

SOLUCION EFICIENTE EN MICROCOMPUTADORES DE SISTEMAS GRANDES DE ECUACIONES LINEALES QUE SE PRESENTAN EN EL ANALISIS DE ESTRUCTURAS

José Jaime García A. Profesor Asociado.
Gilberto Gómez R. Profesor Asistente.

Departamento de Mecánica de Sólidos y Materiales
Universidad del Valle

RESUMEN

En muchos tipos de análisis estáticos y dinámicos de estructuras es necesario resolver sistemas grandes de ecuaciones lineales. La solución se facilita gracias a que la matriz de coeficientes es simétrica y de banda, lo que reduce tanto el número de posiciones en memoria principal necesario para almacenar sus componentes, como el número de operaciones para obtener la solución.

En este artículo se hace una revisión de los algoritmos para resolver eficientemente en microcomputadores

sistemas grandes de ecuaciones lineales. Se describe primero el algoritmo de banda y se explican luego más detalladamente los de perfil y el método frontal, que son los más usados en programas comerciales. Se discuten posteriormente otros procedimientos que requieren una mínima parte de la memoria principal, y se presentan finalmente los resultados obtenidos usando el método frontal. A este respecto se han desarrollado subprogramas en la Universidad del Valle, los cuales están a disposición de docentes e investigadores. 

INTRODUCCION

Para analizar problemas de estructuras usando el método de los elementos finitos es necesario resolver sistemas grandes de ecuaciones lineales, las cuales tienen una matriz de coeficientes que es simétrica y de banda; aunque con ello se reduce en gran medida el número de posiciones de memoria necesario para alojar sus componentes y el número de

operaciones para obtener la solución, la cantidad de grados de libertad puede ser tan considerable, que es frecuentemente imposible mantener toda la matriz en la memoria principal de microcomputadores. Se hace entonces necesario utilizar algoritmos que permitan manejar el sistema usando adicionalmente memoria de disco.

El propósito de este artículo es dar a conocer los diversos procedimientos que existen para solucionar ecuaciones lineales en microcomputadores, y poner a disposición de los interesados las rutinas que se han implementado en la Sección de Estructuras de la Universidad del Valle. Se espera que con la información que aquí se presenta, no siempre de fácil consecución en nuestro medio, se puedan entender y usar mejor los programas comerciales, y que estudiantes e investigadores enriquezcan sus bases para desarrollar aplicaciones propias.

Se revisan primero los procedimientos para solucionar eficientemente los sistemas de ecuaciones lineales que resultan en análisis de estructuras. Se explican luego en detalle los métodos de perfil y frontal, que son los más usados en aplicaciones comerciales. Por último se presentan algunos resultados de tiempos de ejecución.

REVISION DE METODOS

En los diferentes tipos de análisis de sistemas estructurales es necesario solucionar sistemas grandes

elementos, según los grados de libertad a los cuales estén asociados. Se han desarrollado varios algoritmos que buscan aprovechar las características de simetría y de banda de la matriz.

ALGORITMO DE BANDA

La primera alternativa propone usar el algoritmo de banda¹, que consiste en almacenar la parte superior de la matriz en un arreglo rectangular (Fig. 1), en el cual los elementos de la diagonal de la matriz se ubican en la primera columna del arreglo. Este tiene un número de filas igual al número de ecuaciones del sistema, **NE**, y un número de columnas igual al ancho de banda, **AB**.

El valor de **AB** depende de la configuración de la malla y de la forma en que se numeran los nodos: **AB** es igual a **n+1**, donde **n** es la máxima diferencia entre los grados de libertad de todos los nodos de un mismo elemento, considerados todos los elementos. En consecuencia, la numeración de los nodos de una malla debe ir en la dirección del lado más corto, tal como se ilustra en la Fig. 2.b; el valor de **AB** para cada caso mostrado corresponde al supuesto de que se tiene un grado de libertad por nodo.

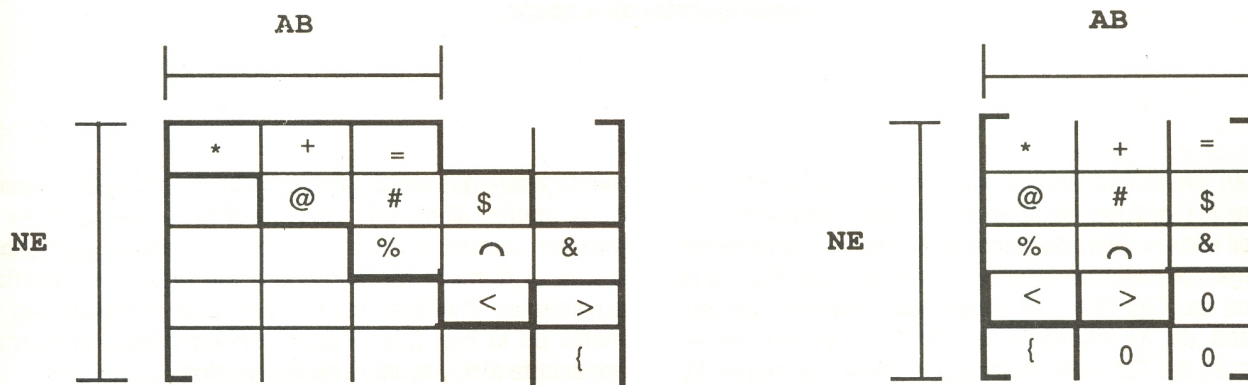


Fig. 1. Almacenamiento en arreglo rectangular

de ecuaciones lineales; se citan como ilustración los casos de análisis estático lineal, análisis dinámico con algoritmos de integración paso a paso en el tiempo, análisis no lineales y análisis de frecuencias de vibración o cargas críticas de pandeo (problema de valores y vectores característicos).

En todos estos casos, la matriz de coeficientes del sistema (que aquí se seguirá llamado simplemente matriz), se ensambla con las contribuciones de los

Dos desventajas tiene el algoritmo de banda. En primer lugar, en la parte inferior derecha del arreglo rectangular se presenta un triángulo de ceros (Fig. 1), el cual, en caso de mallas grandes, puede tener un número apreciable de componentes; por ejemplo, si el ancho de banda es de 100, el número de ceros es de 4950. Por otro lado, como el ancho del arreglo queda definido por **AB**, es decir, por la diferencia máxima entre los grados de libertad de un mismo elemento, considerados todos ellos, en caso de mallas irregulares, en las

25	26	27	28	29	30
17	18	19	20	21	22
9	10	11	12	13	23
1	2	3	4	14	24
			5	15	16
			6	7	8

a. Numeración inadecuada, AB = 12

4	8	12	18	24	30
3	7	11	17	23	29
2	6	10	16	22	28
1	5	9	15	21	27
			14	20	26
			13	19	25

b. Numeración óptima, AB = 8

Fig. 2. Ancho de banda según la numeración de los nodos, elementos rectangulares de 4 nodos.

cuales se presentan variaciones grandes en la diferencia entre los grados de libertad en cada elemento, se tiene por ello un elevado número de ceros a la derecha del arreglo; por ejemplo, según la numeración dada a la malla de la Fig. 2.b., la diferencia máxima en los elementos de la izquierda es de 5, y en los de la derecha es de 7, lo cual implica que en las primeras 12 filas del arreglo habrá 2 columnas de ceros en su parte derecha. En mallas más grandes se presentan cantidades de ceros considerables.

ALGORITMO DE PERFIL O BANDA VARIABLE

Con el método de perfil²⁻⁵, también llamado de banda variable, se eliminan las mencionadas desventajas del algoritmo de banda.

Dada la conectividad de la malla (números de nodo que corresponden a cada elemento), se determina primero el perfil de la matriz, es decir, la zona en la cual hay contribuciones de los elementos (Fig. 3.). Numérica-

mente esto equivale a calcular la altura de cada columna de la matriz, altura que es igual a $m+1$, donde m es la máxima diferencia entre el grado de libertad correspondiente al número de columna y los grados de libertad adyacentes menores que aquel; por ejemplo, en la malla de la Fig. 2.b., 3 es el menor grado de libertad adyacente al 8, y la altura de la columna es, por tanto,

$$(8 - 3) + 1 = 6$$

como puede verse en la Fig. 3., en donde se presenta la forma de la primera parte de la matriz correspondiente a la malla de la Fig. 2.b..

Calculado ya el perfil de la matriz, es decir, el arreglo que contiene las alturas de cada columna, $ID(I)$, se ensambla la matriz en un vector unidimensional, tal como se muestra en la Fig. 3.. Si se suman secuencialmente las componentes de $ID(I)$, se obtiene un arreglo que identifica la posición en el vector de cada diagonal de la matriz, con lo cual puede manejársela en un almacenamiento unidimensional.

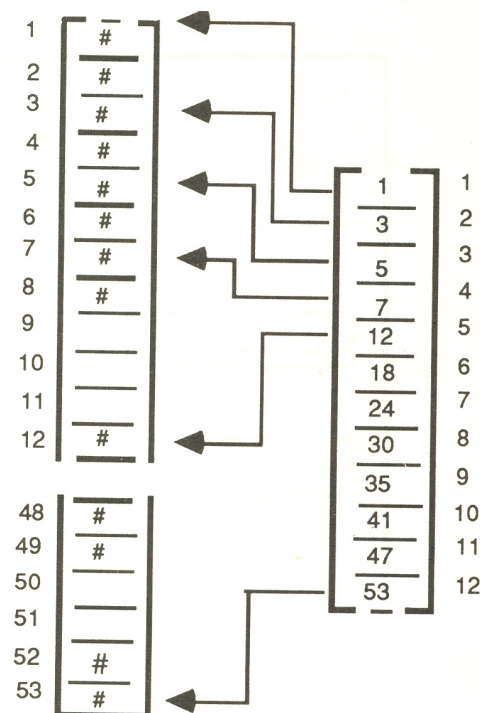
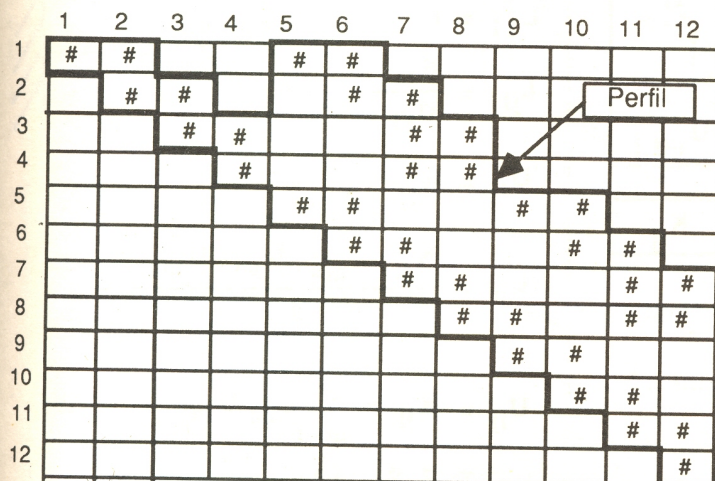


Fig. 3. Almacenamiento de banda variable o perfil

SOLUCION DE LAS ECUACIONES

Eliminación de Gauss

El sistema de ecuaciones

$$[A] \underline{X} = \underline{P} \quad (1)$$

se soluciona utilizando el método de eliminación de Gauss, que consiste en transformar mediante operaciones sencillas de filas la matriz $[A]$ en una matriz triangular superior $[U]$, la cual tiene un triángulo inferior izquierdo de ceros, es decir

$$[T][A] = [U] \quad (2)$$

El sistema (1) se soluciona entonces aplicando primero la matriz de transformación $[T]$ a cada vector de cargas,

$$[T] \underline{P} = \underline{y} \quad (3)$$

y calculando luego mediante substitución inversa el vector $\underline{X}$

$$[U] \underline{X} = \underline{y} \quad (4)$$

Cuando se efectúa un análisis estático, el método de Gauss equivale físicamente a soltar cada grado de libertad que se elimina. Mediante este método se pueden generar también las denominadas matrices condensadas, las cuales permiten hacer el análisis estático de estructuras grandes dividiéndolas en varias componentes (subestructuras).

Como el método de Gauss opera las filas de la matriz $[A]$ desde arriba hasta abajo, es apropiado para solucionar sistemas si se tiene el almacenamiento de banda.

Algoritmo de Crout

El algoritmo de Crout permite efectuar el procedimiento de eliminación de Gauss moviéndose, no a lo largo de las filas como tradicionalmente se hace, sino en dirección de las columnas, lo cual es especialmente apropiado para utilizarse en combinación con el algoritmo de perfil.

El procedimiento consiste en factorizar la matriz en una triangular inferior, $[L]$, y otra triangular superior, $[U]$. Los elementos de $[L]$ y $[U]$ se calculan fácilmente con base en los elementos de la matriz en la forma mostrada en la Fig. 4..

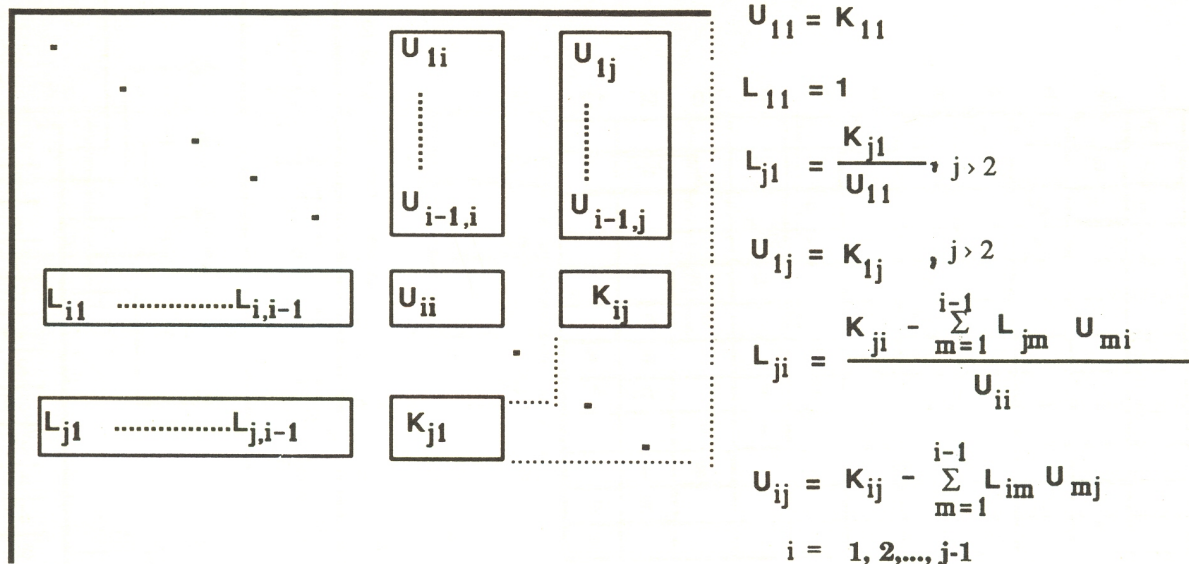


Fig. 4. Algoritmo de Crout

OTROS METODOS PARA EL ALMACENAMIENTO Y SOLUCION DEL SISTEMA

El método de perfil o banda variable combinado con el algoritmo de Crout hace un uso eficiente de la memoria. Sin embargo, en problemas bi y tridimensionales de estructuras es difícil manejar toda la matriz de coeficientes en la memoria principal de un microcomputador; si se tiene por ejemplo una malla ortogonal de vigas de 20 x 20 nodos con 6 grados de libertad por nodo, el número total de ecuaciones es de aproximadamente 2400 (un poco menos, si se tienen en cuenta los puntos fijos), y el número de entradas en la matriz de coeficientes es de

$$2400 \times 100 = 240000$$

suponiendo un ancho de banda medio de 100; sólo para procesar esta matriz en un microcomputador se requerirán

$$240000 \times 4 = 960000 \text{ bytes}$$

suponiendo que se trabaja con reales de precisión sencilla en FORTRAN. Estas cifras, que se dan para un problema bidimensional de tamaño mediano, sobrepasan los límites de la mayoría de los microcomputadores.

Es necesario entonces pensar en otros algoritmos que permitan manejar la matriz haciendo uso adicional de la memoria secundaria o de disco.

METODO FRONTAL

El método frontal es probablemente la alternativa de solución más ampliamente usada en programas comerciales. Según el profesor Irons⁶, la idea surgió en una oficina de ingeniería de la compañía Boeing en Seattle en 1958 y fue redescubierta después por varios investigadores. El primer artículo que lo explica detalladamente se publicó en el año 1970⁷. Desde entonces han sido propuestas diversas variaciones^{8,9} que intentan optimizar su funcionamiento.

Según el procedimiento de banda variable operfil, para solucionar el sistema de ecuaciones es necesario formar antes toda la matriz global de coeficientes. Según el procedimiento frontal, la matriz de coeficientes se va resolviendo a medida que se va formando, es decir, que cuando se completan las contribuciones en cierto grado de libertad, éste se elimina mediante el procedimiento de Gauss. En el ejemplo de la Fig. 5. las contribuciones en el grado de libertad 1 se completan después de que se adiciona la matriz de rigidez del elemento (3); en ese momento se elimina este grado de libertad.

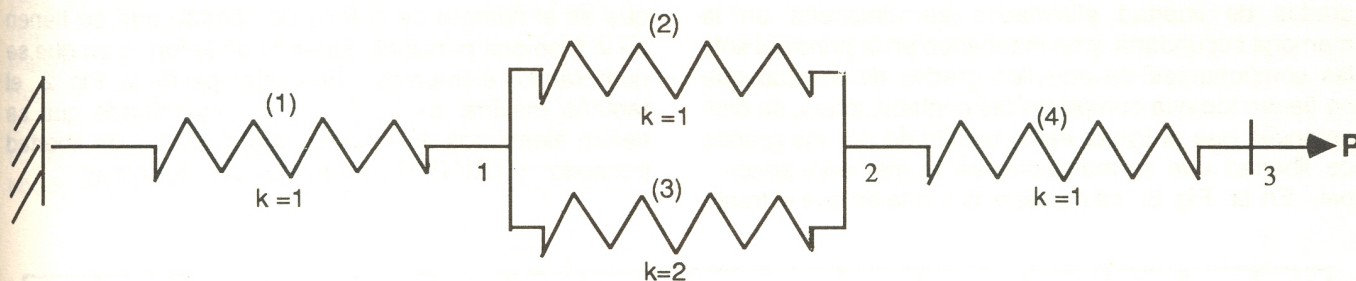


Fig. 5. Sistema de resortes

A continuación se describe detalladamente el procedimiento a la luz de un ejemplo:

Después de la adición de las matrices de los elementos (1), (2) y (3), y la eliminación del grado de libertad 1, se tiene

$$\begin{array}{c} 1 \quad 2 \quad 3 \\ \begin{matrix} 1 \\ 2 \\ 3 \end{matrix} \begin{bmatrix} 4 & -3 & 0 \\ -3 & 3 & 0 \\ 0 & 0 & 0 \end{bmatrix} \end{array} \xrightarrow[1]{\text{se elimina}} \begin{bmatrix} 4 & -3 & 0 \\ 0 & 3/4 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

se adiciona luego la matriz del elemento (4) y se elimina el grado de libertad 2, con lo cual se obtiene sucesivamente

$$\begin{array}{c} 1 \quad 2 \quad 3 \\ \begin{matrix} 1 \\ 2 \\ 3 \end{matrix} \begin{bmatrix} 4 & -3 & 0 \\ 0 & 7/4 & -1 \\ 0 & -1 & 1 \end{bmatrix} \end{array} \xrightarrow[2]{\text{se elimina}} \begin{bmatrix} 4 & -3 & 0 \\ 0 & 7/4 & -1 \\ 0 & 0 & 3/7 \end{bmatrix}$$

y se llega así a la matriz triangular superior deseada.

Según el procedimiento anterior la información de los grados de libertad eliminados se almacena en la memoria secundaria, y se mantienen en la principal sólo las componentes de aquellos grados de libertad que no tienen todavía completas las contribuciones; se dice entonces que el frente está constituido por los grados de libertad que se mantienen en la memoria principal. En la Fig. 6. se muestra la forma en que el frente

se va moviendo sobre la malla. El tamaño de frente, que es el número de grados de libertad que se tienen en la memoria principal, depende de la forma en que se numeran los elementos. En el ejemplo de la Fig. 2. el tamaño máximo de frente es 12, suponiendo que se tienen elementos de 4 nodos con 2 grados de libertad por nodo. (LEMFRO es igual al tamaño del frente).

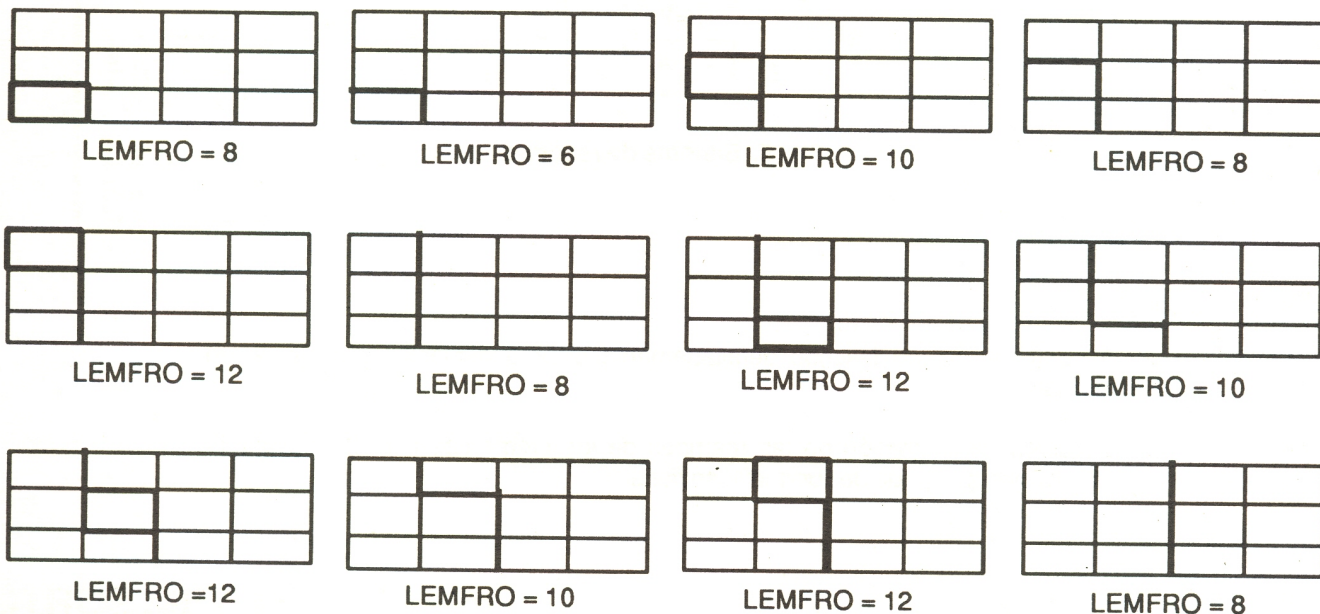


Fig. 6. Avance del frente en malla de elementos

Implementación

Al implementar el método frontal es necesario considerar los siguientes aspectos:

1. Se necesita un módulo preprocesador para definir el tamaño del frente. Este módulo debe marcar además la última aparición de cada nodo, lo cual se hace generalmente en el arreglo con la conectividad de los elementos, dándole un valor negativo al número de nodo cuando haga la última aparición.
2. Se crea un arreglo en el cual se almacenan las ecuaciones que se van eliminando. Cuando se copia este arreglo se escribe en el disco. Esta alternativa elimina la necesidad de escribir en el disco cada ecuación que se elimina, mejorando por tanto los tiempos de ejecución.
3. La instrucción central de la rutina es un ciclo en los elementos. Para cada elemento se determinan primero las direcciones que tienen sus grados de libertad en el frente (arreglo LDEST (I)); luego se

adicionan la rigidez y las cargas elementales en los vectores que recorren el frente (SUSTIF(I) y SUBLOD(I)); por último se procede a eliminar aquellos grados de libertad que hayan completado sus contribuciones.

Otras Alternativas

El método frontal no se puede usar cuando el frente es tan grande que la submatriz de rigidez y los subvectores de carga no caben en memoria principal.

Un algoritmo que permite superar la dificultad consiste en usar el método de perfil¹⁰, particionando la matriz de rigidez en bloques, tal como se ilustra en la Fig. 8.. El proceso de solución se efectúa cargando sucesivamente cada uno de los bloques en la memoria principal. Este método tiene la desventaja de que es necesario recalcular o almacenar en el disco la matriz de rigidez de aquellos elementos que pertenecen a dos bloques diferentes. Por tanto, en los casos en que es posible su ejecución, el método frontal es siempre una mejor alternativa.

Otra propuesta¹¹ consiste en usar el método frontal, pero ensamblando el sistema de ecuaciones nodo por nodo, y no elemento por elemento, tal como se hace tradicionalmente. Se empieza por el primer nodo de la malla y se ensamblan en él todas las contribuciones de los elementos que lo contienen; al final de este proceso se pueden eliminar las ecuaciones asociadas con el nodo, las cuales se van encargando sucesivamente

para ser transformadas. Esta alternativa hace un uso mínimo de la memoria principal, pues sólo 2 ecuaciones deben estar cargadas para efectuar el proceso; sin embargo requiere de gran número de instrucciones de lectura y escritura, por lo que los tiempos de ejecución llegan a ser unas 7 veces mayores que cuando se usa el método frontal tradicional¹¹.

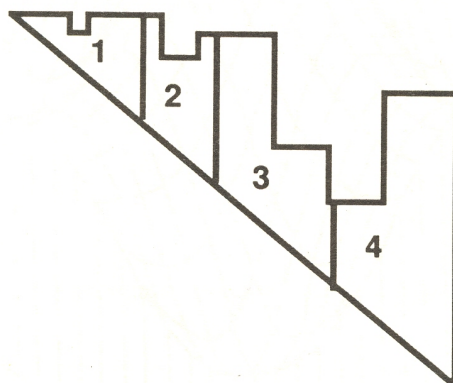


Fig. 8. Partición de la matriz de rigidez

Resultados

El método frontal presentado aquí se probó usando un sistema ortogonal de vigas con 12 grados de libertad por elemento. En la Tabla 1 aparecen algunos datos

sobre tiempos de ejecución y en la Fig. 9 se presenta la malla deformada de 20 x 20 nodos. 🍏

TABLA 1. TIEMPOS DE EJECUCION USANDO EL METODO FRONTAL PROCESADOR 386, 25 MHz, COPROCESADOR MATEMATICO			
MALLA	FRENTE MAXIMO	No. INCOGNITAS	TIEMPO TOTAL DE EJECUCION
5 X 2	24	60	6 seg.
5 X 4	36	120	8 seg.
9 X 8	60	432	29 seg.
15 X 8	60	720	51 seg.
20 X 12	84	1440	2 m. 13 seg.
21 X 50	138	6300	14m. 40 seg.

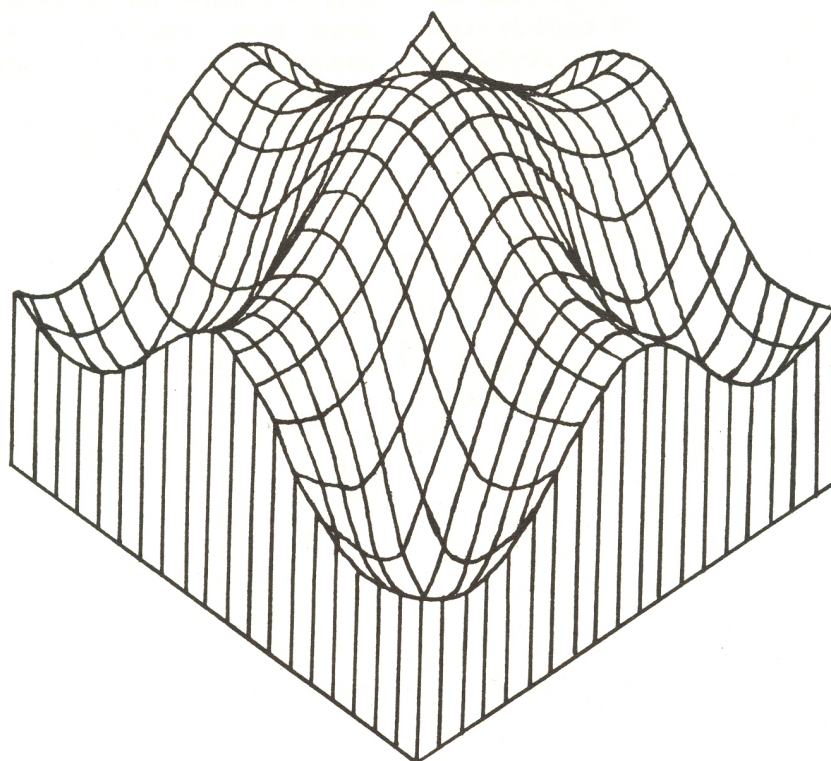


Fig. 9. Malla deformada de vigas ortogonales

BIBLIOGRAFIA

- [1]. MEYER C. "Solution of Linear Equations-State-Of-The Art". Journal of the Structural Division, vol. 99, N° ST7, July 1973.
- [2]. HUGHES, T.J.R. "The finite Element Method". Prentice-Hall, 1987.
- [3]. BATHE K. "Finite Element Procedures in Engineering Analysis". Prentice-Hall, New Jersey, 1982.
- [4]. BETTES P. y BETTES J. "A profile matrix solver with built-in constraint facility". Eng. Comput., vol. 3, September, 1986.
- [5]. ZIENKIEWICS O.C. "The Finite Element Method". Third edition, McGraw-Hill Book Company (U.K.) limited, London, 1967.
- [6]. IRONS B. y Ahmad S. "Techniques of Finite Elements". Ellis Horwood Limited, Chichester, 1980.
- [7]. IRONS B.M. "A frontal solution Program for Finite Element Analysis". Int. J. Numerical Meth. Engng, 2, 5-32, 1970.
- [8]. BEER G. y HAAS W. "A partitioned frontal solver for Finite Element Analysis". International Journal for Numerical Methods in Engineering, vol. 18, 1623-1654, 1982.
- [9]. NOUR-OMID B., and Taylor R.L. "An Algorithm for assembly of stiffness matrices into a compacted data structure". Eng. Comput., vol. 1, December, 1984.
- [10]. ZIENKIEWICS O.C. y Taylor R.L. "The Finite Element Method". Fourth edition. vol. 1, McGraw-Hill Book Company, London, 1989.
- [11]. IDA N. y Lord William. "Solution of Linear Equations for Small Computer Systems". International Journal for Numerical Methods in Engineering, vol. 20, 625-641, 1984.