

## APLICACIONES DE UN NUEVO ALGORITMO PARA CÁLCULOS CIENTÍFICOS Y DE DISEÑO

### APPLICATIONS OF A NEW ALGORITHM FOR SCIENTIFIC AND DESIGN CALCULATIONS

M. A. STADTHERR\*, L. E. SCRIVEN

Departamento de Ingeniería Química y Ciencia de Materiales, Universidad de Minnesota  
(Department of Chemical Engineering and Materials Science, University of Minnesota)  
Minneapolis, MN 55455 U. S. A.

\* Dirección actual: Departamento de Ingeniería Química, Universidad de Wisconsin, Madison, WI 53715  
(Present address: Department of Chemical Engineering, University of Wisconsin, Madison, WI 53715)

#### Resumen

Se necesitan procedimientos sistemáticos para encontrar la estrategia más eficiente en la resolución de conjuntos extensos de ecuaciones simultáneas, ya sean estas ecuaciones de diseño en ingeniería o cálculos científicos. El algoritmo de Stadtherr *et al.* (1974 *b*), fija valores de los parámetros (selecciona un conjunto de variables), selecciona la ecuación por la cual se debe calcular cada variable correspondiente (conjunto de salidas), determina como éstas se deben iterar (conjunto de separación), e indica la secuencia de cálculos (orden de precedencia) — todo esto para minimizar el número de variables a iterar. Debido a la falta de una teoría comprensible, la capacidad de resolución del algoritmo sólo puede ser juzgado por sus resultados. Por ello se presentan aquí seis nuevos ejemplos: equilibrio en reacción vapor-sólido, equilibrio en la combustión de gases complejos, optimización de reactor, flujo compresible e isentrópico, evaporación de triple efecto (según Ledet e Himmelblau 1970), y extracción de multicomponentes (según Ramirez y Vestal 1972). El más grande tiene 28 ecuaciones y 35 variables y se analiza fácilmente sin la ayuda de computador, lo que indica la utilidad del algoritmo.

#### Introducción

Tanto en ciencia aplicada como en ingeniería, se presentan grandes conjuntos de ecuaciones simultáneas, lineales y no lineales. Generalmente sólo unas pocas variables aparecen en cada ecuación y se dice que el conjunto es "disperso". En los últimos años se han presentado una sucesión

#### Abstract

Systematic procedures are needed to find the most efficient strategy for solving large sets of simultaneous equations, whether the equations arise in engineering design or scientific calculation. The algorithm of Stadtherr *et al.* (1974 *b*) assigns parameters (design variable set), selects the equation by which each dependent variable is to be calculated (output set); determines how these are to be iterated (tear set), and prescribes the sequence of computations (precedence order) — all to minimize the number of iterated variables. Because a comprehensive theory is lacking, the algorithm's power can be judged only by its results. Six new examples are presented: vapor-solid reaction equilibrium, complex gaseous combustion equilibrium, reactor optimization, isentropic compressible flow, triple-effect evaporation (after Ledet and Himmelblau 1970), and multicomponent extraction (after Ramirez and Vestal 1972). The largest has 28 equations and 35 variables and is easily analyzed without the aid of a computer.

#### Introduction

Large sets of simultaneous equations, linear and nonlinear, arise in both applied science and engineering. Usually only a few variables appear in each equation and the set is said to be "sparse". Recent years have seen a succession of increasingly effective algorithms for uncovering effi-

de algoritmos de creciente efectividad para la obtención de estrategias eficientes para resolver sistemas de ecuaciones dispersos. Los sistemas de ecuaciones de mayor interés han sido aquellos que aparecen en el diseño de redes eléctricas y de procesos químicos. A partir de experiencias con cálculos científicos, Stadtherr *et al.* (1973, 1974 *a*) han desarrollado un nuevo algoritmo que parece ser superior a algoritmos previamente disponibles para muchos tipos de cálculos. Debido a que todavía no se dispone de una teoría matemática comprensiva, el mérito de un algoritmo sólo puede ser juzgado por los resultados al aplicarlo en casos específicos. Stadtherr *et al.* (1973, 1974 *a*) han descrito tres ejemplos simples y dos más complejos. Stadtherr y Scriven (1974 *b*) describen dos ejemplos de equilibrio químico complejo en fase gaseosa. Aquí aplicamos el nuevo algoritmo a otros seis ejemplos de cálculos científicos y de ingeniería, algunos de los cuales implican sistemas grandes de ecuaciones. En los casos más simples el algoritmo nos lleva directamente a las estrategias de soluciones normales. En los ejemplos 4, 5 y 6 el algoritmo nos conduce a mejoras potenciales en estrategias de cálculo respecto de las publicadas anteriormente.

A los efectos de encontrar una estrategia de cálculo un sistema de ecuaciones puede representarse convenientemente por una matriz de ocurrencia, en la cual se pone una  $x$  en la posición  $ij$  si la variable  $j$  aparece en ecuación  $i$ , y en caso contrario, la posición se deja en blanco. "Particionar" un conjunto de ecuaciones es separarlo en conjuntos menores, los cuales pueden ser resueltos entonces secuencialmente en un cierto "orden de precedencia". Un conjunto descomponible en una serie de subconjuntos  $1 \times 1$  es "triangular"; uno que no puede ser particionado es "irreducible". La correspondencia uno a uno que asocia cada ecuación con la variable para la cual debe ser resuelta es el "vector variables de salida". En una matriz de ocurrencia la asignación de un vector de salida puede ser indicada encerrando las entradas en rectángulos (ver figuras). Cuando el número de variables  $M$  en un sistema de ecuaciones excede el

cient strategies for solving sparse sets of equations. The types of equation sets of greatest interest have been those that arise in electrical network design and chemical process design. From experience with scientific calculations, Stadtherr, Gifford and Scriven (1973, 1974 *a*) have developed a new algorithm which appears to be superior to previously available algorithms for many sorts of calculations. Because no comprehensive mathematical theory is available yet, the merits of an algorithm can be judged only by the results of applying it in specific cases. Stadtherr *et al.* (1973, 1974 *a*) described three simple examples and two more substantial ones. Stadtherr and Scriven (1974 *b*) describe two examples of complex chemical equilibria in the gas phase. Here we apply the new algorithm to six other examples of scientific and engineering calculations, some of which involve large systems of equations. In the simplest cases the algorithm takes us straight to the standard solution strategies. In Examples 4, 5 and 6 the algorithm leads us to potential improvements on solution strategies which have been published previously.

A set of equations can be represented conveniently by an occurrence matrix, in which an  $x$  is entered in position  $ij$  if variable  $j$  occurs in equation  $i$ , and if it does not, the position is left blank. To 'partition' a set of equations is to break it into smaller sets which can then be solved sequentially in a certain "precedence order". A set decomposable into a series of  $1 \times 1$  subsets is "triangular"; one that cannot be partitioned is "irreducible". The one-to-one correspondence that associates each equation with the variable for which it is to be solved is the "output set". In an occurrence matrix the assignment of an output set can be indicated by enclosing the appropriate entries in boxes (see Figures). When the number of variables  $M$  in an equation set exceeds the number of independent equations  $N$ ,  $M-N$  of the variables must be specified before calculations can proceed. These are parameters or "design

número de ecuaciones independientes  $N$ ,  $M-N$  de las variables se deben especificar antes de proceder con los cálculos. Estos son parámetros o "variables de diseño" y forman el "vector de variables de diseño" o de parámetros.

Los sistemas de ecuaciones irreducibles tienen que ser resueltos por métodos iterativos. Con sistemas de ecuaciones dispersos es posible a menudo llegar a una solución iterando sólo algunas variables. Las variables de iteración y las ecuaciones usadas para verificar las suposiciones en iteraciones sucesivas forman un "conjunto de iteración". Los algoritmos anteriores requerían que el conjunto de iteración fuera escogido de tal modo que el resto de las variables pudieran ser resueltas triangularmente, es decir una variable de cada ecuación sucesiva. Sin embargo existen casos en los cuales uno puede resolver directamente pequeños conjuntos de ecuaciones debido a que son lineales o pueden ser reducidas a cuadráticas u otras formas solucionables. Tal conjunto es llamado un "subconjunto permisible". El número de ecuaciones en un subconjunto permisible es su "rango". El nuevo algoritmo relaja el requerimiento de triangularidad: reconoce que sólo se necesita encontrar un conjunto de iteración tal que los cálculos pueden proseguir acíclicamente formando un bloque triangular. Esto es, resolviendo una sucesión de subconjuntos permisibles de ecuaciones.

El nuevo algoritmo asigna el vector de salida, los conjuntos de iteración, el conjunto de variables de diseño y el orden de precedencia, lo cual minimiza el tamaño del bloque triangular. Emplea el mismo concepto básico que Christensen (1970) llamó poner índices ("indexing"). Los rasgos sobresalientes de este nuevo algoritmo son las subrutinas para identificar y ordenar por orden de precedencia eficientemente cualquier subconjunto permisible de rango mayor que uno. Este algoritmo tiene la ventaja sobre muchos anteriores, de que sistemas de ecuaciones pequeños e incluso moderadamente grandes pueden ser fácilmente llevadas a cabo sin recurrir a una computadora. Por cierto, las estrategias a las cuales se llega en los ejemplos que siguen, se obtienen fácilmente a mano.

variables" and comprise the "design variable set" or parameter set.

Irreducible equation sets have to be solved by iterative methods. With sparse equation sets it is often possible to reach a solution by guessing, or "tearing", only a few of the variables. The variables torn and the equations used to check the guesses in successive iterations form the "tear set". Previous algorithms required that the tear set be so chosen that the remaining variables could be calculated triangularly, that is, one variable from each successive equation. However there are cases in which one can solve small sets of equations directly because they are linear or can be reduced to solvable quadratic or other forms. Such a set is called an "allowable subset". The number of equations in an allowable subset is its "rank". The new algorithm relaxes the triangularity requirement: it recognizes that one need only find a tear set such that the calculations can proceed acyclically in "block triangular form", that is, by the solution of a succession of allowable subsets, of equations.

The new algorithm assigns the output set, tear set, design variable set, and precedence order which minimize the size of the block triangular tear set. It employs the same basic concept which Christensen (1970) named "indexing". The unique features of this new algorithm are subroutines for efficiently identifying and precedence ordering any allowable subsets of rank greater than one. This algorithm has the advantage over many earlier ones, that for small and even moderately large systems of equations it can be implemented easily without recourse to a computer. Indeed, the strategies to which it leads in the examples below are all found readily by hand.

El nuevo algoritmo, que es descrito completamente por Stadtherr *et al.* (1973, 1974 a); se aplica ahora a seis ejemplos. Tres de ellos son cálculos científicos: equilibrio de reacción vapor-sólido, un equilibrio en la combustión de un gas complejo y un problema de un flujo compresible e isentrópico. Los otros tres son cálculos representativos de diseño en ingeniería: reactores agitados en serie, un evaporador de triple efecto y un sistema de extracción de multicomponentes.

### Ejemplo 1

Considere el problema de equilibrio de reacción vapor-sólido que Lever (1965) resolvió durante el estudio del crecimiento del vapor de cristales de germanio empastado en galio, un material semiconductor del tipo *p*. Una solución sólida diluida de galio en germanio se supone en equilibrio con una fase de vapor consistente en Ga, GaI<sub>3</sub>, Ge<sub>2</sub>, GeI<sub>4</sub>, I<sub>2</sub> e I. Lever formuló las ecuaciones que describen el estado de equilibrio y que se presentan en la forma de una matriz de ocurrencia en Fig. 1 (a). *P* es la presión total, *p<sub>i</sub>* son presiones parciales, *T* es la temperatura, y *x* es la fracción molar de galio en el sólido. Hay dos fases y tres componentes atómicos; de acuerdo a la regla de las fases existen tres grados de libertad. Por lo tanto, tres variables se deben especificar como parámetros.

La aplicación del nuevo algoritmo revela que especificando el conjunto de parámetros {*p<sub>GeI<sub>4</sub></sub>*, *T*, *p<sub>GaI<sub>3</sub></sub>*} permite que el sistema sea resuelto triangularmente como en Fig. 1 (b). El algoritmo también establece que hay otros conjuntos de parámetros, por ejemplo {*p<sub>GeI<sub>2</sub></sub>*, *p<sub>GeI<sub>4</sub></sub>*, *p<sub>GaI<sub>3</sub></sub>*}, que también lleva a la misma estrategia de solución. Estos conjuntos de parámetros son elecciones convenientes si se debe determinar la composición de la fase sólida en equilibrio con una fase de vapor dada. Por otro lado, cuando se buscan las presiones parciales en la fase gaseosa en equilibrio con un sólido de composición dada, es más conveniente elegir un conjunto de parámetros que contenga la fracción molar *x*. Si *x* se preasigna como un parámetro (suprimiendo la columna correspondiente a *x* antes de aplicar el algoritmo) el resultado es la estrategia de bloque

The new algorithm, which is described completely by Stadtherr *et al.* (1973, 1974 a), is now applied to six examples. Three are scientific calculations: a vapor-solid reaction equilibrium, a complex gaseous combustion equilibrium, and an isentropic compressible flow problem. The other three are representative engineering design calculations: a series of stirred-tank reactors, a triple-effect evaporator and a multicomponent extraction system.

### Example 1

Consider the vapor-solid reaction equilibrium problem which Lever (1965) solved in the course of studying the vapor growth of gallium-doped germanium crystals, a *p*-type semiconductor material. A dilute solid solution of gallium in germanium is presumed to be in equilibrium with a vapor phase consisting of Ga, GaI<sub>3</sub>, GeI<sub>2</sub>, GeI<sub>4</sub>, I<sub>2</sub> and I. Lever formulated the equations describing the equilibrium state and they are presented in the form of an occurrence matrix in Fig. 1 (a). *P* is total pressure, *p<sub>i</sub>*'s are partial pressures, *T* is temperature, and *x* is the mole fraction of gallium in the solid. There are two phases and three atomic components; according to the phase rule there are three degrees of freedom. Hence three variables must be specified as parameters.

Applying the new algorithm reveals that specifying the parameter set {*p<sub>GeI<sub>4</sub></sub>*, *T*, *p<sub>GaI<sub>3</sub></sub>*} allows the system to be solved triangularly as in Fig. 1 (b). The algorithm also establishes that there are other parameter sets, for example {*p<sub>GeI<sub>2</sub></sub>*, *p<sub>GeI<sub>4</sub></sub>*, *p<sub>GaI<sub>3</sub></sub>*}, which also lead to the same solution strategy. These parameter sets are convenient choices if the composition of the solid phase in equilibrium with a given vapor phase is to be determined. On the other hand, when the partial pressures in the gas phase in equilibrium with a solid of given composition are sought, it is more convenient to choose a parameter set containing the mole fraction *x*. If *x* is pre-designated as a parameter (by deleting the column corresponding to *x* before applying the algorithm) the result is the block-triangular strategy shown in Fig. 1 (c).

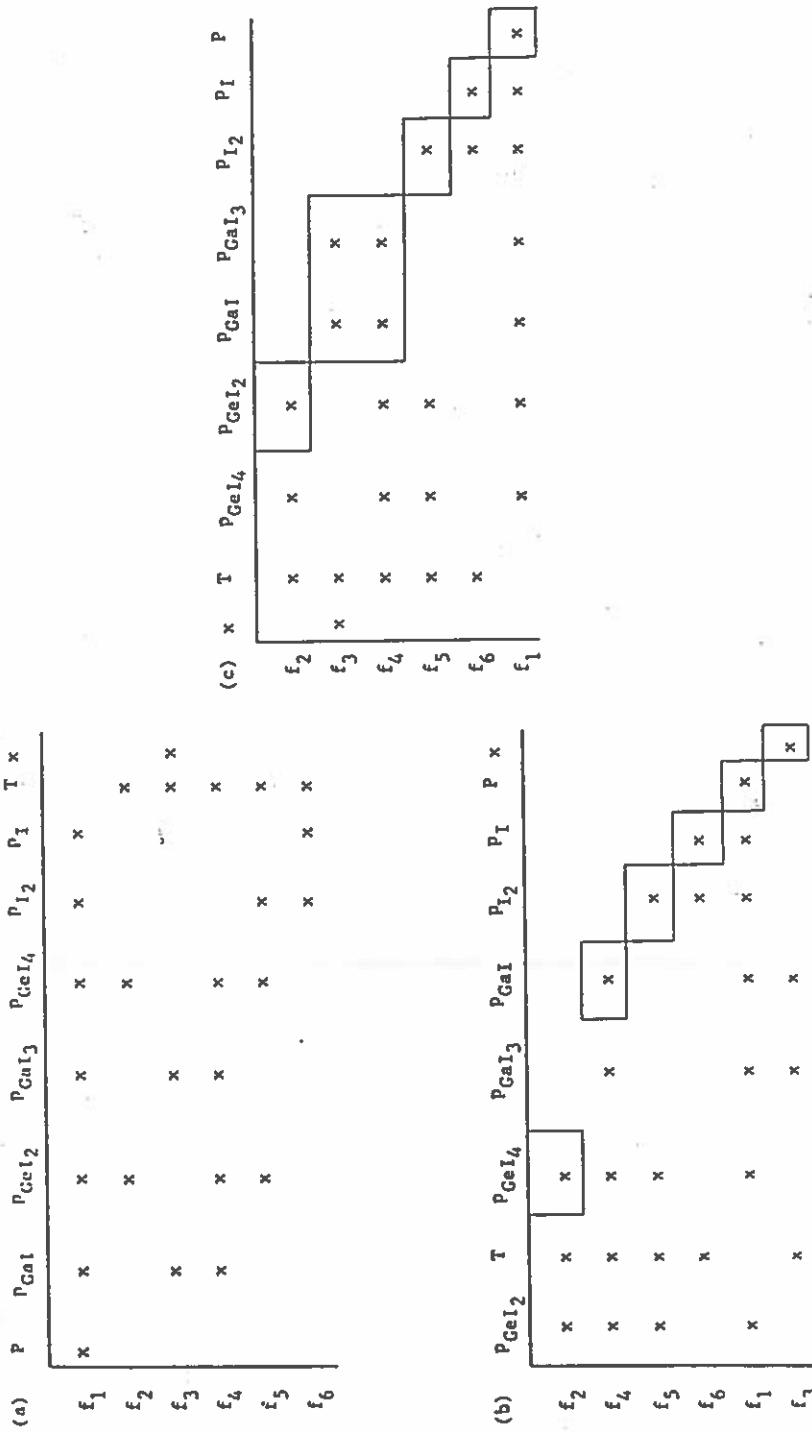


Fig. 1. -- (a) Matriz de ocurrencia para el equilibrio de la reacción vapor - sólido (Ge-Ga-I (ejemplo 1)). La variable x es la fracción molar de Ga en la fase sólida. (b) El resultado del algoritmo sin pre-designar parámetros. El conjunto de parámetros  $\{P_{GeI}, T, P_{GeI}, \}$  produce un conjunto de ecuaciones triangulares. (c) El resultado con x pre-designada como parámetro. El subconjunto de rango dos  $\{f_5, f_6; P_{GeI}, P_{GeI}, \}$  es permutable.

(a) Occurrence matrix for Ge-Ga-I vapor-solid reaction equilibrium (example 1). Variable x is the mole fraction of Ga in the solid phase. (b) The result of the algorithm with no pre-designation of parameters. The parameter set  $\{P_{GeI}, T, P_{GeI}, \}$  yields a triangular equation set. (c) The result with x pre-designated as a parameter. The rank-two subset  $\{f_5, f_6; P_{GeI}, P_{GeI}, \}$  is allowable.

triangular mostrada en Fig. 1 (c). El subconjunto de rango dos  $\{f_3, f_1; p_{0st}, p_{0at}\}$  puede ser resuelto directamente sin iteración y por lo tanto es permisible. El conjunto de parámetros indicados en Fig. 1 (c) es el conjunto que Lever decidió usar.

The rank-two subset  $\{f_3, f_1; p_{0st}, p_{0at}\}$  can be solved directly without iteration and therefore it is allowable. The parameter set indicated in Fig. 1 (c) is the set which Lever decided to use.

**Ejemplo 2**

**Example 2**

Se considera el análisis del flujo isentrópico, compresible, de un gas no ideal desde un punto estacionario, como lo formuló Shapiro (1953). La matriz de ocurrencia para el conjunto de ecuaciones gobernantes está dado en Fig. 2 (a). Si la entropía estacionaria,  $s_0$ , y entalpía estacionaria,  $h_0$  son preasignadas como parámetros, la aplicación del algoritmo resulta en Fig. 2 (b), la que indica que para hacer los cálculos triangulares, ya sea la presión  $p$  o la densidad  $\rho$  deben ser establecidas como el parámetro adicional. Cuál de ellos elegir puede depender de la forma de la ecuación de estado  $f_5(s, p, \rho) = C$ : si  $f_5$  puede ser resuelto explícitamente para  $p$ , pero no para  $\rho$ , entonces  $p$  es el parámetro más conveniente y *vice versa*. Vale la pena hacer notar que cuando la ecuación de estado se conoce sólo en la forma de datos experimentales, el algoritmo puede ser usado para decidir la manera más

Consider the analysis of isentropic, compressible flow of a non-ideal gas from a stagnation point, as formulated by Shapiro (1953). The occurrence matrix for the set of governing equations is given in Fig. 2 (a). If the stagnation entropy,  $s_0$ , and stagnation enthalpy,  $h_0$ , are predesignated as parameters, applying the algorithm results in Fig. 2 (b), which indicates that to make the calculations triangular either the pressure  $p$  or the density  $\rho$  should be assigned as the additional parameter. Which to choose may depend on the form of the equation of state  $f_5(s, p, \rho) = C$ : if  $f_5$  can be solved explicitly for  $p$  but not for  $\rho$ , then  $p$  is the more convenient parameter, and *vice versa*. It is well worth remarking that when the equation of state is known only in the form of experimental data, the algorithm can be used to decide the most convenient way to plot or tabulate the data. Here, for instance,  $f_4(s, \rho,$

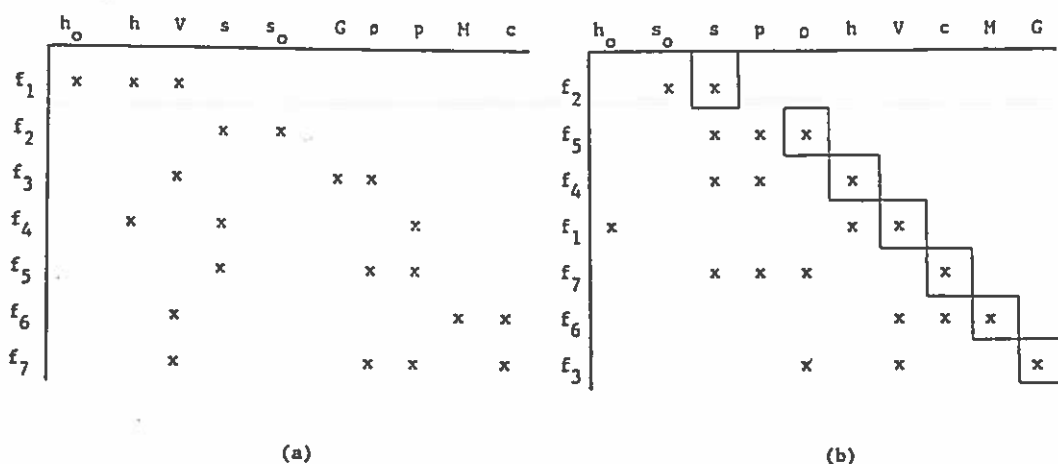


Fig. 2. — (a) Matriz de ocurrencia para flujo isentrópico y compresible (ejemplo 2). (b) El resultado del algoritmo con  $h_0$  y  $s_0$  preasignados como parámetros. Ya sea  $p$  (como se muestra) o  $\rho$  deberían elegirse como un parámetro adicional para hacer triangular el conjunto de ecuaciones.

(a) Occurrence matrix for isentropic, compressible flow (example 2). (b) The result of the algorithm with  $h_0$  and  $s_0$  predesignated as parameters. Either  $p$  (as shown) or  $\rho$  should be chosen as an additional parameter to make the equation set triangular.

conveniente de graficar o tabular los datos. Aquí, por ejemplo,  $f_i(s, \rho, h) = 0$  debe ser resuelto para  $h$ , y por lo tanto los datos deberían tabularse de tal forma que  $h$  pueda encontrarse fácilmente cuando se dan valores de  $s$  y  $\rho$ .

Este ejemplo es muy simple: la estrategia de solución es bastante estandar (Shapiro, 1953) y se encuentra fácilmente de nuevo sin la ayuda de un algoritmo. Sin embargo, el éxito del nuevo algoritmo en este simple problema indica su utilidad potencial para cálculos de flujo compresible más complejos, particularmente aquellos que implican gases no ideales, gases reactivos, llamas, deflagraciones y detonaciones.

### Ejemplo 3

Se considera el problema de optimización en la selección de tamaños y temperaturas de operación en reactores de agitación en serie con flujo continuo, de tal forma que la conversión deseada sea alcanzada en un tiempo total de residencia mínimo (Aris, 1969). Se supone que sólo existe una reacción exotérmica, y que la velocidad de reacción por unidad de volumen es una función conocida del avance de la reacción  $\xi$  y de la temperatura  $T$ . Sea  $R(\xi)$  la velocidad de reacción positiva mayor para un  $\xi$  dado con  $T$  restringido por límites superior e inferior. Como Aris lo demuestra, el problema de optimización consiste en la solución de  $N + 1$  variables con  $N - 1$  ecuaciones derivadas de la forma

$$0 = f_i(\xi_i, \xi_{i+1}, \xi_{i+2}) = \frac{1}{R(\xi_{i+1})} - (\xi_{i+1} - \xi_{i+2}) \frac{R'(\xi_{i+1})}{[R(\xi_{i+1})]^2} - \frac{1}{R(\xi_i)}$$

Ahora la ecuación  $f_i$  es fácilmente solucionable para  $\xi_{i+2}$ , pero  $R(\xi)$  es, en general, una función complicada, de manera que los subconjuntos de  $1 \times 1$ ,  $\{f_i; \xi\}$  y  $\{f_i; \xi_{i+1}\}$  no son permisibles. Así,  $\{f_i; \xi_{i+2}\}$  son los únicos subconjuntos de  $1 \times 1$  permisibles, y en el caso de cinco reactores agitados en serie la matriz de ocurrencia es como se muestra en Fig. 3 (a); los subconjuntos no permisibles están indicados por entradas en círculos.

Las líneas verticales quebradas indican que las conversiones de entrada y salida están especificadas como variables de di-

$h) = 0$  must be solved for  $h$ , and hence the data should be so tabulated that  $h$  can be easily found when values of  $s$  and  $\rho$  are given.

This example is a very simple one: the solution strategy is quite standard (Shapiro 1953) and is easily discovered anew with no assistance from an algorithm. Nevertheless, the success of the new algorithm on this simple problem points to its potential utility for more complex compressible flow calculations, particularly those involving non-ideal gases, reactive gases, flames, deflagrations and detonations.

### Example 3

Consider the optimization problem of selecting sizes and operating temperatures of continuous-flow, stirred-tank reactors in series so that the desired conversion is achieved in the minimum total holding time (Aris 1969). Suppose there is but a single exothermic reaction, and the reaction rate per unit volume is a known function of extent-of-reaction  $\xi$  and temperature  $T$ . Let  $R(\xi)$  be the greatest positive reaction rate for given  $\xi$  with  $T$  constrained between upper and lower bounds. Then, as Aris shows, the optimization problem consists of solving for  $N + 1$  variable with  $N - 1$  derivative equations of the form

Now equation  $f_i$  is easily solved for  $\xi_{i+2}$  but  $R(\xi)$  is generally a complicated function and so the  $1 \times 1$  subsets  $\{f_i; \xi\}$  and  $\{f_i; \xi_{i+1}\}$  are disallowed. Thus  $\{f_i; \xi_{i+2}\}$  are the only allowable  $1 \times 1$  subsets, and in the case of five stirred tanks in series the occurrence matrix is as shown in Figure 3 (a); the disallowed subsets are indicated by circled entries. The broken vertical lines indicate that the input and output conversions are specified as design variables or parameters in the optimization problem under consideration. It

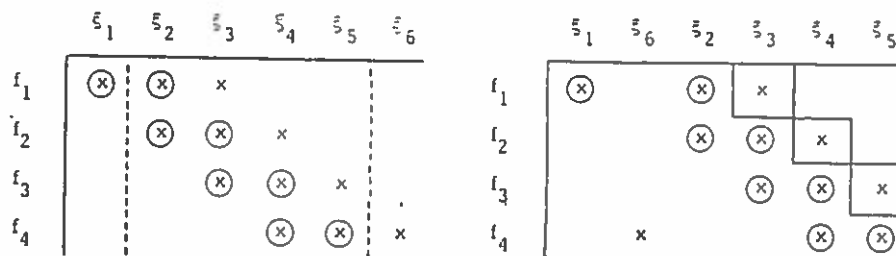


Fig. 3. -- (a) Matriz de ocurrencia para la optimización de 5 reactores de agitación en serie con flujo continuo (ejemplo 3). Las conversiones  $\xi_1$  y  $\xi_5$  son variables de diseño preasignadas. Las entradas encerradas en círculos son subconjuntos no permisibles. (b) La estrategia de mono-separación encontrada con el uso del nuevo algoritmo: separar  $\xi_2$  y verificar con ecuación  $f_4$ .

(a) Occurrence matrix for optimization of five continuous flow, stirred-tank reactors in series (example 3). Conversions  $\xi_1$  and  $\xi_5$  are pre-designated design variables. Circled entries are disallowed subsets. (b) The one-tear strategy found using the new algorithm: tear  $\xi_2$  and check with equation  $f_4$ .

seño o parámetros en el problema de optimización bajo consideración. Basta sólo inspeccionar para ver que la mejor estrategia es separar  $\xi_2$  y verificar con  $f_4$ . Esto es confirmado por el nuevo algoritmo, que conduce a poner índice en la Columna  $\xi_5$  (aunque la Columna  $\xi_2$  también tiene sólo dos entradas, no es una candidata porque no contiene una salida permisible) como se muestra en Fig. 3 (b). Aris demostró cómo implementar esta estrategia por medio de una construcción gráfica. Como en el último ejemplo, el algoritmo obtiene una estrategia bien conocida, pero ello lo hace en una forma sistemática.

#### Ejemplo 4

Se considera el equilibrio en la combustión de boro-hidruro de aluminio, combustible de alta energía. Cálculos del equilibrio en la fase gaseosa fueron hechos por Sachsel *et al.* (1949). Como constituyentes del sistema gaseoso ideal se considera: Al, AlO,  $Al_2O_3$ , B,  $B_2$ , BO,  $B_2O_3$ ,  $B_4O_6$ ,  $H_2O$ ,  $O_2$ ,  $H_2$ , H, O y OH. Se elige como componentes  $H_2O$ ,  $H_2$ ,  $Al_2O_3$  y  $B_2O_3$ . Escrito en términos de números molares,  $n_i$ , el conjunto de ecuaciones gobernantes está dado por la matriz de ocurrencia en Fig. 4 (a). Las entradas numéricas representan exponentes en las ecuaciones de acción de masas y las  $x$  representan términos lineales en las ecuaciones de balance atómico. También se requiere la ecuación para el número total de moles,  $n = \sum_i n_i$  pero no se incluye en la matriz de ocurrencia, por-

is already plain by inspection that the best strategy is to tear  $\xi_2$  and check with  $f_4$ , and this is confirmed by the new algorithm, which leads to indexing on Column  $\xi_5$  (Column  $\xi_2$ , though it too has only two entries, is not a candidate because it contains no allowable output) and the result shown in Fig. 3 (b); Aris demonstrated how to implement this strategy by graphical construction. As in the last example, the algorithm merely reproduces a well-known strategy, but it does so in a reassuringly systematic way.

#### Example 4

Consider the combustion equilibrium of the high-energy fuel, aluminum borohydride. Gas-phase equilibrium calculations were made by Sachsel *et al.* (1949). The constituents of the ideal gas system are taken to be Al, AlO,  $Al_2O_3$ , B,  $B_2$ , BO,  $B_2O_3$ ,  $B_4O_6$ ,  $H_2O$ ,  $O_2$ ,  $H_2$ , H, O, and OH. As components  $H_2O$ ,  $H_2$ ,  $Al_2O_3$ , and  $B_2O_3$  are chosen. Written in terms of mole numbers,  $n_i$ , the set of governing equations is given by the occurrence matrix in Fig. 4 (a). Numerical entries represent exponents in the mass-action equations and  $x$ 's represent linear terms in the atom-balance equations. The equation for the total number of moles,  $n = \sum_i n_i$ , is also required but is not included in the occurrence matrix because the total mole number  $n$  is auto-



que el número total de moles  $n$  es automáticamente elegido como una variable de separación (Stadtherr y Scriven, 1973). Si todos los subconjuntos que se pueden resolver directamente se consideran permisibles, el nuevo algoritmo produce una estrategia de mono-separación (bi-separación cuando se cuenta  $n$ ) en Fig. 4 (b). El subconjunto de rango dos  $\{f_{10}, f_{11}; n_{II,0}, n_{0II}\}$  contiene ecuaciones lineales solamente (nótese las entradas numéricas en la submatriz) y puede ser resuelto en forma simple. Sin embargo, los dos subconjuntos de rango tres (submatrices de  $3 \times 3$  en Fig. 4 (b)) no se pueden resolver tan fácilmente. El subconjunto  $\{f_1, f_2, f_{14}; n_{\Delta 10}, n_{\Delta 1}, n_{\Delta 10}\}$  puede ser reducido a una ecuación cuadrática, pero sólo a expensas de la introducción, en el proceso de elevación al cuadrado, de una raíz positiva extraña que de alguna manera debe ser eliminada durante cada iteración sucesiva. El subconjunto  $\{f_5, f_6, f_{12}; n_{B,0}, n_{B0}, n_{B,0}\}$  puede ser reducido a una ecuación cuadrática (nótese las entradas  $-1/2$  y  $-2$  en la submatriz) otra vez a expensas de la introducción de una raíz positiva extraña. De forma que, aunque el esquema de solución en Fig. 4 (b) usa el número mínimo de separaciones, puede que no sea la mejor estrategia, debido a las dificultades en eliminar las raíces extrañas y la resolución de una ecuación cuadrática durante cada iteración (la solución práctica de la ecuación cuadrática requiere separación, pero esto se hace eficientemente con subrutinas normales para encontrar raíces de polinomios). Sachs et al. (1949) usaron una estrategia equivalente a aquella dada en Fig. 4 (b).

Si sólo se admiten como permisibles los subconjuntos lineales y cuadráticos de resolución fácil definidos por Stadtherr y Scriven (1974 b), el nuevo algoritmo da origen a la estrategia de bi-separación (de tri-separación cuando se cuenta  $n$ ) lo que se muestra en Fig. 4 (c). El subconjunto  $\{f_{10}, f_{11}; n_{II,0}, n_{0II}\}$  es lineal; el subconjunto  $\{f_2, f_{12}; n_{\Delta 1,0}, n_{\Delta 10}\}$  es cuadrático y tiene sólo una raíz positiva. Aunque esta estrategia usa una variable de separación adicional, representa un adelanto potencial comparada con la estrategia de Fig. 4 (b) porque implica sólo subconjuntos de fácil resolución que no introducen raíces extrañas.

máticamente chosen as a tear variable (Stadtherr and Scriven 1973). If all directly solvable subsets are regarded as allowable, the new algorithm yields the one-tear (two-tear when  $n$  is counted) strategy in Fig. 4 (b). The rank-two subset  $\{f_{10}, f_{11}; n_{II,0}, n_{0II}\}$  contains linear equations only (note the numerical entries in the submatrix) and can be solved simply. However, the two rank-three subsets ( $3 \times 3$  submatrices in Fig. 4 (b)) are not as easily solved. The subset  $\{f_1, f_2, f_{14}; n_{\Delta 10}, n_{\Delta 1}, n_{\Delta 10}\}$  can be reduced to a quadratic equation, but only at the expense of introducing, in the process of squaring, one extraneous positive root which somehow must be eliminated during each successive iteration. The subset  $\{f_5, f_6, f_{12}; n_{B,0}, n_{B0}, n_{B,0}\}$  can be reduced to a quartic equation (note the entries  $-1/2$  and  $-2$  in the submatrix) again at the expense of introducing one extraneous positive root. So even though the scheme of solution in Fig. 4 (b) uses the minimum number of tears, it may not be the best strategy, because of the difficulties of eliminating the extraneous roots and of solving a quartic equation during each iteration (practical solution of the quartic requires tearing, but this is done efficiently by standard subroutines for finding roots of polynomials). Sachs et al. (1949) employed a strategy equivalent to that given by Fig. 4 (b).

If only the easily solved linear and quadratic subsets defined by Stadtherr and Scriven (1974 b) are admitted as being allowable, the new algorithm yields the two-tear (three-tear when  $n$  is counted) strategy shown in Fig. 4 (c). The subset  $\{f_{10}, f_{11}; n_{II,0}, n_{0II}\}$  is linear; the subset  $\{f_2, f_{12}; n_{\Delta 1,0}, n_{\Delta 10}\}$  is quadratic and has only one positive root. Although this strategy uses an additional tear variable, it is a potential improvement on the strategy of Fig. 4 (b) because it involves only easily solvable subsets which introduce no extraneous roots.

(a)

	$n_{H_2O}$	$n_{H_2}$	$n_{Al_2O_3}$	$n_{B_2O_3}$	$n_{Al}$	$n_{Al_2O_3}$	$n_{AlO}$	$n_B$	$n_{B_2O_3}$	$n_{B_4O_6}$	$n_H$	$n_{O_2}$	$n_O$	$n_{OH}$
$f_1$	3/2	-3/2	-1/2		1									
$f_2$	1/2	-1/2	-1/2			1								
$f_3$	3/2	-3/2	-1/2				1							
$f_4$	3	-3	-1					1						
$f_5$	1/2	-1/2	-1/2						1					
$f_6$										1				
$f_7$		-1/2									1			
$f_8$	-2	2										1		
$f_9$	-1	1											1	
$f_{10}$	-1	1/2												1
$f_{11}$	x	x	x	x	x	x	x	x	x	x	x	x	x	x
$f_{12}$	x	x	x	x	x	x	x	x	x	x	x	x	x	x
$f_{13}$														
$f_{14}$			x											

(b)

	$n_{H_2}$	$n_H$	$n_{H_2O}$	$n_{OH}$	$n_{Al_2O_3}$	$n_{AlO}$	$n_{Al}$	$n_{Al_2O_3}$	$n_{B_2O_3}$	$n_{B_4O_6}$	$n_B$	$n_{O_2}$	$n_O$	$n_{OH}$
$f_7$	-1/2	1												
$f_{10}$	1/2		-1	1										
$f_{11}$	x	x	x	x										
$f_1$	-3/2		3/2		-1/2	1								
$f_2$	-1/2		1/2		-1/2		1							
$f_{14}$					x	x	x							
$f_8$	2							1						
$f_9$	1								1					
$f_5$	-1/2		1/2							-1/2	1			
$f_6$					x	x	x	x	x	-2		1		
$f_{12}$	-3/2		3/2							-1/2			1	
$f_3$	-3		3							-1				1
$f_4$														
$f_{13}$														

(a) Occurrence matrix for the Al (BH)<sub>3</sub> - O<sub>2</sub> combustion system (example 4). (b) The result of the algorithm if all directly solvable subjets are allowable: test n<sub>H</sub>, and check with f<sub>7</sub>. (c) The result of the algorithm if only linear and quadratic subjets are allowable: test n<sub>H</sub>, n<sub>H<sub>2</sub>O</sub>, and n<sub>H<sub>2</sub></sub>, and check with equations f<sub>13</sub> and f<sub>14</sub>.

(c)

	$n_{H_2}$	$n_H$	$n_{H_2O}$	$n_{OH}$	$n_{B_2O_3}$	$n_{B_4O_6}$	$n_{O_2}$	$n_O$	$n_{Al_2O_3}$	$n_{AlO}$	$n_{Al}$	$n_{B_2}$	$n_B$
$f_7$	-1/2	1											
$f_{10}$	1/2		-1	1									
$f_{11}$	x	x	x	x									
$f_5$	-1/2		1/2		-1/2		1						
$f_6$								1					
$f_8$	2								1				
$f_9$	1									1			
$f_2$	-1/2		1/2							-1/2	1		
$f_{12}$	-3/2		3/2									1	
$f_1$	-3/2		3/2										1
$f_{14}$									x	x	x		
$f_4$	-3		3										1
$f_3$	-3/2		3/2										
$f_{13}$													

Fig. 4.— (a) Matrix de ocurrencia para el sistema de combustión Al (BH)<sub>3</sub> - O<sub>2</sub> (ejemplo 4). (b) El resultado del algoritmo si todos los subconjuntos de resolución directa son permisibles: separar n<sub>H</sub>, y verificar con f<sub>7</sub>. (c) El resultado del algoritmo si solo subconjuntos lineales y cuadráticos son permisibles: separar n<sub>H</sub>, y n<sub>H<sub>2</sub>O</sub>, y verificar con ecuaciones f<sub>13</sub> y f<sub>14</sub>.

*Ejemplo 5*

Se considera el ejemplo usado por Ledet e Himmelblau (1970) de un evaporador de triple efecto y con alimentación en paralelo. Los balances de masa y energía, las propiedades físicas y las ecuaciones de velocidad están dados por la matriz de ocurrencia en Fig. 5 (a). Por razones discutidas por Ledet e Himmelblau, es deseable impedir la resolución de ciertas ecuaciones para ciertas variables. Estos subconjuntos no permisibles de  $1 \times 1$  se indican por entradas encerradas en círculos en la matriz de ocurrencia. Usando su algoritmo para ordenar sistemas de ecuaciones, Ledet e Himmelblau idearon la estrategia que se muestra en Fig. 5 (b); ésta requiere cuatro separaciones.

La aplicación del nuevo algoritmo muestra que el número de separaciones puede ser reducido a tres. La estrategia de tri-separación a la que lleva el nuevo algoritmo se muestra en Fig. 5 (c). Por cuanto ningún subconjunto de rango mayor que uno aparece en el esquema, el algoritmo de Christensen (1970) también daría una estrategia de tri-separación. Sin embargo, el algoritmo de Christensen no sólo pierde subconjuntos permisibles cuando ellos existen, como tampoco contiene ninguna disposición para salidas no permisibles. De esta manera puede poner una salida no permisible en el conjunto de salida.

*Ejemplo 6*

Se considera el problema de diseño del extractor multicomponente y de dos fases presentado por Ramírez y Vestal (1972) como ejemplo. Usando su algoritmo<sup>\*</sup>, Ramírez y Vestal encontraron la estrategia que aparece en Fig. 6 (a) como una matriz de ocurrencia ya ordenada. Aunque ellos dicen que esta estrategia requiere seis variables de iteración, en verdad requiere sólo dos, porque todos los subconjuntos, los dos de  $2 \times 2$  y los dos de  $4 \times 4$ , contienen sólo ecuaciones lineales y pueden ser fácilmente resueltas sin iteración. Así, se puede decir que mientras hay seis "sepa-

\* Los autores no avalaron el algoritmo de Ramírez y Vestal, que fue cuestionado por un árbitro, pero esto no le resta significado al Ejemplo 6.

*Example 5*

Consider the forward-feed, triple-effect evaporator example used by Ledet and Himmelblau (1970). The mass and energy balances, physical property and rate equations are given by the occurrence matrix in Fig. 5 (a). For reasons discussed by Ledet and Himmelblau, it is desirable to prohibit the solution of certain equations for certain variables. These disallowed  $1 \times 1$  subsets are indicated by circled entries in the occurrence matrix. Using their algorithm for ordering equation systems, Ledet and Himmelblau (1970) devised the strategy shown in Fig. 5 (b); it requires four tears.

Applying the new algorithm shows that the number of tears can be reduced to three. The three-tear strategy to which the algorithm leads is shown in Fig. 5 (c). Inasmuch as no subsets of rank greater than one appear in the scheme, the algorithm of Christensen (1970) would also yield a three-tear strategy. However, Christensen's algorithm not only misses allowable subsets when they exist, but also contains no provision for disallowed outputs, and thus it could put a disallowed output in the output set.

*Example 6*

Consider the two-phase, multicomponent extractor design problem which Ramírez and Vestal (1972) presented as an example. Using their algorithm<sup>\*</sup>, Ramírez and Vestal found the strategy which appears in Fig. 6 (a) as an already ordered occurrence matrix. Though they state that this strategy requires six iterative variables, it in fact requires only two, because the two  $2 \times 2$  subsets and the two  $4 \times 4$  subsets all contain linear equations only and can easily be solved without iteration. Thus, one can say that while there are six "trian-

\* The authors did not validate the Ramírez-Vestal algorithm and a referee called it into question, but this does not detract from the significance of example 6.

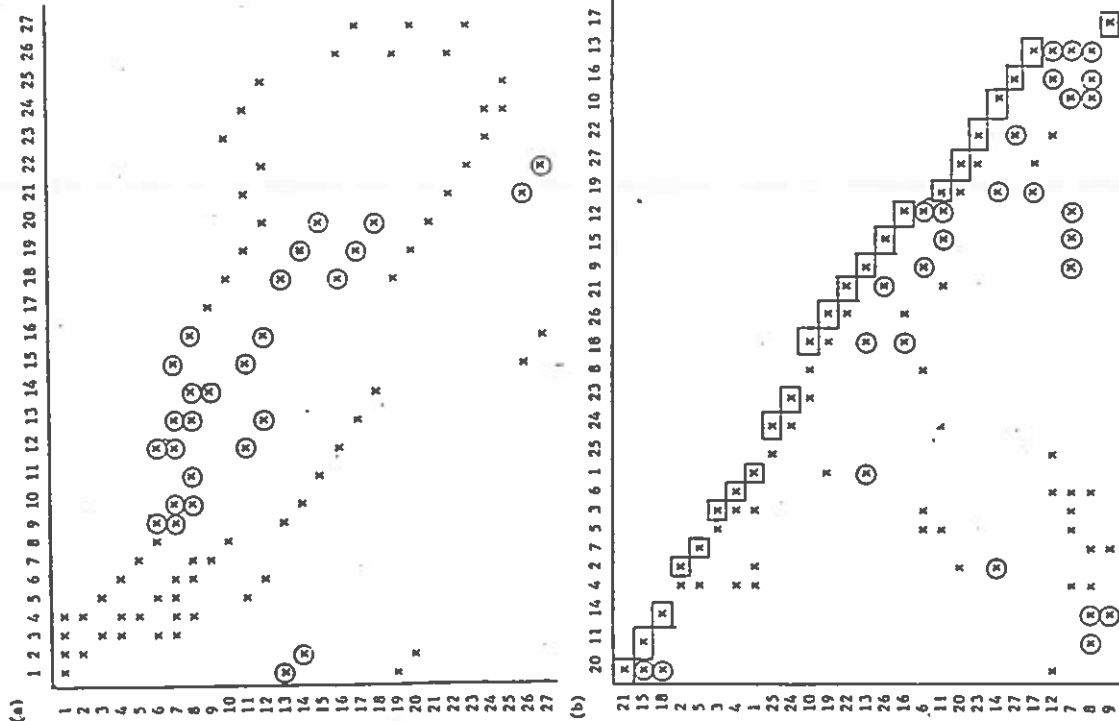


Fig. 5. -- (a) Matriz de ocurrencia para evaporador de triple efecto y con alimentación en paralelo (ejemplo 5). Los círculos corresponden a salidas no permisibles. Las ecuaciones y variables están numeradas de acuerdo a Ledet e Hinnelblad (1970). (b) La estrategia de extra-separación que resulta del algoritmo de Ledet: separar variables 4, 5, 25 y 8; verificar con ecuaciones 6, 12, 7 y 8. (c) Un resultado mejor encontrado usando el nuevo algoritmo. Hay sólo tres separaciones: variables 5, 4, y 18; estas se verifican con ecuaciones 25, 7 y 8.

(a) Occurrence matrix for forward-feed triple-effect evaporator (Example 5). Circled entries are disallowed outputs. Equations and variables are numbered as by Ledet and Hinnelblad (1970). (b) The four-tear strategy which results from Ledet's algorithm: tear variables 4, 5, 25 and 8; check with equations 6, 12, 7, 8. (c) An improved strategy found using the new algorithm. There are only three tears: variables 5, 4, and 18; these are checked with equations 25, 7, and 8.

raciones triangulares", hay sólo dos "separaciones triangulares de bloque".

El número de separaciones triangulares de bloque puede ser reducido de dos a uno usando el nuevo algoritmo. En Fig. 6 (b) se muestra la estrategia mejorada. El conjunto variable de diseño es

$$\{ X_{1,2}^H, z_{3,2}, z_{5,2}, z_{3,4}, X_{1,3}^I, X_{2,3}^I, \tau \}$$

El problema puede ser resuelto separando  $F_1$ , resolviendo tres subconjuntos permisibles de  $2 \times 2$  y dos subconjuntos permisibles de  $4 \times 4$ , todos lineales y verificando con el residuo en la ecuación 23. Los subconjuntos permisibles incluyen aquellos en Fig. 6 (a); el subconjunto adicional de  $2 \times 2$  es  $\{f_{22}, f_{24}; F_1, F_2\}$  y representa un precio bajo a pagar por reducir una doble iteración entrelazada, a una sola iteración.

**Conclusión**

El algoritmo desarrollado por Stadtherr *et al.* (1974 b) lleva a esquemas eficientes en la resolución de problemas de multi ecuaciones en ciencia aplicada y en diseño en ingeniería, y a menudo da origen a estrategias de solución nuevas y mejores, como se demostró en los ejemplos anteriores. El más extenso de éstos tiene 28 ecuaciones y 35 variables y fue fácilmente analizado sin la ayuda de una computadora. Los rasgos sobresalientes de este nuevo algoritmo son las subrutinas, que localizan y ordenan por precedencia, pequeños subconjuntos de ecuaciones que pueden ser resueltos sin iteración. Un defecto importante es que en su forma actual no toma totalmente en cuenta las velocidades relativas de convergencia de estrategias distintas. Aún así, su éxito en el descubrimiento de mejoras potenciales sobre estrategias ideadas con algoritmos anteriores, pone en claro que son posibles aún más avances. Una posibilidad por ejemplo, es el mejor procedimiento de índice que admitiría subconjuntos permisibles (de rango mayor que uno) en secuencias de índice.

El objetivo final es por supuesto un procedimiento completamente sistemático que tomara totalmente en cuenta tanto las propiedades estructurales como las propiedades numéricas de un sistema de ecuaciones y que diera origen a aquella estrategia de solución que requiera el mínimo de gastos totales en computación.

ular tears", there are only two "block-triangular tears".

The number of block-triangular tears can be reduced from two to one by using the new algorithm. The improved strategy is shown in Fig. 6 (b). The design variable set is

The problem can be solved by tearing  $F_1$ , solving three  $2 \times 2$  and two  $4 \times 4$  allowable subsets, all linear, and checking with the residual in equation 23. The allowable subsets include those in Fig. 6 (a); the additional  $2 \times 2$  subset is  $\{f_{22}, f_{24}; F_1, F_2\}$  and is a trivial price to pay for reducing a nested double iteration to a single iteration.

**Conclusion**

The algorithm developed by Stadtherr *et al.* (1974 b), leads to efficient schemes for solving multi-equation problems in applied science and engineering design, and often results in new and improved solution strategies, as demonstrated in the preceding examples. The largest of these has 28 equations and 35 variables and was easily analyzed without the aid of an automatic computer. The key features of this new algorithm are subroutines which locate and precedence order small subsets of equations that can be solved without iteration. A significant shortcoming is that in its present form it does not take fully into account the relative convergence rates of alternative strategies. Nevertheless its success in uncovering potential improvements on strategies devised with earlier algorithms makes clear that further advances are possible. One possibility, for example, is an improved indexing procedure which would admit allowable subsets (of rank greater than one) into indexing sequences.

The ultimate goal is of course a completely systematic procedure which fully accounts for both the structural properties and the numerical properties of an equation system and leads to that solution strategy which requires the minimum total computational expense.

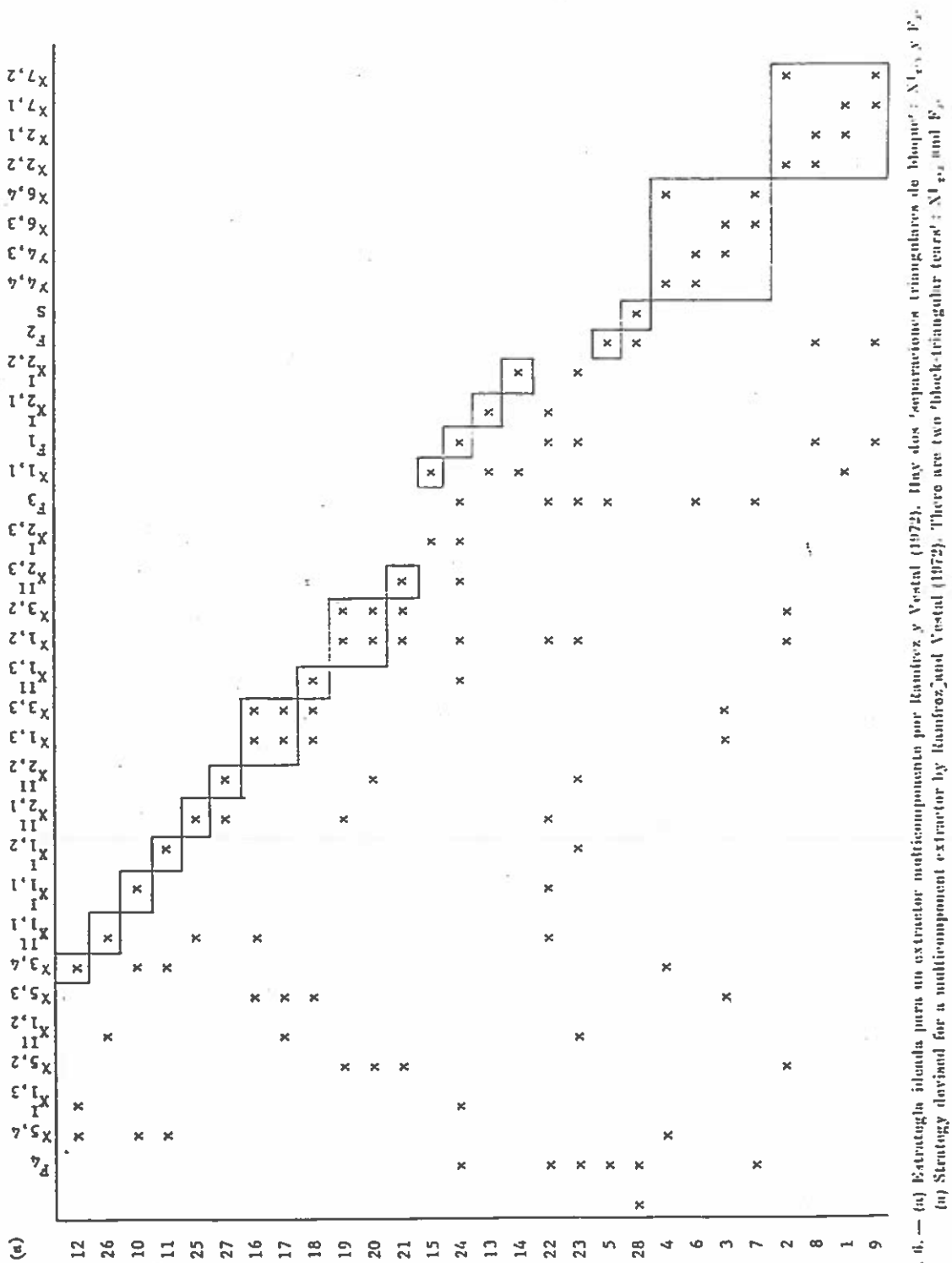


Fig. 6. — (a) Estrategia ilanda para un extractor multicomponente por Kanfroz y Vestal (1972). Hay dos separaciones triangulares de bloques:  $N_{1,2}$  y  $F_2$ . (b) Strategy devised for a multicomponent extractor by Kanfroz and Vestal (1972). There are two block-triangular terms:  $N_{1,2}$  and  $F_2$ .

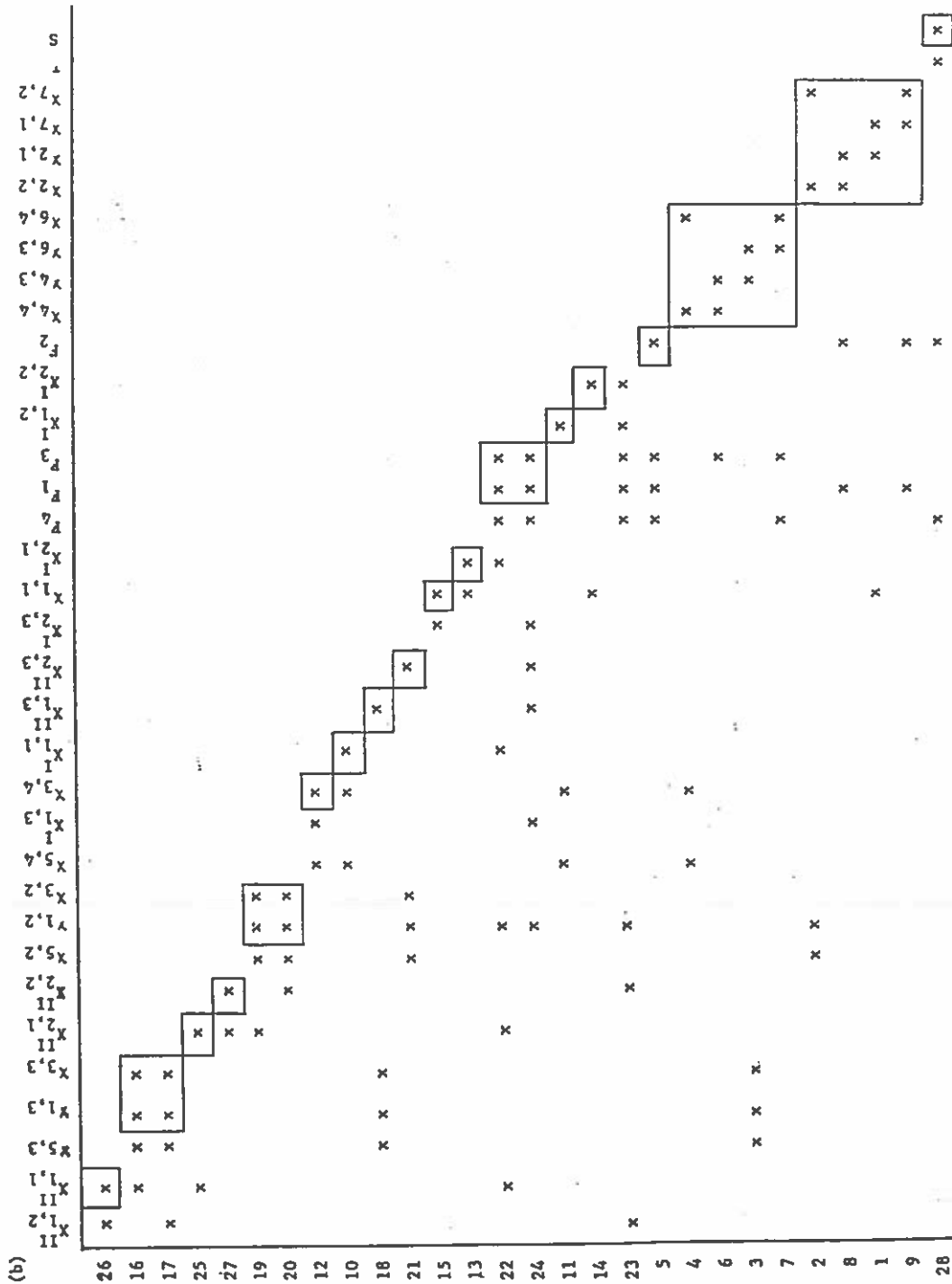


FIG. 6.— (a) Mejor estrategia encontrada con el uso del nuevo algoritmo. Hay sólo una variable de separación, F, ésta se verifica con la ecuación 23 (Ejemplo 6).  
 (b) Improved strategy found using the new algorithm. There is only one tear variable, F; this is checked with equation 23. (Example 6).

## A P E N D I C E \*

## SUBROUTINA 1. Ordenamiento de precedencia aciclico

La subrutina 1 se utiliza para ordenar un sistema de ecuaciones en un orden de precedencia de tipo bloque triangular de subconjuntos permisibles que pueden ser resuelto aciclicamente, sin iteraciones, siempre que ello sea posible. Esencialmente consiste en una generalización del algoritmo de Lee et al (AIChE J. 12, 6, 1966).

1.1 Establecer la matriz de ocurrencia para el sistema de ecuaciones a ser resuelto. Prepararse para escribir la matriz de ocurrencia reordenada la cual indicará, una vez completada, la estrategia de solución óptima.

1.2 Definir enales subconjuntos son permisibles en el sistema.

1.3 Encontrar una fila en la matriz de ocurrencia que tenga una sola entrada. Esta entrada representa un subconjunto de  $1 \times 1$  factible de remover. Si el subconjunto es permisible, se lo elimina tachando fila y columna en las que estaba la entrada y se las coloca en la *primer* fila y columna libres, de la matriz reordenada.

Indicar que esta fila (ecuación) debe ser resuelta para tal columna (variable) señalando la entrada dentro de un círculo.

Repetir la etapa 1.3 hasta que no puedan efectuarse más entradas en esta nueva matriz.

Nótese que si aparece una fila sin entradas, el sistema está sobreespecificado y debe ser reanalizado.

1.4 Encontrar una columna con sólo una entrada. Esta entrada representa un subconjunto de  $1 \times 1$  factible de remover. Si dicha entrada es permisible se tacha la fila y columna correspondientes y se ubica en la *última* fila y columna no ocupadas de la matriz reordenada.

\* This appendix is presented only in spanish due to the fact of being a partial translation of the paper of Stadhierr et al (1974a).

Indicar que esta entrada representa una asignación de conjunto de salida recuadrándola. Si hay alguna columna que ahora no tiene entradas, se la tacha y se la ubica en la última columna no ocupada en la matriz reordenada. Se repite el paso 1.4 hasta que no se puedan realizar más entradas en la matriz reordenada.

1.5 Encontrar todas las filas no tachadas donde aparezcan solamente dos entradas. Si dos de esas filas tienen sus entradas en las mismas dos columnas, se ha encontrado un subconjunto de  $2 \times 2$  factible de remover. Si el subconjunto es permisible, se lo tacha y se lo ubica en las dos *primeras* filas y columnas no ocupadas de la matriz reordenada y se lo recuadra.

Repetir los pasos 1.3 y 1.5 hasta que no puedan realizarse más entradas en la matriz reordenada.

1.6 Encontrar todas las columnas no tachadas donde aparezcan solamente dos entradas. Si dos de esas columnas tienen sus entradas en las mismas dos filas, se ha encontrado un subconjunto de  $2 \times 2$  factible de remover. Si el subconjunto es permisible, se lo tacha y se lo coloca en las dos *últimas* filas y columnas no ocupadas de la matriz reordenada y se las recuadra.

Si aparecen ahora columnas sin entradas, se les tacha y coloca en la última columna libre de la matriz reordenada.

Repetir pasos 1.4 y 1.6 hasta que no sea posible realizar nuevas entradas en la matriz reordenada

1.7 Encontrar todas las filas no tachadas en las que aparezcan tres o menos entradas. Si tres de dichas filas tienen sus entradas en las mismas tres columnas, un subconjunto de  $3 \times 3$  factible de remover ha sido encontrado. Si es permisible, se lo tacha y se lo ubica en las primeras tres filas y columnas libres de la nueva matriz, se repiten las etapas 1.3, 1.5 y 1.7 hasta que no se pueda proceder a más pasajes a la matriz reordenada.



1.8 Encontrar todas las columnas no tachadas en las que haya tres o menos entradas. Si tres de dichas columnas tienen sus entradas en las mismas tres filas, se ha encontrado un subconjunto de  $3 \times 3$  factible de remover. Si el mismo es permisible, se lo tacha y se lo ubica en las tres últimas filas y columnas libres de la nueva matriz.

Si ahora se encuentra alguna columna sin entrada alguna se la tacha y se la agrega en la última columna libre de la nueva matriz.

Se repiten las etapas 1.4, 1.6 y 1.8 hasta que no se pueda proceder a nuevos pasajes a la matriz reordenada.

1.9 Repetir las etapas 1.7 y 1.8 reemplazando tres por cuatro. Luego reemplazar cuatro por cinco, etc., hasta que el rango del subconjunto permisible más grande sea alcanzado.

Los subconjuntos removibles en las etapas 1.3 y 1.9 deben ser examinados para decidir si son permisibles, o sea que pueden ser resueltos por técnicas analíticas. La matriz de ocurrencia convencional no suministra información suficiente para ello. Cuando se descomponen pequeños sistemas manualmente, es posible determinar fácilmente la permisibilidad.

Para un ordenamiento por computadora de grandes sistemas, es necesario modificar la matriz de ocurrencia de manera que pueda ser utilizada directamente para determinar la permisibilidad de un subconjunto (ello es descrito en otro trabajo de los autores).

La búsqueda de subconjuntos removibles de rango mayor que uno en etapas 1.5 y 1.9 no es importante en sistemas pequeños, pero para grandes sistemas, examinar todas las combinaciones de filas o columnas elegibles puede insumir un tiempo prohibitivo. Afortunadamente la subrutina 2 posee un medio eficiente de encontrar pequeños subconjuntos factibles de remover.

#### Subrutina 2. Localización de subconjuntos removibles

La subrutina 2 soluciona los problemas combinatorios de encontrar subconjuntos removi-

bles de rango mayor que uno. Se basa en la idea que un subconjunto factible de removerse no puede contener una fila o columna con una sola entrada, ya que esto implicaría que existe un subconjunto removible de rango menor y debería haberse procedido previamente a su remoción.

La descripción de la subrutina 2 corresponde a su uso con las etapas 1.6 y 1.8 de la subrutina 1; para usarla con las etapas 1.5 y 1.7 reemplazar la palabra columna por fila y viceversa. Para encontrar subconjuntos de  $2 \times 2$  removibles en la etapa 1.6 de la subrutina 1:

2.1 Formar la submatriz constituida por todas las columnas con solamente dos entradas y las filas donde aparecen estas entradas.

2.2 Encontrar una fila con una sola entrada. Tacharla así como la columna donde aparece la entrada. Tachar toda fila sin entradas. Repetir esta etapa hasta que no queden filas con menos de dos entradas.

2.3 Encontrar la fila con el mayor número de entradas. Tacharla así como todas las columnas sobre las que existan entradas. Buscar una fila en la que se haya reducido en dos el número de entradas. Tacharla. Las dos filas tachadas en esta etapa, junto con las dos columnas conteniendo las entradas comunes, forman un subconjunto de  $2 \times 2$  factible de ser removido, al que debe verificarse su permisibilidad.

Si hay más de una fila con dos entradas menos o filas que hayan perdido tres o más entradas, tachar todas esas filas, pues esto significa que se han encontrado más de un subconjunto de  $2 \times 2$ , de los cuales solo uno puede ser removido. Todos los subconjuntos encontrados deben ser examinados en cuanto su permisibilidad a fin de determinar cual, si existe alguno, es el mejor para ser removido.

Tachar toda columna vacía.

Repetir las etapas 2.2 y 2.3 hasta que la submatriz sea vacía, lo cual significa que se han encontrado todos los subconjuntos de  $2 \times 2$  removibles.

Para analizar los subconjuntos de  $3 \times 3$  removibles en la etapa 1.8 de la subrutina 1:

2.4 Formar la submatriz constituida por todas las columnas con tres o menos entradas y

las filas donde aparecen estas entradas. Aplicar la etapa 2.2.

- 2.5 Encontrar la fila con el número más grande de entradas y tacharla. Tachar temporariamente las columnas en las cuales tenga entradas dicha fila. Si hay *dos* filas cuyo número de entradas ha sido reducida en dos o más y si al tacharlas quedasen *tres* columnas vacías, efectuar dicho tachado. Las *tres* filas tachadas y las *tres* columnas vacías, forman un subconjunto de  $3 \times 3$  removible. Nótese que otra vez debe tenerse en cuenta la localización simultánea de más de un subconjunto de los cuales solo uno es realmente removible. Todas las columnas temporariamente tachadas que contengan menos de dos entradas pueden ser tachadas en forma definitiva, el resto deberá ser restituido. Aplicar las etapas 2.2 y 2.5 hasta que la submatriz sea vacía y se hayan ubicado todos los subconjuntos de  $3 \times 3$ .

- 2.6 Para extender lo expuesto a subconjuntos de rango mayor incrementar sucesivamente en uno las cantidades subrayadas.

Este procedimiento es un medio efectivo de encontrar subconjuntos de  $2 \times 2$  y  $3 \times 3$  removibles, aunque se vuelve menos eficaz cuando se lo utiliza para ubicar subconjuntos más grandes. Pero los subconjuntos permisibles son usualmente pequeños y raramente será deseable buscar subconjuntos removibles grandes.

### Un algoritmo para la estrategia de solución

El algoritmo emplea las subrutinas 1 y 2 para determinar una estrategia de solución que minimice el número de variables supuestas. Las secuencias de índices no son examinadas hasta que todos los posibles indicados simples hayan sido agotados. (Para descomposición de grandes sistemas por computadora puede ser adecuado el procedimiento de indicio de Christensen conjuntamente con las subrutinas, aunque los resultados obtenidos por el nuevo algoritmo son a menudo superiores a los alcanzados por el algoritmo de Christensen). Para asignar una estrategia de solución:

1. Aplicar la subrutina 1. Si se alcanza a llenar la matriz reordenada, el conjunto de variables supuestas es vacío. Aquellas columnas no asignadas como salidas representan variables de diseño o parámetros.

2. Si falla la etapa 1, encontrar la columna con el mínimo número de entradas  $C_{MIN}$ . Indiciar sobre una de tales columnas tachándola e igualmente las filas donde aparecen entradas, ubicándolas en las últimas filas y columnas no ocupadas en la matriz reordenada y asignando la columna índice como la salida de la más alta fila permisible recuadrando tal entrada: ésta debe representar un subconjunto permisible. Esto introduce  $t = C_{MIN} - 1$  variables supuestas.

3. Aplicar las etapas 1.4, 1.6, 1.8 y 1.9 de la subrutina 1, usando la subrutina 2 como sea necesario. Si esto logra completar la matriz reordenada, asignar como variables a suponer las últimas  $t$  columnas aún no asignadas como salida. Agregar al conjunto de supuestas las filas para las cuales no hay salida. Designar como variables de diseño las columnas restantes que no sean salida.

4. Si la etapa 3 falla, rehacer las etapas 2 y 3, iniciando sobre otra columna con  $C_{MIN}$  entradas.

Nótese que cualquier fila y columna tachadas precisamente en las etapas 2 y 3 deben ser reubicadas antes de proceder al nuevo indicio. Si ha sido indiciado sobre cada una de las columnas que poseen  $C_{MIN}$  entradas y no se completa la matriz reordenada, es necesario indiciar sobre una secuencia de columnas.

Para indiciar una secuencia de columnas:

5. Encontrar una secuencia límite. Esto se logra haciendo primero un indiciado sobre la columna con  $C_{MIN}$  entradas que produce el grado de descomposición más grande en la etapa 4, luego aplicando 2 y 3 a la submatriz remanente y repitiendo el proceso hasta que se complete la matriz reordenada.

El número total de variables supuestas  $t_{max}$  requerido por esta secuencia de columnas índices es una cota superior sobre el tamaño óptimo del conjunto de variables a suponer.

6. Buscar otra secuencia para la cual  $t < t_{max}$ . Nótese que a causa de un indiciado sobre una columna con  $c$  entradas introduce  $t = c - 1$

variables a suponer, todas las columnas con más de  $t_{\max}$  entradas pueden ser excluidas en la consideración de la primera etapa de indicación. Similarmente, en cualquier etapa se excluirán aquellas que hagan  $t \geq t_{\max}$ , con lo cual el número de secuencias posible se reduce sustancialmente. Continuar hasta que no quede ninguna secuencia para la cual  $t$  puede ser menor que  $t_{\max}$  (una secuencia índice puede ser solo una columna).

### Referencias

- Aris, R., « Elementary Chemical Reactor Analysis », Section 7.8, Prentice-Hall, Englewood Cliffs, N. J. (1969) and translation, « Análisis de Reactores », Editorial Alhambra S. A., Madrid, Buenos Aires, Mexico, Rio de Janeiro (1973).
- Christensen, J. H., « The Structuring of Process Optimization », *A. I. Ch. E. J.* **16**, 177 (1970)
- Ledet, W. P., Himmelblau, D. M., « Decomposition Procedures for the Solving of Large Scale Systems », *Adv. Chem. Eng.* **8**, 185 (1970).
- Lever, R. F., « Vapor transport and Thermodynamic Equilibria in the Germanium - Gallium - Iodine System », *J. Phys. Chem. Solids* **26**, 1629 (1965).
- Ramirez, W. F., Vestal, C. R., « Algorithms for Structuring Design Calculations », *Chem. Eng. Sci.*, **27**, 2243 (1972).
- Sachsel, G. F., Mantis, M. E., Bell, J. C., « A Note on the Calculation of Multicomponent Propellant Gas Compositions », pp. 620-623 in « Third Symposium on Combustion and Flame and Explosion Phenomena », Williams and Wilkins, Baltimore, Md. (1949).
- Shapiro, A. H., « The Dynamics and Thermodynamics of Compressible Fluid Flow », Volume 1, Ronald Press, New York (1953).
- Stadtherr, M. A., Gifford, W. A., Scriven, L. E., « Efficient Solution of Sparse Sets of Design Equations » (summary), Preprints of the Fifth Interamerican Congress of Chemical Engineering, Rio de Janeiro, July 22-28 (1973).
- Stadtherr, M. A., Gifford, W. A., Scriven, L. E., « Efficient Solution of Sparse Sets of Design Equations », *Chem. Eng. Sci.*, **29**, 1025 (1974a).
- Stadtherr, M. A., Scriven, L. E., « On Efficient Calculation of Complex Chemical Equilibria », *Chem. Eng. Sci.*, in press (1974b).

### References

Recibido : Agosto 1973

Aceptado : Noviembre 1973

Received : August 1973

Accepted : November 1973