BASIS FACTORIZATION FOR BLOCK-ANGULAR LINEAR PROGRAMS: UNIFIED THEORY OF PARTITIONING AND DECOMPOSITION USING THE SIMPLEX METHOD

by

Carlos Winkler

TECHNICAL REPORT SOL 74-19

December 1974

SYSTEMS OPTIMIZATION LABORATORY
DEPARTMENT OF OPERATIONS RESEARCH
Stanford University
Stanford, California

Research and reproduction of this report were partially supported by the Office of Naval Research under contract N-00014-67-A-0112-0011; U.S. Atomic Energy Commission Contract AT(04-3) 326 PA #18; and the National Science Foundation, Grant GP 31393X2; Stanford University. This research was completed at the International Institute for Applied Systems Analysis (IIASA), Laxenburg, Austria.

TABLE OF CONTENTS

CHAPTER PAGE

TABLE OF CONTENTS

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

I wish to thank very much Professor George B. Dantzig for introducing me to the field of Large Scale Systems and for encouraging and guiding this research, which greatly benefited from his comments and suggestions. My thanks too, to Professors Frederick S. Hillier and John A. Tomlin for serving on my thesis committee.

My stay in Stanford was supported by scholarships from the Universidad Tecnica Federico Santa Maria, Valparaiso, Chile, and from the Organization of American States. In addition this research was partially supported by the Systems Optimization Laboratory in Stanford and it was completed at the International Institute for Applied Systems Analysis (IIASA), Laxenburg, Austria.

Thanks also to the typing and draughting staff (IIASA) for the preparation of the final draft and to Gail A. Lemmond (Stanford) who took care of all the details in Stanford.

Last but not least, this work would not have been possible without the constant support and encouragement of my wife Cristina.

CHAPTER 1

GENERAL

1.1. Introduction and Summary

Many algorithms have been proposed over the years to take advantage of the special structure of block-angular linear systems. Among those not based on the Dantzig-Wolfe decomposition principle [13], we have Dantzig and Van Slyke's Generalized Upper Bounding [12], Balas' Infeasibility Pricing Method [1], Rosen's Primal Partitioning Method [32], and the methods of Kaul [22], Müller-Mehrbach [27], Bennett [4], Orchard-Hays [29], Ohse [28], Knowles [23], Beale [3], Gass [15], Ritter [31], Hartmann and Lasdon [20], etc.

Grigoriadis and White [17], [19], show that many of the methods for block-angular linear problems with coupling constraints can be viewed as having a common data handling structure and differing only in the strategy used as to the vector pair selected to enter and to leave the basis.

In the following we present a block-angular basis factorization theory that provides a unifying framework for partitioning and decomposition methods not based on the Dantzig-Wolfe decomposition principle and which allows us to view them as special instances of the Simplex Method using basis factorization.

In its generality it gives us an additional degree of freedom, since it can be specialized to any of the previous approaches or alternatively to obtain new variants. This, in addition to a more thorough theoretical understanding, allows us to design specialized algorithms to take full advantage of a particular block-angular structure. For block-angular linear problems with coupling constraints such an algorithm has been programmed with good experimental results (see Appendix A). In addition the theory gives us a good starting point for developing nested factorization methods.

In the remainder of this Chapter we will clarify the sense in which we use certain concepts and terminologies and motivate the development in later Chapters.

In Chapter 2 we develop and validate the General Block-angular Basis Factorization (GBBF) and show how to update the factorized terms in the representation of the inverse as one column substitutes for another in the basis.

Chapter 3 is devoted to the use of the GBBF in the Simplex Method. First its use in performing the backward and forward transformations is analysed and its implications on the choice of simplex strategy are discussed. Then some algorithms are developed that take full advantage of the structure and some consideration is given to alternative ways of implementing them on computer codes.

In Chapter 4 GBBF is used to give a unified presentation of Partitioning and Decomposition methods not based on the Dantzig-Wolfe decomposition principle. Existing methods for block-angular linear problems with coupling constraints, or coupling variables, or both, are shown to be variants of the Simplex Method using GBBF with various strategies as to the vector pair to enter and to leave the basis. Some new strategies that look promising in conjunction with GBBF are presented.

Chapter 5 is devoted to Nested Factorizations that arise in cases where some of the components of the original factorization have also a block-angular structure that can conveniently be factorized further. Nested factorization methods to solve staircase problems are analysed.

Finally in Chapter 6 some comments and conclusions are presented.

Appendix A contains experimental results of tests with a basis factorization algorithm for block-angular linear problems with coupling constraints.

1.2. Concepts, Terminologies and Motivations

It will be convenient to clarify the sense in which we use certain concepts and terminologies:

Simplex Method: Any LP algorithm that follows a path along adjacent basic solutions of the set of linear relations in such a way that no basis is repeated.

Accordingly we distinguish two aspects of the Simplex Method:

Strategy: rules as to how to iteratively move from one basic solution to the next, i.e. criteria as to the vector pair selected to enter and leave the basis.

Data-Handling Structure: information as to what to carry forward and in what form from one iteration to the next.

Improvements in the Simplex Method usually involve changing one or both of the above. For example the data handling structure started in 1947 with the simplex tableau [10]. This was followed by the revised simplex using the explicit inverse and this was soon followed by the product form of the inverse [11].

Each of these data handling structures can be combined with any of the selection strategies such as the usual primal, dual or primal-dual selection criteria [7].

A strategy may be efficient with a given data-handling structure and not so with a different data-handling structure. Moreover criteria such as the greatest change in the objective function [40] may be efficient compared to the others if a tableau simplex structure is used, but some other criteria may be better if the product form structure is used.

With the above concepts in mind, the advantages of a general theory become clearer. If we are able to identify or discover a common body of data-handling structures for general block-angular systems it will be much easier to separate the strategy from the data-handling aspects in the existing algorithms. In an analogous way in identifying the strategies, it will be much easier to get a feeling for the convergence characteristics (efficiency) of the method by first comparing it with alternative strategies for the general simplex method

Also by studying the original matrix structure and how the data-handling aspects are treated, we may be able to identify a strategy that makes best use of both.

Other advantages are that convergence follows from that of the simplex method and this makes it possible to conveniently write one code to test many different methods or strategies.

In the remainder the terminology primal (dual, primal-dual) strategy will be used to refer to the rules used in the primal (dual, primal-dual) simplex method as to how to iteratively move from one basic solution to the next.

<u>Nice Properties under the assumption that the Block-angular Sub-Matrices are Square and Nonsingular</u>*

To motivate the data-handling aspects, consider the "square" block-angular basis structure. (I.1)

$$B_N = \begin{bmatrix} I_0 & \cdots & A_i & \cdots & A_k \\ & \ddots & & \bigcirc & \\ & & B_i & & \\ & \bigcirc & & \ddots & \\ & & & & B_k \end{bmatrix} \qquad \begin{array}{lll} I_0 & m_0 \times m_0 & \text{Identity} \\ B_i & m_i \times m_i & \text{nonsingular} \\ m_T = \sum_{i=0}^{k} m_i & & \end{array}$$

This basis has certain nice properties. To see this, consider first a special case, the matrix $\hat{B}_j$ associated with block j and its inverse:

*The actual basis structure of a block-angular linear program need not, of course, have square blocks along the diagonal but later we will associate with it a basis that does.

$$\hat{B}_j = \begin{bmatrix} I_0 & 0 \cdots & \cdots 0 A_j 0 \cdots & \cdots 0 \\ 0 & I_1 & & \\ & & \ddots & \\ & & B_j & \\ & & & \ddots \\ & & & & I_k \end{bmatrix}, \quad \hat{B}_j^{-1} = \begin{bmatrix} I_0 & & -\bar{A}_j & \\ & I_1 & & \\ & & \ddots & \\ & & B_j^{-1} & \\ & & & \ddots \\ & & & & I_k \end{bmatrix} \qquad (I.2)$$

where $\bar{A}_j = A_j B_j^{-1}$.

We can now express

$$B_N = \prod_{i=1}^{k} \hat{B}_i \quad \text{and} \quad B_N^{-1} = \prod_{i=1}^{k} \hat{B}_i^{-1} \qquad (I.3)$$

where the terms forming the products can be commuted.

Some of these nice properties of the square block-angular basis are:

1) Instead of inverting one big matrix of dimension $m_T \times m_T$, one can invert k small matrices of dimension $m_i \times m_i$ $(i = 1,\ldots k)$.

2) To represent (in terms of basis) an incoming vector d "belonging" to block j, we have that

$$\hat{d} = B_N^{-1} d = \hat{B}_j^{-1} d \qquad (I.4)$$

i.e. we need only the inverse associated with the block. This implies savings in computations and data transfer. To show (I.4) we prove instead $d = B_N \hat{d} = \hat{B}_j \hat{d}$. Partitioning d and $\hat{d}$, we may write this out more explicitly

$$\hat{d}_0 + \sum_{i=1}^{k} A_i \hat{d}_i = d_0$$

$$B_i \hat{d}_i = d_i \qquad i = 1,\ldots,k \tag{I.5}$$

But since d is in block j, for $i \neq j$ $d_i = 0$, which implies $\hat{d}_i = 0$ for $i \neq j$, since the B_i's are nonsingular. Hence equations (I.5) reduce to

$$\hat{d}_0 + A_j \hat{d}_j = d_0$$

$$B_j \hat{d}_j = d_j \tag{I.6}$$

But this corresponds to $\hat{B}_j \hat{d} = d$ and hence $\hat{d} = \hat{B}_j^{-1} d$.

3) The "price" vector Π is defined by $\Pi B_N = C$ (see Ch. 3). To calculate the partition Π_j of Π corresponding to block j, we need only to compute

$$(C_0, 0,\ldots,0, \Pi_j, 0,\ldots,0) = (C_0, 0,\ldots,0, C_j, 0,\ldots,0)\hat{B}_j^{-1} \tag{I.7}$$

implying the same kind of savings as in 2). Relation (I.7) follows from the structure of B_N, which implies $\Pi_0 = C_0$ and $\Pi_0 A_i + \Pi_i B_i = C_i$ for $i = 1,\ldots,k$.

Our motivation then is to preserve as much of these nice properties (mentioned above) as we can for the more general case when the block-angular basis arises from problems having either coupling constraints or coupling variables or both.

CHAPTER 2

BLOCK-ANGULAR BASIS FACTORIZATION THEORY

2.1. The Problem

Consider the block-angular linear problem with coupling constraints and variables

$$\max z^{(+)}$$

$$
\begin{array}{lrcl}
\text{s.t.} & U_z + D_0x_0 + D_1x_1 + \cdots\cdots D_kx_k + H_0y & = & b_0 \\
 & G_1x_1 \qquad\qquad\qquad + H_1y & = & b_1 \\
 & \ddots \qquad\qquad\qquad \vdots & & \vdots \\
(P) & G_kx_k + H_ky & = & b_k \\
\end{array}
$$

$$(x_0,\ x_1,\ldots,x_k,\ y) \geq 0$$

where U is $m_0 \times 1$, D_i is $m_0 \times n_i$ $i = 0,1,\ldots,k$, H_i is $m_i \times n_{k+1}$ $i = 0,1,\ldots,k$, G_i is $m_i \times n_i$ $i = 1,\ldots k$, b_i is $m_i \times 1$ $i = 0,1,\ldots k$, x_i is $n_i \times 1$ $i = 0,1,\ldots k$, y is $n_{k+1} \times 1$ and z scalar.

(+) We assume that for min $\{cx\colon Ax = b,\ x \geq 0\}$ we let $z = -cx$ and solve max $\{z\colon \begin{pmatrix} 1 & c \\ & A \end{pmatrix}\begin{pmatrix} z \\ x \end{pmatrix} = \begin{pmatrix} 0 \\ b \end{pmatrix},\ x \geq 0$ and for max $\{cx\colon\ Ax = b,\ x \geq 0\}$ we let $z = cx$ and max $\{z\colon\ \begin{pmatrix} -1 & c \\ & A \end{pmatrix}\begin{pmatrix} z \\ x \end{pmatrix} = \begin{pmatrix} 0 \\ b \end{pmatrix},\ x \geq 0\}$. Thus, both for minimizing and maximizing, we can use a negative reduced cost criteria to indicate that a non-basic column will improve the current solution if it replaces one in the basic set. This will be assumed throughout.

We assume that each of the matrices G_i and (UD_0) have rank equal to their row count. This can always be achieved (if necessary) by augmenting the system with artificial variables with appropriate coefficient structure.

The constraints $Uz + \sum_{i=0}^{k} D_i x_i + H_0 y = b_0$ will be called coupling constraints and rows corresponding to then will also be referred to as common rows. Similarly the y variables will be called coupling variables.

2.2. Constructive Development of the Block-angular Basis Factorization

Let J_i = {set of indices (of columns) associated with activities in block i} $i = 1,\ldots,k$

J_0 = {indices of columns in D_0}

J_{k+1} = {indices associated with activities y}

$A|_J$ = restriction of matrix A to columns with indices in set J.

Let B_T be a basis for problem (P) and suppose that M is the set of indices of basic columns. Let

$$L_i = M \cap J_i \qquad \text{and consider}$$

$$G_i|_{L_i} \qquad i = 1,\ldots,k$$

Let K_i be the indices of a maximum set of linearly independent columns in $G_i|_{L_i}$ and*

$$K = \bigcup_{i=1}^{k} K_i \quad .$$

By assumption the rank of G_i is equal to its row count, so that we can augment the columns of $G_i|_{K_i}$ by including enough other columns of G_i to form a basis B_i of linearly independent columns in G_i. Let M_i be the indices of the set of columns of G_i forming B_i (i.e. $B_i = G_i|_{M_i}$).

The nonsingular matrix

$$B_N = \begin{bmatrix} I_0 & A_1 & \cdots & A_k \\ & B_1 & & \\ & & \ddots & \\ & & & B_k \end{bmatrix} \qquad \text{where } A_i = D_i|_{M_i}$$

is square block-angular and has the "nice" properties discussed earlier. We now express the relationship between B_T and B_N in the form of a product:

$$B_T = B_N B_A \tag{2.1}$$

where

$$B_A = B_N^{-1} B_T \tag{2.2}$$

---

* For many practical applications it has been observed that the number of elements in K is close to $\sum_{i=1}^{k} m_i$. It is this that makes the factorization scheme which follows efficient in practice.

The columns of B_A Corresponding to K are unit columns so that it is convenient for discussion purposes here to permute its rows and columns so that the units form a submatrix identity I in the lower right partition.

$$\text{permuted } B_A = {}_pB_A = \begin{pmatrix} B_W & \\ V & I \end{pmatrix} \tag{2.3}$$

where as we have noted the number of columns in I is for an important class of practical applications close to that of $\sum_{i=1}^{k} m_i$.

Columns corresponding to K (or to I above) are called trivial, the remaining, $M \cap K^c$ (where K^c is the complement of K) are called non-trivial. We refer to the upper left matrix as the "Working Basis" or "WB" for short.

Without loss of generality we assume that

$${}_pB_A = P\ B_A P \tag{2.4}$$

(2.4)

where P is a permutation matrix satisfying PP = I.

We can further factorize ${}_pB_A$ into

$${}_pB_A = \begin{pmatrix} B_W & \\ V & I \end{pmatrix} = \begin{pmatrix} B_W & \\ & I \end{pmatrix} \begin{pmatrix} I_W & \\ V & I \end{pmatrix} = {}_p\hat{B}_W\ {}_p\hat{V} \tag{2.5}$$

and hence express the basis in factorized form as the product

$$B_T = B_N\ P\ {}_p\hat{B}_W\ {}_p\hat{V}\ P \tag{2.6}$$

or by permuting again the factors ${}_p\hat{B}_W$ and ${}_p\hat{V}$ (i.e. $\hat{B}_W = P_p\hat{B}_W P$ and $\hat{V} = P_p\hat{V}P$)

$$B_T = B_N \hat{B}_W \hat{V} \tag{2.7}$$

<u>Lemma 1</u>: B_W, the Working Basis, is nonsingular.

<u>Proof</u>: Obviously B_W is square. Hence

$$0 \neq \det B_T = \det B_N \det \hat{B}_W \det \hat{V}$$

Since permutations do not change the absolute value of the determinant

$$|\det \hat{V}| = |\det {}_p\hat{V}| = 1$$

and

$$\det \hat{B}_W \neq 0 \quad ;$$

Moreover, by permuting $\hat{B}_W$ we get

$$0 \neq \det \begin{pmatrix} B_W & \\ & I \end{pmatrix} = \det B_W \quad .$$ ||*

Hence we can work with the following factorized representation for the inverse

$$B_T^{-1} = \hat{V}^{-1} \hat{B}_W^{-1} B_N^{-1} \tag{2.8}$$

---

* Double slashes will be used for end of proof.

For applications it is not necessary to permute the matrices $\hat{B}_W$ and $\hat{V}$ to have rows and columns of B_W and V (see (2.5)) in the upper left and lower left corners respectively. However for the development of the formulas for updating the factorized representation of the inverse when one column replaces another in the basis it will be convenient, for notational purposes, to work with the permuted matrices. Therefore let

$$_pB_T = P\ B_T\ P \quad \text{and} \quad _pB_N = P\ B_N\ P \tag{2.9}$$

Then from (2.6)

$$_pB_T = {_pB_N}\ {_p\hat{B}_W}\ {_p\hat{V}} \tag{2.10}$$

Notice that expression (2.10) differs from (2.7) only in that all terms are permuted. Thus, for simplicity, in what follows the left subscript p will be dropped when working with the permuted matrices, since this will be clear from the context.

2.3. Some Properties of the Factorized Representation of the Inverse

Recalling the nice properties of square block-angular systems, we see for the general block-angular case that in addition to the block-inverses we have to carry the inverse of the Working Basis and the matrix V of $\hat{V}$. Hence under the assumption that the dimension m_W of B_W is "small" relative to m_T, or more precisely the number of non-zeroes in V and B_W^{-1} (or some representation of B_W^{-1}) is "small", then the additional amount of information stored and manipulated will be small. In particular with regard to preserving as much as possible of the nice properties.

1) Instead of inverting one big $m_T \times m_T$ matrix we can still invert and maintain k small $m_i \times m_i$ matrices $(i = 1,\ldots,k)$. However, in addition an $m_w \times m_w$ Working Basis will need to be inverted and maintained; also V will be needed.

2) The first step in updating a vector from block j proceeds the same as that described earlier - and hence the same computational advantages carry through. However, in addition we have to use B_w^{-1} and V. Hence if, as we have assumed, the nonzeroes in B_w^{-1} and V are low relative to those of the block inverses $\hat{B}_i$, $i \neq j$, not required in the first step, we will get savings in the forward transformation over a direct representation of B_T^{-1} .

3) For calculating a Π_j the situation is similar to that of the updates in (2). As will be shown in Chapter 3 there is the additional advantage that when the basic variables which correspond to columns not in the Working Basis are feasible, then the $\hat{V}$ matrix is not needed in the backward transformation. This is always the case in Phase 2.

Because no simple statement can be made at this point on how much work is required to update the factorized representation of the inverse (after the replacement of one column in the basis by another), we will defer discussion of this to later. In section 2.4 we show how to do this updating efficiently.

Thus with the additional effort to maintain and to make use of B_W^{-1} and V, we can carry over much of the desirable properties of independent square block-angular problems. If the dimension m_W of B_W is not too large (relatively) and the additional work in updating the factorized representation of the inverse is not too excessive, we can expect the block-angular factorization method to be more efficient than working directly on the basis B_T using general methods.

We now explore more deeply these points. First we introduce some notation. We classify columns to be either Type A or Type B.

Type A: Those that, except for the common rows, have non-zeroes in rows corresponding to at most one block $i = 0,1,\ldots,k$, i.e. those with indices belonging to $J_A = \bigcup_{i=0}^{k} J_i$.

Type B: otherwise, i.e. $J_B = J_{k+1}$

Furthermore, the basic columns of Type A are further subclassified into

Type A1: Those basic columns associated with block i, for $i = 1,\ldots,k$ (i.e. Type A columns) that belong to their own block basis B_i.

Type A2: Otherwise, i.e. basic columns associated with block i, for $i = 1,\ldots,k$ that belong to the Working Basis.

Let B_{wO} be the matrix of columns common to B_T and the Working Basis. Partition B_{wO} according to

$$B_{wO} = \begin{array}{c} \text{common rows}\left\{\right. \\ \\ \\ \end{array} \left[\begin{array}{c|c} B_{OB} & B_{OA} \\ \hline U_B & U_A \\ \hline \tilde{V}_B & \tilde{V}_A \end{array}\right] \begin{array}{c} \left.\right\} \text{rows in WB} \\ \\ \end{array}$$

$$\uparrow \text{type B column} \qquad \uparrow \text{type A columns}$$

Let

$$\tilde{B}_{wO} = B_N^{-1} B_{wO} = \begin{pmatrix} B_w \\ V \end{pmatrix} = \begin{bmatrix} \hat{B}_{OB} & \hat{B}_{OA} \\ \hat{U}_B & \hat{U}_A \\ V_B & V_A \end{bmatrix} \qquad (2.11)$$

i.e.

$$B_W = \begin{bmatrix} \hat{B}_{OB} & \hat{B}_{OA} \\ \hat{U}_B & \hat{U}_A \end{bmatrix} \qquad (2.12)$$

partitioned as above.

We call a column that is in B_N but not in B_T a pseudobasic column; its corresponding variable will be referred to as pseudobasic also.

Recall from section 2.2 that K_i was chosen to have the indices of a maximum set of linearly independent columns in $G_i|_{L_i}$ and that $M \cap K^c$ contains the indices of columns in the Working Basis (where K^c is the complement of $K = \bigcup_{i=1}^{k} K_i$).

Observe that:

a) The maximum sets of linearly independent columns in $G_i|_{L_i}$ $(i = 1,\ldots,k)$ need not be unique. If this is the case we could choose the indices of columns in any such set to be in K_i and hence in K. Thus alternative factorizations are possible that lead to different Working Basis' of the same dimension.

b) If K_i is not required to contain the indices of a maximum set, but only of a subset of linearly independent columns in $G_i|_{L_i}$, then the resulting factorization would have a Working Basis of higher dimension (less indices in K, more in K^c, i.e. of columns in WB).

Following the constructive procedure in section 2.2 we always obtain a Working Basis with the smallest possible dimension. For the case when one column replaces another in the basis, we want to obtain the new factorization from the old one. Herefore it is convenient to establish some easy to check conditions for a Working Basis being minimal (i.e. there being no alternative factorization giving rise to a WB of smaller dimension).

Theorem 1: The Working Basis is minimal if and only if $\hat{U}_A = 0$

Proof: Let B_W be minimal. Assume (on the contrary) that $\hat{U}_A \neq 0$. Pick a non-zero element of $\hat{U}_A$ and suppose it is on row j in the partition corresponding to some block i. This non-zero element is used as a pivot to replace the pseudobasic columns of block basis i associated with row j. The new B_N now includes one more vector previously in the Working Basis. Thus, the new WB will have one less non trivial vector - a contradiction !

$$\therefore \qquad \text{WB minimal} \Longrightarrow \hat{U}_A = 0$$

Now suppose $\hat{U}_A = 0$ and the Working Basis is not minimal. Then for at least one block i (for i = 1,...,k) there is a set of linearly independent columns among those with indices in $M_i \cup L_i$ that constitutes a basis and which does not include at least one of the pseudobasic variables (i.e. those with indices in $M_i \cap L_i^c$ where L_i^c is the complement of L_i). Suppose this new block basis B_i^{new} is partitioned as

$$B_i^{new} = \begin{pmatrix} {}_1B^1 & {}_1B^2 & {}_1B^3 & {}_1B^4 \\ {}_2B^1 & {}_2B^2 & {}_2B^3 & {}_2B^4 \\ {}_3B^1 & {}_3B^2 & {}_3B^3 & {}_3B^4 \\ {}_4B^1 & {}_4B^2 & {}_4B^3 & {}_4B^4 \end{pmatrix}$$

with superscripts

1 : basic columns that remained

2 : pseudobasic columns that remained

3 : new columns (previously in WB partition) that have replaced basic columns (possible none)

4 : New columns (previously in WB) that have replaced pseudobasic columns (at least one)

and left subscript

1 : rows in which basic columns that remain were basic

2 : rows in which pseudobasic columns that remain were basic

3′ : rows in which basic columns replaced were basic

4 : rows in which pseudobasic columns replaced were basic.

Then pre multiplying B_i^{new} by B_i^{-1}

$$B_i^{-1}\ B_i^{new} = \begin{pmatrix} P_1 & 0 & {}_1\hat{B}^3 & {}_1\hat{B}^4 \\ 0 & P_2 & {}_2\hat{B}^3 & {}_2\hat{B}^4 \\ 0 & 0 & {}_3\hat{B}^3 & {}_3\hat{B}^4 \\ 0 & 0 & {}_4\hat{B}^3 & {}_4\hat{B}^4 \end{pmatrix}$$

where P_1 and P_2 are permutations of Identities.

But $\begin{pmatrix} {}_2\hat{B}^3 & {}_2\hat{B}^4 \\ {}_4\hat{B}^3 & {}_4\hat{B}^4 \end{pmatrix}$ has as coefficients those of columns that were in the WB and corresponding to rows in the U partition. Thus they constitute a subset of coefficients of $\hat{U}_A$ and hence they are all 0. But this implies that rows corresponding to left subscript 4 (at least one) are 0 and hence that B_i^{new} is singular, which is a contradiction.

$$\therefore \quad \hat{U}_A = 0 \Longrightarrow \text{WB minimal}$$

and $\quad$ $$\text{WB minimal} \Longleftrightarrow \hat{U}_A = 0 \quad .$$ ||

<u>Lemma 2</u>: The dimension m_w of minimal Working Basis satisfies

$$m_w \leq m_0 + m_B \leq m_0 + n_{k+1} \tag{2.13}$$

where m_B is the number of coupling variables in the basis.

<u>Proof</u>: For B_w, a minimal Working Basis $\hat{U}_A = 0$; thus

$$B_w = \begin{pmatrix} \hat{B}_{OB} & \hat{B}_{OA} \\ \\ \hat{U}_B & 0 \end{pmatrix}$$

Suppose U_B is $m_R \times m_B$ where m_B is the number of type B variables (coupling variables) in the basis. Then

$$m_w = m_0 + m_R$$

Now for B_w to be nonsingular $\hat{U}_B$ has to have full row rank. This requires

$$m_R \leq m_B$$

and hence

$$m_w \leq m_0 + m_B \leq m_0 + n_{k+1} \quad .$$ ||

2.4. Updating the Factorized Representation of the Inverse

Before presenting a procedure for updating the representation of the factorized inverse after the replacement of one column in the basis by another, some results that are needed later will be developed. It will be convenient to use * as a superscript to denote a matrix in the updated representation to distinguish it from the corresponding matrix before the updating. Also, unless stated otherwise, partitions of $m_T \times m_T$ matrices will be assumed to have been permuted to correspond with those of the factorization, i.e. so as to have rows and columns in Working Basis in the upper left corner.

2.4.1. Increase or Reduction in the Dimension of the Working Basis

Some of the update situations will involve an increase or a reduction in the dimension of the Working Basis. In developing the updating formulas for these cases we assume that the inverse is given in product form.

We want to decrease the dimension of a Working Basis when it has a structure like

$$B_W = \begin{pmatrix} B_W^* & \\ v & 1 \end{pmatrix} \text{ to another with structure } \begin{pmatrix} B_W^* & \\ & 1 \end{pmatrix}$$

which can substitute for it in applications of a product form representation.

Let

$$E_v = \begin{pmatrix} I & \\ -v & 1 \end{pmatrix}, \quad \text{then}$$

$$B'_w E_v = \begin{pmatrix} B^*_w & \\ v & 1 \end{pmatrix} \begin{pmatrix} I & \\ -v & 1 \end{pmatrix} = \begin{pmatrix} B^*_w & \\ & 1 \end{pmatrix} \text{and}.$$

$$\begin{pmatrix} B^{*-1}_w & \\ & 1 \end{pmatrix} = E_v^{-1} \; B_w^{-1} \tag{2.14}$$

and hence it is accomplished by adding an elementary row eta to the representation of the inverse.

Similarly to add a row, i.e. to get from

$$B_w \text{ or } \begin{pmatrix} B_w & \\ v & 1 \end{pmatrix} \text{ to } \begin{pmatrix} B_w & \\ v & 1 \end{pmatrix}, \text{ then}$$

$$\begin{pmatrix} B_w & \\ v & 1 \end{pmatrix}^{-1} = E_v \begin{pmatrix} B_w^{-1} & \\ & 1 \end{pmatrix} \tag{2.15}$$

and again is accomplished through an elementary row transformation.

2.4-2 General Updating Formulas

Theorem 2: Let E, E_N be the elementary matrices that update B_T^{-1} and B_N^{-1}, i.e. $B_T^{*-1} = EB_T^{-1}$ and $B_N^{*-1} = E_N B_N^{-1}$, and let

$$E = \begin{pmatrix} E_1 & E_2 \\ E_3 & E_4 \end{pmatrix} \text{ and } E_N = \begin{pmatrix} E_N^1 & E_N^2 \\ E_N^3 & E_N^4 \end{pmatrix}$$

correspond to the partitioning used in the factorization (which is assumed not to change). Further suppose $E_2 = 0$ or $E_N^3 = 0$, then

$$B_W^{*-1} = (E_1 - E_2 V) B_W^{-1} (E_N^1)^{-1} \quad . \tag{2.16}$$

Proof: We have

$$B_T^{*-1} = \hat{V}^{*-1} \hat{B}_W^{*-1} B_N^{*-1} = EB_T^{-1} = E\hat{V}^{-1} \hat{B}_W^{-1} B_N^{-1}$$

or

$$\hat{B}_W^{*-1} = \hat{V}^* E \hat{V}^{-1} \hat{B}_W^{-1} B_N^{-1} B_N^*$$

But

$$B_N^{-1} B_N^* = E_N^{-1} \quad . \text{ Let}$$

$$\tilde{E}_N = \begin{pmatrix} \tilde{E}_N^1 & \tilde{E}_N^2 \\ \tilde{E}_N^3 & \tilde{E}_N^4 \end{pmatrix} = E_N^{-1} \quad \text{, then}$$

$$\hat{B}_W^{*-1} = \hat{V}^* E \hat{V}^{-1} \hat{B}_W^{-1} \tilde{E}_N$$

and writing this product in partitioned form

$$\begin{pmatrix} B_W^{*-1} & \\ & I \end{pmatrix} = \begin{pmatrix} I_W & \\ V^* & I \end{pmatrix}\begin{pmatrix} E_1 & E_2 \\ E_3 & E_4 \end{pmatrix}\begin{pmatrix} I_W & \\ -V & I \end{pmatrix}\begin{pmatrix} B_W^{-1} & \\ & I \end{pmatrix}\begin{pmatrix} \tilde{E}_N^1 & \tilde{E}_N^2 \\ \tilde{E}_N^3 & \tilde{E}_N^4 \end{pmatrix}$$

and now restricting ourselves to the rows and columns in the Working Basis

$$B_W^{*-1} = (E_1\ E_2)\begin{pmatrix} I_W & \\ -V & I \end{pmatrix}\begin{pmatrix} B_W^{-1} & \tilde{E}_N^1 \\ & \tilde{E}_N^3 \end{pmatrix}$$

$$B_W^{*-1} = \big((E_1 - E_2V), E_2\big)\begin{pmatrix} B_W^{-1} & \tilde{E}_N^1 \\ & \tilde{E}_N^3 \end{pmatrix}$$

$$B_W^{*-1} = (E_1 - E_2V)B_W^{-1}\,\tilde{E}_N^1 + E_2\tilde{E}_N^3$$

Since E_N is an elementary column matrix we have

$$E_N = \left[\begin{array}{l} \begin{pmatrix} I & E_N^2 \\ & E_N^4 \end{pmatrix} \Longrightarrow E_N^{-1} = \begin{pmatrix} I & -E_N^2(E_N^4)^{-1} \\ & (E_N^4)^{-1} \end{pmatrix} \Longrightarrow \begin{cases} \tilde{E}_N^1 = (E_N^1)^{-1} \\ \tilde{E}_N^3 = 0 = -E_N^3(E_N^1)^{-1} \end{cases} \\ \text{or} \\ \begin{pmatrix} E_N^1 & \\ E_N^3 & I \end{pmatrix} \Longrightarrow E_N^{-1} = \begin{pmatrix} (E_N^1)^{-1} & \\ -E_N^3(E_N^1)^{-1} & I \end{pmatrix} \Longrightarrow \begin{cases} \tilde{E}_N^1 = (E_N^1)^{-1} \\ \tilde{E}_N^3 = E_N^3(E_N^1)^{-1} \end{cases} \end{array}\right.$$

Hence in either case we get the same expressions for $\tilde{E}_N^1$ and $\tilde{E}_N^3$. Subsituting above

$$B_W^{*-1} = (E_1 - E_2V)B_W^{-1}(E_N^1)^{-1} - E_2E_N^3(E_N^1)^{-1} \quad .$$

But under the conditions in the hypothesis $E_2E_N^3 = 0$, so that

$$B_W^{*-1} = (E_1 - E_2V)B_W^{-1}(E_N^1)^{-1} \quad . \qquad ||$$

As will be seen in section 2.4-3, most of the update situations can be arranged to satisfy the conditions of the above theorem and usually $(E_N^1)^{-1} = I$ and $E_2 = 0$, so that $B_W^{*-1} = E_1B_W^{-1}$, or under conditions such that it simplifies to $B_W^{*-1} = (I_W - \eta v)B_W^{-1}$. The following results will always allow us to express these updating relationships as product of elementary transformation matrices.

<u>Theorem 3</u>: Let $\eta \varepsilon R^m$ be a column vector and $v \varepsilon R^m$ a row vector. Suppose $v\eta - 1 \neq 0$, then $I_m - \eta v$ is non-singular. Furthermore if $v_P \neq 0$ is a component of v then

$$(I_m - \eta v) = E_{R_1} E_{C_1} E_{R_2} \qquad (2.17)$$

where E_{R_1} and E_{R_2} are the elementary row matrices given by

$$E_{R_1} = \begin{pmatrix} I_1 & & \\ -\,{}^{v_1}/_{v_P} & -\,{}^{a}/_{v_P} & -\,{}^{v_2}/_{v_P} \\ & & I_2 \end{pmatrix} \qquad (2.18)$$

$$E_{R_2} = \begin{pmatrix} I_1 & & \\ bv_1 & bv_P & bv_2 \\ & & I_2 \end{pmatrix} \tag{2.19}$$

$a \neq 0$, $b \neq 0$ arbitrary constants, $v = (v_1, v_P, v_2)$, and E_C is an elementary column matrix given by

$$E_{C_1} = \begin{pmatrix} I_1 & -(^1/_b)\eta_1 & \\ & \bar{\eta}_P & \\ & -(^1/_b)\eta_2 & I_2 \end{pmatrix} \tag{2.20}$$

with $$\bar{\eta}_p = -\frac{1-v\eta}{ab} \quad \text{and} \quad \eta = \begin{pmatrix} \eta_1 \\ \eta_P \\ \eta_2 \end{pmatrix}. \tag{2.21}$$

Proof: Note that if $v = 0$, the theorem is trivially true. If not then there exists some $v_P \neq 0$. It is easy to verify by direct multiplication that

$$(I_m - \eta v) = E_{R_1} E_{C_1} E_{R_2}$$

and therefore

$$\det(I_m - \eta v) = \det E_{R_1} \cdot \det E_{C_1} \cdot \det E_{R_2}$$

$$= \left(-\frac{a}{v_P}\right) \left(\frac{1-v\eta}{ab}\right)\left(bv_P\right) \neq 0$$

if $(1 - v\eta) \neq 0$. ||

Corollary:

$$(I - \eta v)^{-1} = \begin{pmatrix} I_1 & & \\ -\frac{v_1}{v_p} & \frac{1}{b\,v_p} & -\frac{v_2}{v_p} \\ & & I_2 \end{pmatrix} \begin{pmatrix} I_1 & \frac{\eta_1}{b_\eta} & \\ & 1/\eta & \\ & \frac{\eta_2}{b_\eta} & I_2 \end{pmatrix} \begin{pmatrix} I_1 & & \\ -\frac{v_1}{a} & -\frac{v_p}{a} & -\frac{v_2}{a} \\ & & I_2 \end{pmatrix}$$

Another important point to consider is the possibility of an alternative minimal basis. Especially of interest is the case where we can replace a vector in a block basis by a vector in the Working Basis belonging to the same block.

Theorem 4: Let column b with index in J_i be the basic activity associated with row r in forming the inverse of block i. Let $v_r = (v_B, v_A)$ be the corresponding row of V. Then if $v_A \neq 0$, any one of the columns corresponding to a nonzero component of v_A, say v_{A_j} (i.e. the j-th column in B_w) can be exchanged with b to give a new basis for block i and a new Working Basis. Moreover

$$B_w^{*-1} = E_R B_w^{-1} \tag{2.23}$$

where E_R is an elementary row matrix given by

$$E_R = \begin{bmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & -v_r & & \\ & & & 1 & \\ & & & & \ddots \\ & & & & & 1 \end{bmatrix}$$

and if B_W is minimal, so is B_W^*.

Proof: Recall that by (1.4) for any column from some block P,

$$\hat{d} = B_N^{-1}d = \hat{B}_P^{-1}d$$

and hence by (1.6) $\hat{d}$ can have nonzeros only in the common rows and in the rows of its own block P. It follows, since all columns in the partition corresponding to v_A are of Type A, that under the conditions of the hypothesis any column corresponding to a nonzero component v_{A_j} of v_A must belong to block i and can replace the activity basic in row r of the block basis since its pivot element is different from 0. Recall that B_W minimal implies $\hat{U}_A = 0$ so that

$$\begin{pmatrix} B_W \\ \\ V \end{pmatrix} = \begin{pmatrix} \hat{B}_{OB} & \hat{B}_{OA} \\ \hat{U}_B & \hat{U}_A \\ V_B & V_A \end{pmatrix} = \begin{pmatrix} \hat{B}_{OB} & \hat{B}_{OA} \\ \hat{U}_B & 0 \\ V_B & V_A \end{pmatrix}$$

and hence the updated vector j that will be exchanged with b has zeros in the rows corresponding to the partition $(\hat{U}_B\hat{U}_A)$. Thus the eta vector will have zeros there and all the remaining columns will be unchanged in these rows. Also the representation of the exchanged vector b in terms of the new block basis corresponds to the eta vector so that $U_A^* = 0$ and B_W^* minimal.

The exchange corresponds to a simple permutation of columns, so that $B_T^* = B_T E$,E a simple permutation matrix, for which $E^{-1} = E$ and $B_T^{*-1} = EB_T^{-1}$. Also $B_N^{*-1} = E_N B_N^{-1}$ with

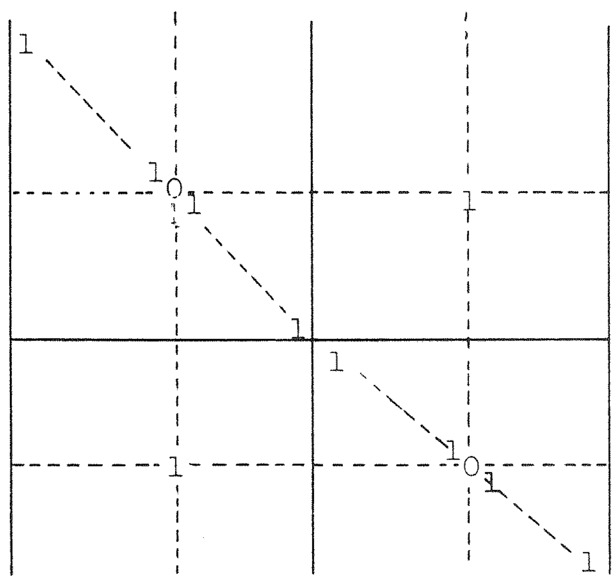
$$E_N = \begin{pmatrix} I & E_N^3 \\ & E_N^4 \end{pmatrix}$$

since the pivoting occurs in a row not in the Working Basis. Thus the updating formula (2.16) becomes

$$B_W^{*-1} = (E_1 - E_2 V) B_W^{-1}$$

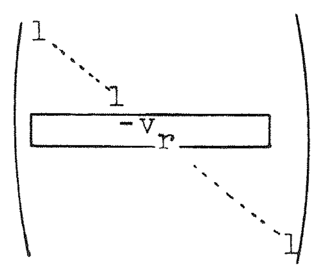
Also

$$E = \begin{pmatrix} E_1 & E_2 \\ E_3 & E_4 \end{pmatrix} =$$



So that E_2V has zeros in all rows except row j, and E_1 is an identity except for row j which is zero. Hence

$(E_1 - E_2V) =$



$= E_R$

$$(E_1 - E_2 V) = \begin{pmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ \hline & & -v_{r_1} & & \\ \hline & & & \ddots & \\ & & & & 1 \end{pmatrix} = E_R$$

and

$$B_W^{*-1} = E_R B_W^{-1} \quad .$$ ||

2.4-3 <u>An Updating Procedure</u>

The replacement of one outgoing column (OC) from the basis by another, the incoming column (IC) gives rise to four somewhat different updating cases:

1) IC of Type A and OC in Working Basis.
2) IC of Type A and OC in B_N.
3) IC of Type B and OC in Working Basis.
4) IC of Type B and OC in B_N.

In Fig. 1 we give a flow-sheet of an efficient updating procedure covering all four cases for the factorized representation of the inverse after the replacement of one column in the basis by another.

We can compactly state some of the important features of the updating procedure in the form of a theorem, and then develop it in greater detail in a constructive way in the proof.

<u>Theorem 5 (Updating Procedure)</u>: The flow-sheet in Fig. 1 gives a valid procedure for updating the factorized representation of

the inverse after the replacement of one column in the basis by another. In particular, if the old Working Basis was minimal so will be the new one, and [except when a pseudobasic variable is driven out of some block basis to keep the Working Basis minimal (see ** in Fig. 1)] at most one block inverse needs to be updated due to the replacement of only one column in it by another [in the exception at most two columns are replaced in the block basis'].

<u>Proof (Validation of the Updating Procedure)</u>: Referring to Fig. 1 we point out that since all tests are of the yes-no type it suffices to show that each path gives a correct updating procedure for the case it involves.

<u>Case I. Incoming Column of Type B</u>

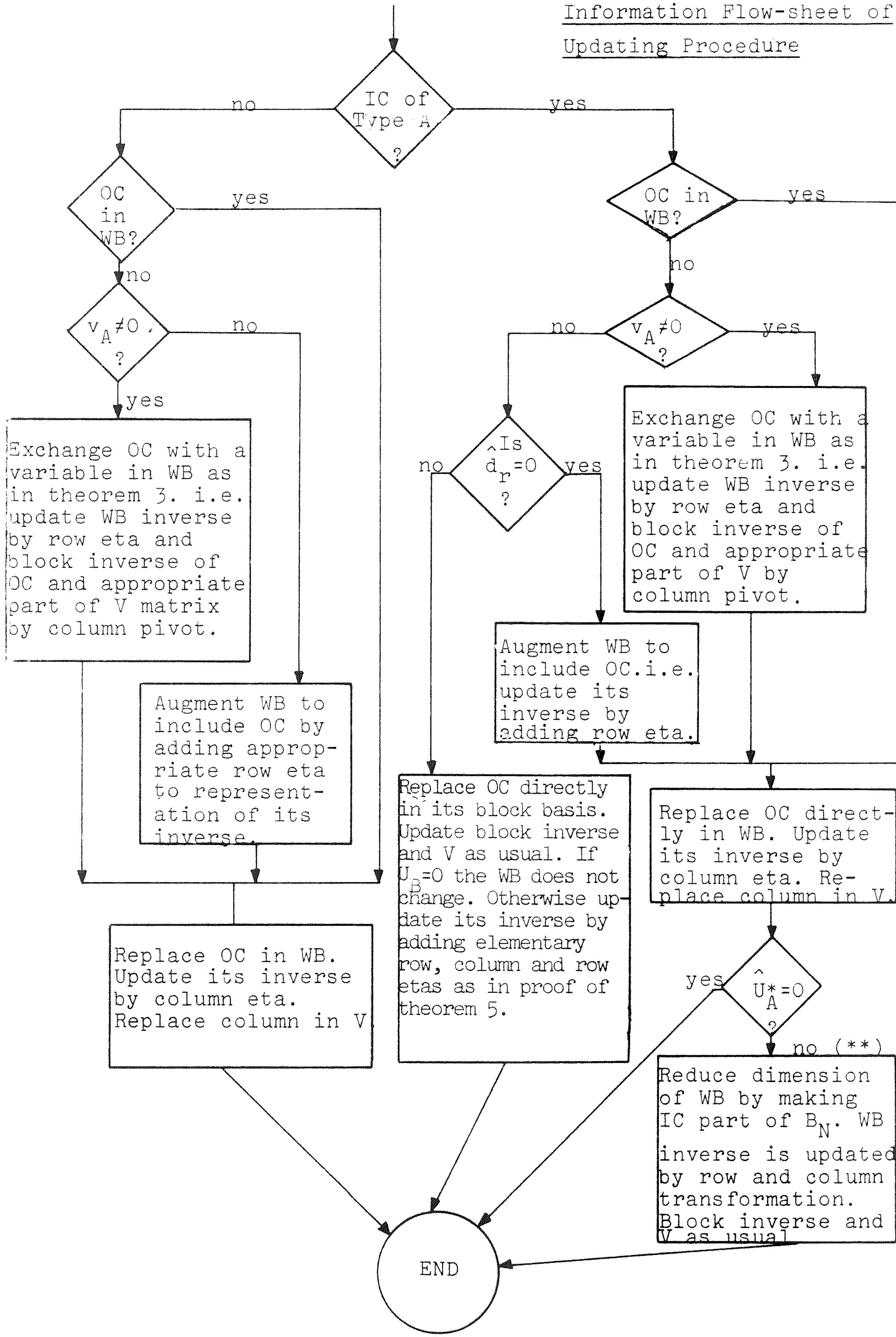
<u>Case I-a. Outgoing Column in Working Basis</u>

Since we start with a minimal Working Basis, $\hat{U}_A = 0$. Letting $B_N^* = B_N$, the updating corresponds to changing one column in $\tilde{B}_{wO}$ (see (2.11)). If the Outgoing Column is of Type B, then $\hat{U}_A^* = \hat{U}_A = 0$ and the new Working Basis is minimal. If the Outgoing Column is of Type A2, then after the exchange, $\hat{U}_A^*$ is equal to $\hat{U}_A$ without the column corresponding to the Outgoing Column (which is now in $\hat{U}_B^*$) and hence $\hat{U}_A^* = 0$ and the new Working Basis is minimal.

The elementary column matrix E that updates $B_T^{*-1} = EB_T^{-1}$ has its pivot element in some row in the Working Basis and hence

FIGURE 1

Information Flow-sheet of Updating Procedure



$$E = \begin{pmatrix} E_1 & 0 \\ E_3 & I \end{pmatrix}$$

also since $B_N^* = B_N$ we have $E_N = I$ and the updating formula (2.16) in theorem 2 reduces to

$$B_W^{*-1} = E_1 B_W^{-1}$$

Also V changes only in the column of the outgoing variable, which is replaced by the partially updated incoming column IC, i.e. B_N^{-1} (IC), restricted to rows not in WB. Notice that all the necessary information is generated during the first step in the forward transformation.

<u>Case I-b. Outgoing Column in some Block Basis</u>

Suppose the Outgoing Variable belongs to block j and corresponds to row r of the inverse. Let $v_2 = (v_B, v_A)$ be the corresponding row of $V = (V_B, V_A)$. If $v_A \neq 0$ pick a component, say $v_{A_i} \neq 0$. By theorem 4 we can assign the outgoing variable to WB and replace it in the block basis by the column corresponding to v_{A_i}, obtaining a new expression for the WB, $B_W^{*-1} = E_R B_W^{-1}$. Besides the block inverse j and V have to be updated by a simple column pivot. After this exchange the outgoing variable is in the Working Basis and we are back to case I-a.

If $v_A = 0$ the dimension of the Working basis is increased by one to include the pivot row and the outgoing variable. This corresponds to going from

$$B_W = \begin{pmatrix} B_{OB} & B_{OA} \\ U_B & U_A \end{pmatrix} \xrightarrow{\text{to}} \begin{pmatrix} B_{OB} & B_{OA} & \\ U_B & U_A & \\ v_B & v_A & 1 \end{pmatrix} = B_W^*$$

As shown in section 2.4-1, the inverse of B_W^* is obtained from B_W^{-1} (see (2.10)) by premultiplying by the elementary row matrix

$$E_R = \begin{pmatrix} I & \\ -v_r & 1 \end{pmatrix}$$

Now the outgoing column is in the WB and we proceed as in case I-a. Since $\hat{U}_A^* = \begin{pmatrix} \hat{U}_A \\ v_A \end{pmatrix} = 0$ the resulting WB is minimal.

Case II. Incoming Column of Type A

Let d be the incoming column and

$$\hat{d} = B_N^{-1} d \qquad , \qquad \bar{d} = B_T^{-1} d$$

Let $\hat{d} = \begin{pmatrix} \hat{d}_O \\ \hat{d}_A \\ \hat{d}_V \end{pmatrix}$, $\bar{d} = \begin{pmatrix} \bar{d}_O \\ \bar{d}_A \\ \bar{d}_V \end{pmatrix}$ be partitioned as B_{wO} (see also (2.11)).

Also let $\hat{d}_r$ and $\bar{d}_r$ be the elements of $\hat{d}$ and $\bar{d}$ on the pivot row.

Case II-a. Outgoing Column in Working Basis

Replace the outgoing column directly in the Working Basis. This corresponds to updating as in case I-a. Let $\bar{B}_W^{-1} = B_W^{*-1} = E_1 B_W^{-1}$. Then for $\bar{B}_W$ we have $\hat{U}_A^* = (\hat{U}_A \hat{d}_A)$, i.e. it consists of zeros except possibly for the column corresponding to the incoming variable (i.e. $\hat{d}_A$). If $\hat{d}_A = 0$ the Working Basis is minimal and we finish by updating V as in case I-a.

Otherwise pick an element $\hat{d}_{A_r} \neq 0$ and use it as pivot to introduce the incoming column into its block basis, displacing a pseudobasic variable. This corresponds to $B_T^* = B_T$ and $B_N^{*-1} = E_N B_N^{-1}$, i.e.

$$E = I \text{ and } E_N = \begin{pmatrix} E_N^1 & \\ E_N^2 & I \end{pmatrix} \text{(see theorem 2)}$$

where

$$E_N^1 = \begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & \eta & & \\ & & & & 1 & \\ & & & & & \ddots \\ & & & & & & 1 \end{pmatrix} \text{ with } \eta_i = \begin{cases} 1/\hat{d}_{A_r} & i = r \\ -\hat{d}_i/\hat{d}_{A_r} & \text{otherwise, for } i \text{ a row index in WB} \end{cases}$$

Accordingly the update formula (2.16) reduces to

$$B_W^{*-1} = \bar{B}_W^{-1}(E_N^1)^{-1} = \bar{B}_W^{-1}\tilde{E}_N$$

where

$$\tilde{E}_N = \begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & \hat{d}_W & & \\ & & & & 1 & \\ & & & & & \ddots \\ & & & & & & 1 \end{pmatrix}$$ and $\hat{d}_W$ is the restriction of $\hat{d}$ to rows in Working Basis.

These changes correpsond to:

1) From B_W to $\bar{B}_W$ replace one column in B_W (without loss of generality the last one)

$$\text{i.e.} \qquad \bar{B}_W = \begin{pmatrix} \tilde{B}_{OB} & \tilde{B}_{OA} & \hat{d}_O \\ \tilde{U}_B & 0 & \hat{d}_A \end{pmatrix} \qquad (2.25)$$

2) From $\bar{B}_W$ to $E_N^1\bar{B}_W$: pivot on row r (without loss of generality the last)

$$\text{i.e.} \qquad E_N^1\bar{B}_W = \begin{pmatrix} \tilde{\tilde{B}}_{OB} & \tilde{\tilde{B}}_{OA} & 0 \\ \tilde{\tilde{U}}_B & 0 & U_r \end{pmatrix} \qquad (2.26)$$

where U_r is a unit vector with unit component on row r. Now we are in the situation of reducing the size of the WB by premultiplying it by an elementary row matrix as discussed in section 2.4-1 to obtain a new minimal WB. Letting B_W^* denote the resulting WB we have

$$B_W^{*-1} = E_R(\bar{B}_W^{-1}\tilde{E}_N) = E_R E_1 B_W^{-1}\tilde{E}_N \qquad (2.27)$$

with

$$E_r = \begin{pmatrix} I_1 & & \\ \hline & v_r/\hat{d}_{A_r} & \\ \hline & & I_2 \end{pmatrix} \text{(see (2.14))} \qquad (2.28)$$

Formula (2.27) gives the expression to update the WB in the case when $\hat{d}_A \neq 0$. It is also necessary to update V. This is done by deleting the column corresponding to the outgoing variable and updating the columns corresponding to type B by applying the same elementary transformation matrix used to update the block inverse when it was modified.

In some computer systems it is inefficient to add new information (in our case etas) to the beginning and end of a file (in our case the eta file). To get around this difficulty we can make use of the following equivalent expression for (2.27).

<u>Proposition 1</u>: Expression (2.27) can also be represented in product form as

$$B_W^{*-1} = E_C E_R E_1 B_W^{-1} \tag{2.29}$$

where E_R and E_1 are as in (2.23) and

$$E_c = \begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & \eta_c & & \\ & & & 1 & & \\ & & & & \ddots & \\ & & & & & 1 \end{pmatrix}, \text{ with } \eta_{c_i} = \begin{cases} -\bar{U}_i / \bar{U}_r & i \neq r \\ 1 / \bar{U}_r & i \neq r \end{cases}$$

and, letting U_r be the r-th unit vector:

$$\bar{U} = (E_R E_1 B_W^{-1}) U_r \tag{2.30}$$

The proof of proposition 1 will be deferred to the end of the section in order not to disrupt the presentation of the

Updating Procedure.

Case II-b. Outgoing Column not in Working Basis

Sub-case II-b-1. $v_A \neq 0$

By theorem 4 it is possible to assign the outgoing column to WB, obtaining a new minimal WB, whose inverse differs from the old one by an elementary row transformation. The exchange also implies updating the appropriate block inverse because of the replacement of the outgoing column by its exchange vector from the WB and to modify V due to the changes in the block inverse. Now we are back to case II-a.

Sub-case II-b-2. $v_A = 0$

Sub-sub-case II-b-2-a. $\hat{d}_r = 0$

Augment the Working Basis to include the OC. As seen before in section 2.4-1, this corresponds to adding an elementary row transformation to the old working basis inverse according to (2.15). Since $v_A = 0$, after replacing the OC by the incoming column in the WB, the form of the WB is given by (2.25) and thus augmenting the Working Basis we fall back to case II-a.

Sub-sub-case II-b-2-b. $\hat{d}_r \neq 0$

In this case the outgoing column can be replaced in its block basis directly by the incoming column. This implies updating the block inverse as usual by adding a column eta to its representation. Columns of V corresponding to Type B variables must also be updated by the same elementary column matrix.

As for the WB, since we are pivoting on a row not in the WB, we have for the elementary matrices in theorem 2

$$E_N = \begin{pmatrix} I_w & E_N^2 \\ & E_N^4 \end{pmatrix} \quad ; \quad E = \begin{pmatrix} I_w & E_2 \\ & E_4 \end{pmatrix}$$

where the columns of E_2 are 0 except for the pivot column which is

$-\left(\frac{\bar{d}_w}{\bar{d}_r}\right)$ with $\bar{d}_w$ the restriction of $\bar{d}$ to rows in WB (recall $\bar{d}_r$ is the pivot element).

Thus (2.16) reduces to

$$B_w^{*-1} = (I_w + \left(\frac{\bar{d}_w}{\bar{d}_r}\right) v_r) B_w^{-1} \tag{2.31}$$

since

$$E_2 V = -\frac{\bar{d}_w}{\bar{d}_r} v_r = -\frac{\bar{d}_w}{\bar{d}_r}(v_B, 0) \tag{2.32}$$

Now there are two possibilities:

A) $v_B = 0$, i.e. $v_r = (v_B, 0) = 0$, and so
$B_w^{*-1} = B_w$, i.e. the WB does not change.

B) $v_B \neq 0$. Consider

$$\bar{d} = \begin{pmatrix} \bar{d}_w \\ \bar{d}_s \end{pmatrix} = B_T^{-1} d = \hat{V}^{-1} \hat{B}_w^{-1} B_N^{-1} d = \hat{V}^{-1} \hat{B}_w^{-1} \hat{d}$$

$$\begin{pmatrix} \bar{d}_w \\ \bar{d}_s \end{pmatrix} = \begin{pmatrix} I_w & \\ -V & I \end{pmatrix} \begin{pmatrix} B_w^{-1} & \\ & I \end{pmatrix} \begin{pmatrix} \hat{d}_w \\ \hat{d}_s \end{pmatrix} = \begin{bmatrix} B_w^{-1}\hat{d}_w \\ \hat{d}_s - VB_w^{-1}\hat{d}_w \end{bmatrix}$$

and

$$\bar{d}_s = \hat{d}_s - V\bar{d}_w$$

in particular for the r-th component

$$\bar{d}_r = \hat{d}_r - v_r\bar{d}_w$$

and since $\bar{d}_r$ is the pivot element it is nonzero and we can divide by it

$$1 = \frac{\hat{d}_r}{\bar{d}_r} - v_r \frac{\bar{d}_w}{\bar{d}_r}$$

i.e.
$$-1 - v_r \frac{\bar{d}_w}{\bar{d}_r} = -\frac{\hat{d}_r}{\bar{d}_r} \neq 0 \qquad (2.33)$$

and hence, since (2.31), (2.33) and $v_B \neq 0$ satisfy its hypothesis we can use theorem 3. Choose some column P with $v_{B_P} \neq 0$, and $b = -1$, $a = \hat{d}_r \neq 0$. Then for this case, according to relations (2.17) through (2.21)

$$B_w^{*-1} = E_{R_2}E_CE_{R_1}B_w^{-1} \qquad (2.34)$$

with

$$E_{R_1} = \begin{pmatrix} I_1 & & & \\ - & v_1 & -v_{B_P} & -v_2 \\ & & & I_2 \end{pmatrix} \quad ; \quad E_C = \begin{pmatrix} I_1 & -\bar{d}_1/\bar{d}_r & \\ & 1/\bar{d}_r & \\ & -\bar{d}_2/\bar{d}_r & I_2 \end{pmatrix} \quad ;$$

$$E_{R_2} = \begin{pmatrix} I_1 & & \\ -v_1/v_{B_P} & -\hat{d}_r/v_{B_P} & -v_2/v_{B_P} \\ & & I_2 \end{pmatrix}$$

where we have partitioned

$$v_r = (v_1, v_{B_P}, v_2) \quad \text{and}$$

$$\bar{d}_w = \begin{pmatrix} \bar{d}_1 \\ \bar{d}_P \\ \bar{d}_2 \end{pmatrix}.$$

To show that the Working Basis in (2.34) is minimal, recall that the same columns remain in the Working Basis, and only their representation in terms of B_N (see (2.2) and (2.3)) may have changed due to pivoting on the r-th row of B_N. But because $v_A = 0$ all type A2 columns have zeros in the pivot row of B_N and remain unchanged. Thus $\hat{U}_A^* = \hat{U}_A = 0$ and the Working Basis in (2.34) is minimal.

This finishes case II-b and now all four possible update cases have been convered. By following all paths in the Updating Procedure we see that at most one vector is replaced among those in the block basis', except when reducing the dimensionality of the Working Basis (see ** box in lower right corner of Fig. 1) in which case it could be two. ||

<u>Proof of Proposition 1</u>: Recall from (2.27) that

$$B_W^{*-1} = E_R E_1 B_W^{-1} \tilde{E}_N$$

Let

$$A = B_W E_1^{-1} E_R^{-1} \tag{2.35}$$

Then

$$B_W^* = \tilde{E}_N^{-1} A \tag{2.36}$$

Without loss of generality we take the pivot column to be the last. Recall that $\tilde{E}_N$ and E_R pivot on the same row, which we again can take to be the last. Then

$$B_W E_1^{-1} = \begin{pmatrix} \hat{B}_{OB} & \hat{B}_{OA} & \hat{d}_O \\ \hat{U}_B & 0 & \hat{d}_A \\ v_B & 0 & \hat{d}_r \end{pmatrix}$$

$$A = B_W E_1^{-1} E_R^{-1} = \begin{pmatrix} \hat{B}_{OB} & \hat{B}_{OA} & \hat{d}_O \\ \hat{U}_B & 0 & \hat{d}_A \\ v_B & 0 & \hat{d}_r \end{pmatrix} \begin{pmatrix} I_O & & \\ & I_B & \\ -v_B/\hat{d}_r & & 1 \end{pmatrix}$$

$$A = \begin{pmatrix} \tilde{\tilde{B}}_{OB} & \hat{B}_{OA} & \hat{d}_O \\ \tilde{\tilde{U}}_B & 0 & \hat{d}_A \\ 0 & 0 & \hat{d}_r \end{pmatrix} \qquad (2.37)$$

and so the r-th row of A is $a_r = (0,\ldots,0,\hat{d}_r)$. Now $\tilde{E}_N^{-1}$ pivots on the last row and reduces the last column of A to the r-th unit vector. Hence

$$B_W^* = \begin{pmatrix} \tilde{\tilde{B}}_{OB} & \hat{B}_{OA} & 0 \\ \tilde{\tilde{U}}_B & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

thus B_W^* is obtained from A by replacing the last column in A by the r-th unit vector. Hence letting

$$\bar{U} = A^{-1}U_r \quad , \quad \eta_{c_i} = \begin{cases} -U_i/\bar{U}_r & i \neq r \\ 1/\bar{U}_r & i \neq r \end{cases}$$

and

$$E_c = \begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & \eta_c & & \\ & & & & 1 & \\ & & & & & \ddots \\ & & & & & & 1 \end{pmatrix}$$

we have

$$B_W^{*-1} = E_c A^{-1}$$

and from (2.35)

$$B_W^{*-1} = E_c E_R E_1 B_W^{-1} \quad . \qquad ||$$