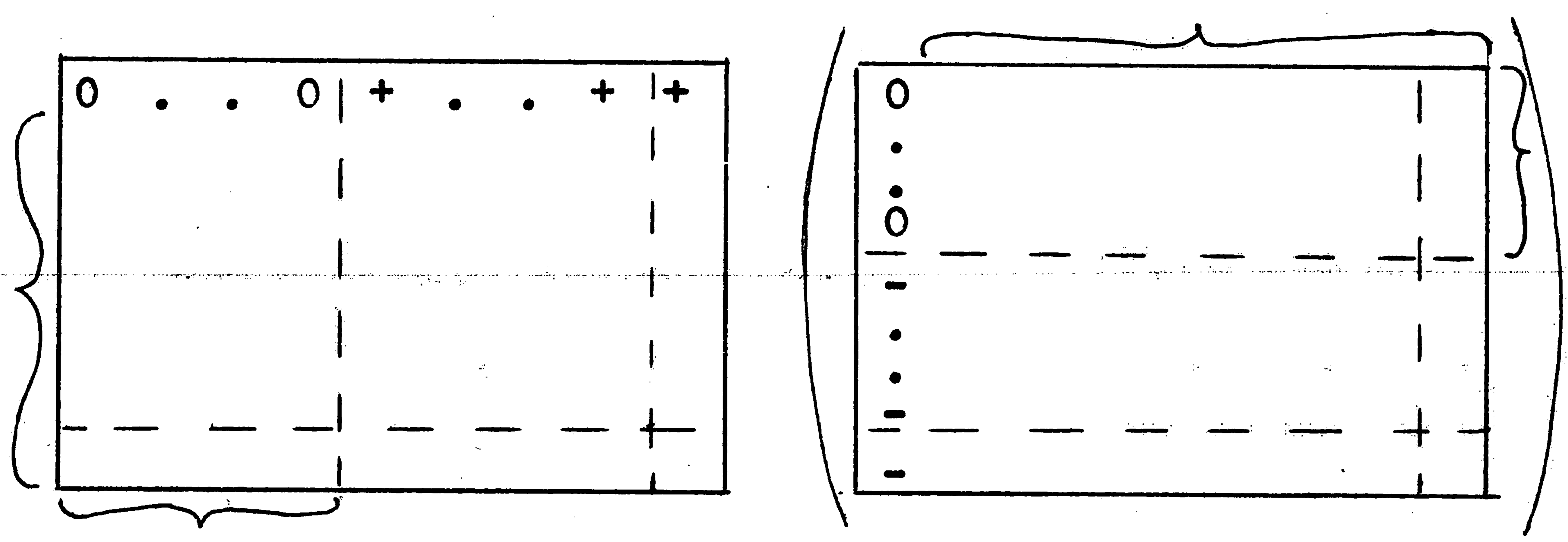
When our pivot choice is an element of the last row or column of the total schema we have one of the following three cases:



If 'a' occurs we have Representation A of the theorem and have found solutions to both the row and column problems.

If 'b' ('c') occurs and every entry in the indicated row (column) is strictly positive (strictly negative) we perform pivot transformations using as pivots those a'_{ij} in the indicated row (column) for which $c'_j < 0$ ($b'_i > 0$) until all c'_j become nonnegative (all b'_i become nonpositive), obtaining Representation B (C) of the theorem. (This occurs in a finite number of steps since once a c'_j becomes nonnegative (b'_i becomes nonpositive) it remains so.)

Otherwise, if zeros occur in the indicated row in 'b' (column in 'c'), we rearrange rows and columns if necessary to obtain:

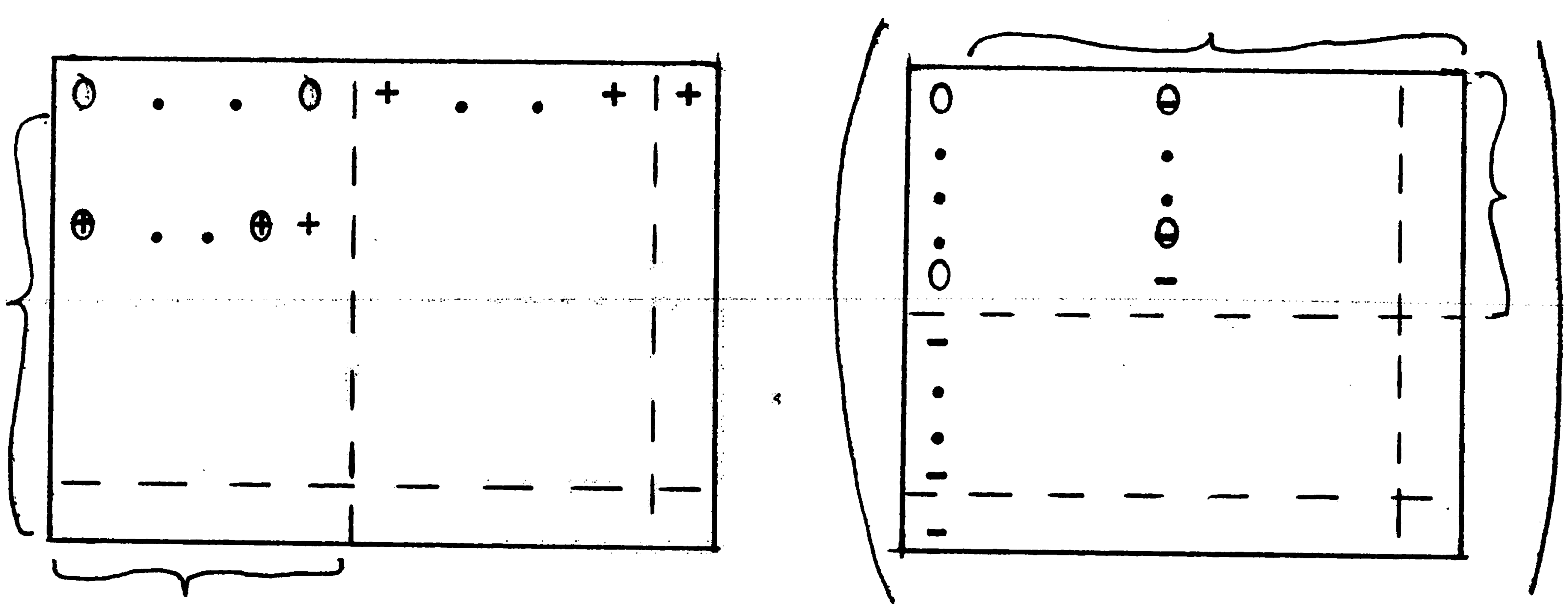


We treat the rows and columns enclosed by brackets as an entire schema and choose pivots and perform pivot transformations as we did for the whole schema until we obtain representation 'a', 'b', or 'c' above in this smaller schema. (We note that because of the zeros, the positive entries in the indicated row (negative entries in the indicated column) remain unchanged during this sucession of pivot transformations.

If we obtain representation 'a', and d' for the smaller schema is strictly less than zero (strictly greater than zero), we have arrived at Representation D of the theorem. If we have 'a', and $d' \geq 0$ ($d' \leq 0$) we pivot on all positive (negative) a'_{ij} in the indicated row (column) of the larger schema for which c'_j of the larger schema is less than zero (b'_i is greater than zero) until all c'_j become nonnegative (all b'_i become nonpositive) obtaining Representation B (C).

If we obtain representation 'c' ('b'), we have obtained Representation D of the theorem.

If we obtain representation 'b' ('c'), we have a schema such as:



If the indicated row (column) in the subschema enclosed by brackets contains no zeros we pivot on positive (negative) entries as before until all $c'_j \geq 0$ (all $b'_i \leq 0$), obtaining Representation B (C). If zeros are present we determine a still smaller subschema consisting of the columns with zeros in the indicated row (the rows with zeros in the indicated column) and all rows (columns) excluding the present indicated one and the previous indicated one, and apply the above rules until we arrive at representation 'a', 'b', or 'c' and repeat the above. Since there are only a finite number of rows and columns in our schema, we must obtain one of the four representations of the theorem in a finite number of steps. This completes the proof of the theorem.

We note that in solving an actual linear programming problem we would ordinarily stop computations upon reaching Representation A or representations 'b' or 'c' since the last two representations cannot lead to an optimal solution.

EXAMPLE

We illustrate the theorem with an example. Consider the following problem.

$$\text{Maximize:} \quad -3x_2 + 4x_3 + 4x_4$$

$$\begin{aligned}
\text{Subject to:}\quad x_1 - 6x_2 - x_3 - 4x_4 & \geq 0 \\
x_1 - 5x_2 - 2x_3 + 10x_4 + 2 & \geq 0 \\
-x_1 + 5x_2 + 3x_4 + 6 & \geq 0 \\
-x_1 + 6x_2 - 2x_4 + 1 & \geq 0 \\
-4x_2 + x_3 + 6x_4 + 10 & \geq 0 \\
x_1,\ x_2,\ x_3,\ x_4 & \geq 0
\end{aligned}$$

In tableau form:

	y_5	y_6	y_7	y_8	y_9	1	
x_1	1	1	-1*	-1	0	0	$= -y_1$
x_2	-6	-5	5	6	-4	-3	$= -y_2$
x_3	-1	-2	0	0	1	4	$= -y_3$
x_4	-4	10	3	-2	6	4	$= -y_4$
1	0	2	6	1	10	0	$= v$
	$=x_5$	$=x_6$	$=x_7$	$=x_8$	$=x_9$	$=u$	

(I)

The level 1 subschema is:

						C	
(1)	1	1	-1	-1	0	0	
(2)	-6	-5	5	6	-4	-3	
(3)	-1	-2	0	0	1	4	R
	(5)	(6)	(7)	(8)	(9)		

(We employ the numbers in parentheses as a labeling device to keep track of which rows and columns of the entire schema we are working with.) No pivot choice exists for this subschema: Our row pivot choice rule gives none (the two columns with $c_j < 0$ do have positive entries, but the maximum of the ratio is zero), and we do not have cases 1 or 2. We therefore determine the 2nd level subschema:

	(7)	(8)	(9)	(5) C	
(1)	-1	-1	0	1	
(3)	0	0	1	-1	R

Again no pivot choice exists, so we determine the 3rd level subschema:

	(7)	(8)	(5) C	
(1)	-1*	-1	1	R

We have case 2 and therefore pivot on the left-most -1 (starred) in the 3rd level subschema obtaining:

	y_5	y_6	y_1	y_8	y_9	1	
x_7	-1	-1	-1	1	0	0	$= -y_7$
x_2	-1	0	5	1	-4	-3	$= -y_2$
x_3	-1*	-2	0	0	1	4	$= -y_3$
x_4	-1	13	3	-5	6	4	$= -y_4$
1	6	8	6	-5	10	0	$= v$
	$=x_5$	$=x_6$	$=x_1$	$=x_8$	$=x_9$	$=u$	

(II)

We now return to the level 1 subschema (which consists of exactly the same rows and columns as before since a pivot choice in a level 3 subschema alters the hierarchy only for levels greater than 2):

	(5)	(6)	(1)	(8)	(9)	C	
(7)	-1	-1	-1	1	0	0	
(2)	-1	0	5	1	-4	-3	
(3)	-1*	-2	0	0	1	4	R

There exists no pivot choice under our row pivot choice rule, but we do have case 2. Pivoting on the -1 produces:

	y_3	y_6	y_1	y_8	y_9	1	
x_7	-1	1	-1	1*	-1	-4	$= -y_7$
x_2	-1	2	5	1	-5	-7	$= -y_2$
x_5	-1	2	0	0	-1	-4	$= -y_5$
x_4	-1	15	3	-5	5	0	$= -y_4$
1	6	-4	6	-5	16	24	$= v$
	$=x_3$	$=x_6$	$=x_1$	$=x_8$	$=x_9$	$=u$	

(III)

The level 1 subschema now consists of the entire schema. Application of our row pivot choice rule to this level 1 subschema yields the "1" (starred) in the fourth column as the pivot choice. Pivoting produces:

	y_3	y_6	y_1	y_7	y_9	1	
x_8	-1	1	-1	1	-1	-4	$= -y_8$
x_2	0	1	6	-1	-4	-3	$= -y_2$
x_5	-1	2	0	0	-1	-4	$= -y_5$
x_4	-6	20	-2	5	0	-20	$= -y_4$
1	1	1	1	5	11	4*	$= v$
	$=x_3$	$=x_6$	$=x_1$	$=x_7$	$=x_9$	$=u$	

(IV)

Looking at the level 1 subschema again we find we have case 1 and we choose 4 (starred) as a pivot. We stop since d' was chosen as a pivot.

The solution to our maximization problem is

$$x_1 = 1, x_2 = 0, x_3 = 1, x_4 = 0$$

with 4 the desired maximum.

COMPARISON WITH BALINSKI - GOMORY METHOD

The simplex method presented by Balinski and Gomory (1) is similar to the one presented above. It differs in that no rule is given to uniquely determine the distinguished entry in building the hierarchy of subschemata for the main schema. In addition the authors work on the hierarchy in a different sequence. They build the entire hierarchy and immediately attempt to accomplish goal k (increasing α and/or β), where the level k subschema is the highest (i.e., has the highest number in our system of numbering subschema levels) in the hierarchy which is not both row and column feasible. Either the row (column) pivot choice rule or case 1 or case 2 then yields a pivot choice. When the pivot transformation has been performed the hierarchy is redefined for the levels $\geq$ k if possible. The level subschema now highest in the hierarchy which is not both row and column feasible is examined as before and a pivot choice made. This continues until termination under the same rules as presented in this paper.

It appears, heuristically at least, that working on goal 1 would be more effective toward solving the entire linear program than working on a higher goal would be. In the simplex method presented in this paper always returning to the level 1 subschema to look for a pivot choice is exactly the same as choosing the distinguished entry in the Balinski - Gomory method so that the subschema is both row and colum feasible whenever possible.

We illustrate by applying the simplex method of (1) to our example above. To avoid ambiguity we shall always choose the left-most or top-most possible distinguished entry when there is a choice to be made. (The authors of (1) make the choice arbitrary.)

In (I) the pivot choice would be the same as ours since the level 3 subschema is the highest one which is not both row and column feasible. Again in (II) the choice would be the same. However in (III) we would not work on the level 1 subschema, but rather determine the level 2 subschema (which is the highest level subschema which is not both row and column feasible):

	(3)	(1)	(9)	(6)
(4)	-1*	3	5	15
	6	6	16	-4

Application of our column pivot choice rule yields the -1 (starred) as a pivot choice. Pivoting produces:

	y_4	y_6	y_1	y_8	y_9	1	
x_7	-1	-14	-4	6	- 6	-4	$= -y_7$
x_2	-1	-13	2	6	-10	-7	$= -y_2$
x_5	-1	-13	-3	5	- 6	-4	$= -y_5$
x_3	-1	-15	-3	5	- 5	0	$= -y_3$
1	6	86	24	-35	46	24	$= v$
	$=x_4$	$=x_6$	$=x_1$	$=x_8$	$=x_9$	$=u$	

We have now performed the same number of pivot transformations as in the example, but we have not reached an optimal solution. In fact, it will take three more pivot transformations to reach the solution.

Now if in (III) we had chosen the -5 instead of the -4 as a distinguished entry, the level 2 subschema would have been:

	(3)	(1)	(9)	(8)
(4)	-1	3	5	-5
	6	6	16	-5

which is both row and column feasible. Under the Balinski-Gomory simplex method one would go back one level since the level 1 subschema is the highest on which is not both row and column feasible. The pivot choice would then be the same as in the example and we would reach the optimal solution immediately.

EXTENSIONS:

Suppose we wish to maximize $\sum_{i=1}^{M} b_i t_i + d$ subject to the inequalities $\sum_{i=1}^{M} a_{ij} t_i + c_j \geq 0$, $j = 1, \ldots, N$, where the t_i are no longer required to be nonnegative. We set up the following schema:

	y_1	$\cdots$	y_N	1	
t_1	a_{11}	$\cdots$	a_{1N}	b_1	$= 0$
$\vdots$	$\vdots$		$\vdots$	$\vdots$	$\vdots$
t_M	a_{M1}	$\cdots$	a_{MN}	b_M	$= 0$
1	c_1	$\cdots$	c_N	d	$= v$
	$= x_1$	$\cdots$	$= x_N$	$= u$	

where we require only the x_j and y_j to be nonnegative for all j. It is easily verified that Theorem 1 and its corollaries are still true for this new program.

To solve this problem we attempt to eliminate the t_i as independent variables in the above column program. To do this we make pivot transformations on the above schema and its sucessive representations using as a pivot choice any non-zero entry corresponding to x-dependent and t-independent column program labels until no longer possible. Two possible cases can result:

1. Every entry which corresponds to x-dependent and t-independent column program labels is zero.
2. No column program independent labels are t's.

If (1) results we have a schema of the form:

	y'_{m+1}	...	y'_{m+n}	0	...	0	1	
x'_1	a'_{11}	...	a'_{1n}	a'_{1n+1}	..	a'_{1N}	b'_1	$= -y'_1$
.	.		.	.		.	.	.
.	.		.	.		.	.	.
x'_m	a'_{m1}	...	a'_{mn}	a'_{mn+1}	..	a'_{mN}	b'_m	$= -y'_m$
t'_{m+1}	0	...	0	$a'_{m+1,n+1}$	.	a'_{m+1N}	b'_{m+1}	= 0
.	.		.	.		.	.	.
.	.		.	.		.	.	.
t'_M	0	...	0	a'_{Mn+1}	..	a'_{MN}	b'_M	= 0
1	c'_1	...	c'_n	c'_{n+1}	...	c'_N	d'	$= v'$
	$=x'_{n+1}$	...	$=x'_{m+n}$	$=t'_1$	...	$=t'_m$	$=u'$	

where the primed variables are a rearrangement of the variables in the original schema and the primed entries are the result of the preceeding pivot steps. If any of the b'_i, $i = m+1$, ..., M is not zero, then we have a row equation which states that a nonzero quantity equals zero. This is clearly impossible, so the row constraints are incompatible, and consequently there is no optimal solution to our maximization program.

If, on the other hand, we have $b'_{m+1} = \dots = b'_M = 0$, the row equations corresponding to dependent 0-labels read "zero equals zero" and can consequently be eliminated from the row program. Clearly the columns corresponding to the independent 0-labels can also be omitted from the row program. Since the t'_i are not required to be nonnegative, the column equations with dependent t'_i variables can be eliminated from the column program since they represent no constraints. Having done this, we see that the coefficients corresponding to the independent t_i' are all zeros, and hence their rows can be omitted from the schema. Thus if case 1 occurs and we have $b_{m+1}' = \ldots = b_M' = 0$, we have a smaller representation for our column program; in fact, it has the same form as the schema we considered earlier. We apply the technique of the theorem to obtain optimal values for x_j (or show that none exist), and then use these values to determine the optimal values for the t_i.

If case (2) occurs, then all t-labels are dependent and all 0-labels are independent and we proceed as above.

A simple example illustrates the method:

$$\begin{aligned} &\text{Maximize} && t_1 - t_2 \\ &\text{subject to} && -t_1 - t_2 + 2t_3 \geq 0 \\ &&& -2t_1 + t_2 + t_3 \geq 0 \\ &&& t_1 + t_2 - 2t_3 + 1 \geq 0 \end{aligned}$$

	y_1	y_2	y_3	1	
t_1	-1	-2	1	1	= 0
t_2	-1	1*	1	-1	= 0
t_3	2	1	-2	0	= 0
1	0	0	1	0	= v
	$=x_1$	$=x_2$	$=x_3$	$=u$	

	y_1	0	y_3	1	
t_1	-3*	2	3	-1	= 0
x_2	-1	1	1	-1	$= -y_2$
t_3	3	-1	-3	1	= 0
1	0	0	1	0	= v
	$=x_1$	$=t_2$	$=x_3$	$=u$	

	0	0	y_3	1	
x_1	-1/3	-2/3	-1	1/3	$= -y_1$
x_2	-1/3	1/3	0	-2/3	$= -y_2$
t_3	1	1	0	0	$= 0$
1	0	0	1	0	$= v$
	$=t_1$	$=t_2$	$=x_3$	$=u$	

We have $b_3' = 0$, so we temporarily ignore the third row and the first two columns.

	y_3	1	
x_1	-1*	1/3	$= -y_3$
x_2	0	-2/3	$= -y_2$
1	1	0	$= v$
	$=x_3$	$=u$	

The level 1 subschema consists of the first two rows with the distinguished entry "1/3". Choosing the pivot entry as in case 2 and performing the pivot transformation we have:

	y_1	1	
x_3	-1	-1/3	$= -y_3$
x_2	0	-2/3	$= -y_2$
1	1	1/3	$= v$
	$=x_1$	$=u$	

which exhibits optimal solutions.

Thus we have $x_1 = 1$, $x_2 = 0$, $x_3 = 0$. Using the third schema above we have $t_3 = 0$

$$t_1 = -1/3x_1 - 1/3x_2 + t_3 = -1/3$$

$$t_2 = -2/3x_1 + 1/3x_2 + t_3 = -2/3$$

BIBLIOGRAPHY

1. Balinski, Michel L. and Gomory, Ralph E., "A Mutual Primal-Dual Simplex Method" in Recent Advances in Mathematical Programming, edited by Graves, Robert L. and Wolfe, Philip, McGraw-Hill, 1963.

2. Balinski, Michel L., Pivot Transformations, Dual Linear Programs, and Simplex Methods, mimeographed lecture notes, Princeton University, 1962.

3. Gass, S. I., Linear Programming Methods and Applications, Second edition, McGraw-Hill, 1964.

4. Llewellyn, Robert W., Linear Programming, Holt, Rinehart, and Winston, 1964.

VITA

Raymond L. Somers, son of Mr. and Mrs. Robert E. Somers, was born in Bethlehem, Pennsylvania on November 6, 1942. He was graduated in 1960 from Hellertown Junior-Senior High School in Hellertown, Pennsylvania. He enrolled in Princeton University, from which he was graduated magna cum laude in 1964. This was followed by two years study toward a Master of Science in Mathematics at Lehigh University. He is presently employed by the computing center of Johns Hopkins Applied Physics Laboratory in Silver Spring, Maryland.