

PARALELIZACION DEL RESOLVEDOR DE UN MÉTODO DE OPTIMIZACIÓN NO LINEAL

Angel Queizan^a, Agustín Gianni^b, Patricia Gauzellino^c y Victorio Sonzogni^d

^aUTN, FR La Plata y UN La Plata, Fac. Ingeniería, Argentina, aqueizan@frlp.utn.edu.ar

^bUTN, FR La Plata, Argentina, agustingianni@gmail.com

^cUN La Plata, Fac. Ingeniería, Argentina, gauze@fcaglp.unlp.edu.ar

^dCIMEC, INTEC-UNL-CONICET, Argentina, sonzogni@intec.unl.edu.ar

Palabras clave: Optimización no lineal, FAIPA, Procesamiento en paralelo.

Resumen. En este trabajo se trata de reducir el tiempo de computo en un programa de optimización no lineal. Se utiliza el algoritmo FAIPA (Feasible Arc Interior Point Algorithm), de punto interior. Partiendo de un punto situado en la región admisible va moviéndose siempre dentro de esa región. El cálculo de la dirección de búsqueda se hace a partir de tres direcciones básicas. Cada una de estas resulta de resolver un sistema de ecuaciones algebraicas lineales. Los tres sistemas tienen la misma matriz de coeficientes, diferenciándose solo en el vector de términos independientes. El tamaño de estos sistemas es igual a la cantidad de variables de diseño más el número de restricciones. Se propone un resolvedor de tipo LDU (es decir a través de una factorización de la matriz, con L matriz triangular inferior; D matriz diagonal y U matriz triangular superior). Como es sabido la factorización insume una cantidad de operaciones del orden del cubo del tamaño de la matriz, y las sustituciones hacia adelante y hacia atrás una cantidad de operaciones del orden del cuadrado de ese tamaño. Para reducir los tiempos de proceso, se realiza una descomposición LDU por bloques. Esto redundará en una importante ganancia de tiempo en procesamiento secuencial. Este esquema se utiliza para hacer la resolución en paralelo.

1 INTRODUCCION

Diferentes procedimientos pueden ser usados en diseño óptimo de estructuras, tales como técnicas de programación no lineal, algoritmos evolutivos o procedimientos heurísticos. (Hernandez Ibañez, 1990; Penteadó Dias, 1997; Estupiñan, 1999). Un procedimiento que ha mostrado ventajas para esto es el método de punto interior con búsqueda a lo largo de un arco. El algoritmo FAIPA (Feasible Arc Interior Point Algorithm) tiene la ventaja de generar un conjunto de puntos admisibles, incluidos en la región de diseño (aunque no son óptimos), de manera de arribar a un punto óptimo al finalizar el proceso. (Herskovits et al., 2005).

El propósito de este trabajo es presentar una posible modificación de FAIPA usando factorizaciones de tipo LDU por bloques y posibilitando realizar los cálculos en paralelo, (Golub y Van Loan, 1996; Sun y Yuan, 2006). De esta manera pueden obtenerse importantes reducciones en tiempo de corrida. El trabajo comienza con una descripción del problema de optimización no lineal. Luego se describe el algoritmo FAIPA y su implementación paralela por bloques. Finalmente se muestran algunos resultados numéricos y comparaciones entre las resoluciones secuenciales y paralelas.

2 PROGRAMACIÓN NO LINEAL

En general los problemas de optimización pueden expresarse como una programación matemática no lineal con restricciones.

Encontrar

$$\begin{aligned} & \text{Min}_{\underline{x}} f(\underline{x}) \\ & \text{Subject to} \quad \begin{cases} \underline{g}(\underline{x}) \leq 0 \\ \underline{h}(\underline{x}) = 0 \end{cases} \end{aligned} \quad (1)$$

donde $\underline{x} \in \mathfrak{R}^n$ es el vector de variables de diseño; f es la función objetivo $f: \mathfrak{R}^n \rightarrow \mathfrak{R}$; \underline{g} y \underline{h} son vectores que contienen las restricciones, correspondiendo \underline{g} al conjunto de restricciones de desigualdad $\underline{g}: \mathfrak{R}^n \rightarrow \mathfrak{R}^m$, y \underline{h} al conjunto de restricciones de igualdad $\underline{h}: \mathfrak{R}^n \rightarrow \mathfrak{R}^p$. En lo que sigue se supondrá que f , \underline{g} , \underline{h} y sus derivadas son continuas. Con un subrayado se indican vectores y con subrayado doble, matrices.

El problema de diseño óptimo estructural se expresa como una función objetivo que representa (por ej.) el peso estructural; las restricciones de desigualdad garantizan que no se exceda un prefijado nivel de tensiones o de desplazamientos; y las restricciones de igualdad representan las condiciones de equilibrio.

Aplicando las condiciones de optimalidad de primer orden de Karush-Kuhn-Tucker al problema (1) se puede escribir:

$$\underline{\nabla} f(\underline{x}) + \underline{\nabla} \underline{g}(\underline{x}) \cdot \underline{\lambda} + \underline{\nabla} \underline{h}(\underline{x}) \cdot \underline{\mu} = \underline{0} \quad (2)$$

$$\underline{\underline{G}}(\underline{x}) \cdot \underline{\lambda} \quad (3)$$

$$\underline{h}(\underline{x}) = \underline{0} \quad (4)$$

$$\underline{g}(\underline{x}) \leq \underline{0} \quad (5)$$

$$\underline{\lambda} \geq \underline{0} \quad (6)$$

siendo $\underline{\lambda}$ y $\underline{\mu}$ vectores de multiplicadores de Lagrange correspondiente a las desigualdades y a las igualdades, respectivamente; $\underline{G}(\underline{x})$ una matriz diagonal tal que $G_{ii}(\underline{x}) = g_i(\underline{x})$, $\underline{\nabla g}(\underline{x}) \in \mathcal{R}^{n \times m}$ la matriz donde la i-esima columna es el gradiente de la i-esima restricción en desigualdad y $\underline{\nabla h}(\underline{x}) \in \mathcal{R}^{n \times p}$ la matriz donde la i-esima columna es el gradiente de la i-esima restricción en igualdad.

Aplicando el método de Newton (Herskovits y Santos, 1997) se llega al siguiente sistema de ecuaciones lineales por bloques:

$$\left[\begin{array}{c|c|c} \underline{B} & \underline{\nabla g}(\underline{x}) & \underline{\nabla h}(\underline{x}) \\ \hline \underline{\Lambda} \cdot [\underline{\nabla g}(\underline{x})]^T & \underline{G}(\underline{x}) & \underline{0} \\ \hline [\underline{\nabla h}(\underline{x})]^T & \underline{0} & \underline{0} \end{array} \right] \cdot \begin{Bmatrix} \underline{x}_\alpha - \underline{x} \\ \underline{\lambda}_\alpha - \underline{\lambda} \\ \underline{\mu}_\alpha - \underline{\mu} \end{Bmatrix} = - \begin{Bmatrix} \underline{\nabla f}(\underline{x}) + \underline{\nabla g}(\underline{x}) \cdot \underline{\lambda} + \underline{\nabla h}(\underline{x}) \cdot \underline{\mu} \\ \underline{G}(\underline{x}) \cdot \underline{\lambda} \\ \underline{h}(\underline{x}) \end{Bmatrix} \quad (7)$$

donde \underline{B} es la matriz Hessiana $\underline{H}(\underline{x}, \underline{\lambda}, \underline{\mu}) = \nabla^2 f(\underline{x}) + \sum_{i=1}^m \lambda_i \cdot \nabla^2 g_i(\underline{x}) + \sum_{i=1}^p \mu_i \cdot \nabla^2 h_i(\underline{x})$, y $\underline{\Lambda}$ una matriz diagonal tal que $\Lambda_{ii} = \lambda_i$.

3 ALGORITMO FAIPA

Este método se basa en determinar tres direcciones de búsqueda, que se obtienen resolviendo tres sistemas de ecuaciones lineales en las variables primales-duales (variables de diseño y multiplicadores asociados a las desigualdades y a las igualdades) (Herskovits et al, 2005).

El algoritmo FAIPA procede de la siguiente forma:

Paso 1) Cálculo de la dirección de descenso admisible \underline{d} .

La dirección \underline{d} se obtiene por la expresión $\underline{d} = \underline{d}_0 + \rho \underline{d}_1$, donde \underline{d}_0 es una dirección de descenso, \underline{d}_1 es una dirección de desvío y ρ es un parámetro que garantiza que \underline{d} es una dirección de descenso admisible.

Paso 1.1) Para definir las direcciones \underline{d}_0 y \underline{d}_1 es preciso resolver los siguientes sistemas de ecuaciones lineales independientes,

$$\left[\begin{array}{c|c|c} \underline{B}^k & \underline{\nabla}g(\underline{x}^k) & \underline{\nabla}h(\underline{x}^k) \\ \hline \underline{\Lambda} \cdot [\underline{\nabla}g(\underline{x}^k)]^t & \underline{G}(\underline{x}^k) & \underline{0} \\ \hline [\underline{\nabla}h(\underline{x}^k)]^t & \underline{0} & \underline{0} \end{array} \right] \cdot \begin{Bmatrix} \underline{d}_0 \\ \underline{\lambda}_0 \\ \underline{\mu}_0 \end{Bmatrix}^{(k)} = \begin{Bmatrix} -\underline{\nabla}f(\underline{x}^k) \\ \underline{0} \\ -\underline{h}(\underline{x}^k) \end{Bmatrix} \tag{8}$$

y

$$\left[\begin{array}{c|c|c} \underline{B}^k & \underline{\nabla}g(\underline{x}^k) & \underline{\nabla}h(\underline{x}^k) \\ \hline \underline{\Lambda} \cdot [\underline{\nabla}g(\underline{x}^k)]^t & \underline{G}(\underline{x}^k) & \underline{0} \\ \hline [\underline{\nabla}h(\underline{x}^k)]^t & \underline{0} & \underline{0} \end{array} \right] \cdot \begin{Bmatrix} \underline{d}_1 \\ \underline{\lambda}_1 \\ \underline{\mu}_1 \end{Bmatrix}^{(k)} = \begin{Bmatrix} \underline{0} \\ -\underline{\lambda} \\ \underline{0} \end{Bmatrix} \tag{9}$$

Introduciendo nuevas variables de estado $\underline{\lambda}_0 = \underline{\Lambda} \cdot \underline{s}_0$ y $\underline{\lambda}_1 = \underline{\Lambda} \cdot \underline{s}_1$ las matrices de los sistemas (8) y (9) resultan simétricas. Estos quedan expresados:

$$\left[\begin{array}{c|c|c} \underline{B}^k & \underline{\nabla}g(\underline{x}^k) \cdot \underline{\Lambda} & \underline{\nabla}h(\underline{x}^k) \\ \hline \underline{\Lambda} \cdot [\underline{\nabla}g(\underline{x}^k)]^t & \underline{G}(\underline{x}^k) \cdot \underline{\Lambda} & \underline{0} \\ \hline [\underline{\nabla}h(\underline{x}^k)]^t & \underline{0} & \underline{0} \end{array} \right] \cdot \begin{Bmatrix} \underline{d}_0 \\ \underline{s}_0 \\ \underline{\mu}_0 \end{Bmatrix}^{(k)} = \begin{Bmatrix} -\underline{\nabla}f(\underline{x}^k) \\ \underline{0} \\ -\underline{h}(\underline{x}^k) \end{Bmatrix} \tag{10}$$

y

$$\left[\begin{array}{c|c|c} \underline{B}^k & \underline{\nabla}g(\underline{x}^k) \cdot \underline{\Lambda} & \underline{\nabla}h(\underline{x}^k) \\ \hline \underline{\Lambda} \cdot [\underline{\nabla}g(\underline{x}^k)]^t & \underline{G}(\underline{x}^k) \cdot \underline{\Lambda} & \underline{0} \\ \hline [\underline{\nabla}h(\underline{x}^k)]^t & \underline{0} & \underline{0} \end{array} \right] \cdot \begin{Bmatrix} \underline{d}_1 \\ \underline{s}_1 \\ \underline{\mu}_1 \end{Bmatrix}^{(k)} = \begin{Bmatrix} \underline{0} \\ -\underline{\lambda} \\ \underline{0} \end{Bmatrix} \tag{11}$$

Ambos sistemas son ahora simétricos y puede plantearse una resolución con menor número de operaciones. Puede observarse que ambos sistemas tienen la misma matriz de coeficientes.

Si $\underline{d}_0 = \underline{0}$ se detiene el proceso.

Paso 1.2) Se define la función $\phi : \mathfrak{R}^{n+p} \rightarrow \mathfrak{R}$

$$\phi(\underline{x}, \underline{r}) = f(\underline{x}) + \sum_{i=1}^p c_i \cdot |h_i(\underline{x})|$$

donde \underline{c} es un vector de penalización cuyas componentes inicialmente valen 1 y se modifica en cada iteración de modo que si la componente i-ésima es $c(i) < -1.2 \cdot \mu_o(i)$

entonces se cambia al valor $c(i) = -2 \cdot \mu_o(i)$ (Herskovits y Santos, 1997).

Paso 1.3) Para determinar el parámetro ρ , se calcula el gradiente de la función ϕ con respecto a las variables de diseño.

$$\underline{\nabla}\phi_c(\underline{x}) = \begin{bmatrix} \frac{\partial\phi}{\partial x_1} & \frac{\partial\phi}{\partial x_2} & \frac{\partial\phi}{\partial x_3} & \dots & \frac{\partial\phi}{\partial x_n} \end{bmatrix}^t$$

Si $\underline{d}_1^t \underline{\nabla}\phi_c(\underline{x}) > 0$ entonces

$$\rho = \min \left[\varphi \cdot \|\underline{d}_0\|_2^2 ; (\alpha-1) \cdot \underline{d}_0^t \cdot \underline{\nabla}\phi_c(\underline{x}) / \underline{d}_1^t \cdot \underline{\nabla}\phi_c(\underline{x}) \right]$$

de lo contrario

$$\rho = \varphi \cdot \|\underline{d}_0\|_2^2$$

Y la dirección de descenso queda:

$$\underline{d} = \underline{d}_0 + \rho \underline{d}_1$$

Paso 2) Cálculo de la dirección de búsqueda a lo largo de un arco

Paso 2.1) Cálculo de la dirección de descenso posible

$$\tilde{w}_i^I = g_i(\underline{x} + \underline{d}) - g_i(\underline{x}) - [\underline{\nabla}g_i]^t \cdot \underline{d} \quad i = 1, 2, \dots, m$$

$$\tilde{w}_i^E = h_i(\underline{x} + \underline{d}) - h_i(\underline{x}) - [\underline{\nabla}h_i]^t \cdot \underline{d} \quad i = 1, 2, \dots, p$$

Paso 2.2) Se calcula $\tilde{\underline{d}}$ y $\tilde{\underline{\lambda}}$ resolviendo el sistema de ecuaciones lineales,

$$\begin{bmatrix} \underline{B}^k & \underline{\nabla}g(\underline{x}^k) & \underline{\nabla}h(\underline{x}^k) \\ \underline{\Lambda} \cdot [\underline{\nabla}g(\underline{x}^k)]^t & \underline{G}(\underline{x}^k) & \underline{0} \\ [\underline{\nabla}h(\underline{x}^k)]^t & \underline{0} & \underline{0} \end{bmatrix} \cdot \begin{Bmatrix} \tilde{\underline{d}} \\ \tilde{\underline{\lambda}} \\ \tilde{\underline{\mu}} \end{Bmatrix}^{(k)} = \begin{Bmatrix} \underline{0} \\ -\underline{\Lambda} \cdot \underline{w} \\ -\underline{w} \end{Bmatrix} \quad (12)$$

Nuevamente, definiendo el vector auxiliar $\tilde{\underline{\lambda}} = \underline{\Lambda} \cdot \tilde{\underline{s}}$ se puede simetrizar el sistema que queda en la forma:

$$\begin{bmatrix} \underline{B}^k & \underline{\nabla}g(\underline{x}^k) \cdot \underline{\Lambda} & \underline{\nabla}h(\underline{x}^k) \\ \underline{\Lambda} \cdot [\underline{\nabla}g(\underline{x}^k)]^t & \underline{G}(\underline{x}^k) \cdot \underline{\Lambda} & \underline{0} \\ [\underline{\nabla}h(\underline{x}^k)]^t & \underline{0} & \underline{0} \end{bmatrix} \cdot \begin{Bmatrix} \tilde{\underline{d}} \\ \tilde{\underline{s}} \\ \tilde{\underline{\mu}} \end{Bmatrix}^{(k)} = \begin{Bmatrix} \underline{0} \\ -\underline{\Lambda} \cdot \underline{w} \\ -\underline{w} \end{Bmatrix} \quad (13)$$

Una vez definido \tilde{d} se efectúa la búsqueda lineal sobre un arco.

Paso 2.3) Búsqueda lineal sobre un arco.

Se busca un tamaño de paso t que satisfaga el criterio de búsqueda:

$$\min_t \left\{ \phi \left(\underline{x}^{(k)} + t \cdot \underline{d} + t^2 \cdot \tilde{d} \right) \right\}$$

y que verifique las restricciones

$$g_i \left(\underline{x}^{(k)} + t \cdot \underline{d} + t^2 \cdot \tilde{d} \right) \leq 0 \quad i = 1, 2, \dots, m$$

Paso 3) Actualización de los valores de variables de diseño, multiplicadores de Lagrange $\underline{\lambda} > 0$ y matriz Hessiana (Herskovits y Santos, 1997).. Luego vuelve al primer paso del algoritmo.

4 FACTORIZACION LDU POR BLOQUES Y PARALELIZACIÓN

En la sección precedente se describió el algoritmo de FAIPA para calcular las direcciones de búsqueda. En cada iteración se resuelven tres sistemas de ecuaciones lineales con las siguientes características:

- Los sistemas (10), (11), y (13) tienen la misma matriz de coeficientes, simétrica. Una única factorización LDU es necesaria para los tres sistemas.
- Los sistemas (10) y (11) son independientes y pueden ser resueltos simultáneamente. El sistema (13) precisa conocer la solución de los dos anteriores.

4.1 Factorización LDU por bloques

En la iteración k la matriz de coeficientes es:

$$\underline{A}^k = \begin{bmatrix} \underbrace{\underline{B}^k}_{[n \times n]} & \underbrace{\nabla \underline{g}(\underline{x}^k) \cdot \underline{\Lambda}^k}_{[n \times m]} & \underbrace{\nabla h(\underline{x}^k)}_{[n \times p]} \\ \underbrace{\underline{\Lambda}^k \cdot \nabla \underline{g}(\underline{x}^k)^t}_{[m \times n]} & \underbrace{\underline{G}(\underline{x}^k) \cdot \underline{\Lambda}^k}_{[m \times m]} & \underbrace{\underline{0}}_{[m \times p]} \\ \underbrace{\nabla^t h(\underline{x}^k)}_{[p \times n]} & \underbrace{\underline{0}}_{[p \times m]} & \underbrace{\underline{0}}_{[p \times p]} \end{bmatrix} \quad (14)$$

donde

- n es la cantidad de variables de diseño, m y p , la cantidad de restricciones de desigualdad e igualdad, respectivamente.

- $\underline{\underline{B}}^k$ es la matriz Hessiana definida anteriormente, o bien una aproximación obtenida mediante la formulación BFGS modificada por Powell's (técnica usada en programación secuencial cuadrática).
- Las matrices $\underline{\underline{\nabla g}}(\underline{x}^k)$ y $\underline{\underline{\nabla h}}(\underline{x}^k)$ son:

$$\underline{\underline{\nabla g}}(\underline{x}^k) = \begin{bmatrix} \frac{\partial g_1}{\partial x_1} & \dots & \frac{\partial g_m}{\partial x_1} \\ \dots & \dots & \dots \\ \frac{\partial g_1}{\partial x_n} & \dots & \frac{\partial g_m}{\partial x_n} \end{bmatrix} \quad \underline{\underline{\nabla h}}(\underline{x}^k) = \begin{bmatrix} \frac{\partial h_1}{\partial x_1} & \dots & \frac{\partial h_p}{\partial x_1} \\ \dots & \dots & \dots \\ \frac{\partial h_1}{\partial x_n} & \dots & \frac{\partial h_p}{\partial x_n} \end{bmatrix}$$

- La matriz $\underline{\underline{\Lambda}}^k$ es una matriz diagonal que contiene los multiplicadores de Lagrange asociados a las desigualdades, y en la iteración k

$$\underline{\underline{\Lambda}}^k = \begin{bmatrix} \lambda_1^k & \dots & 0 \\ \dots & \dots & \dots \\ 0 & \dots & \lambda_m^k \end{bmatrix}$$

- La matriz $\underline{\underline{G}}(\underline{x}^k)$ es una matriz diagonal que contiene las desigualdades evaluadas en la iteración k

$$\underline{\underline{G}}(\underline{x}^k) = \begin{bmatrix} g_1^k & \dots & 0 \\ \dots & \dots & \dots \\ 0 & \dots & g_m^k \end{bmatrix}$$

Factorizando la matriz $\underline{\underline{A}}^k$ en bloques $\underline{\underline{L}}\underline{\underline{D}}\underline{\underline{U}}$ se obtiene:

$$\underline{\underline{A}}^k = \begin{bmatrix} \underline{\underline{B}}^k & \underline{\underline{\nabla g}}(\underline{x}^k) \cdot \underline{\underline{\Lambda}}^k & \underline{\underline{\nabla h}}(\underline{x}^k) \\ \underline{\underline{\Lambda}}^k \cdot \underline{\underline{\nabla g}}(\underline{x}^k)^t & \underline{\underline{G}}(\underline{x}^k) \cdot \underline{\underline{\Lambda}}^k & \underline{\underline{0}} \\ \underline{\underline{\nabla^t h}}(\underline{x}^k)^t & \underline{\underline{0}} & \underline{\underline{0}} \end{bmatrix} = \begin{bmatrix} \underline{\underline{L}}_{11} & \underline{\underline{0}} & \underline{\underline{0}} \\ \underline{\underline{L}}_{21} & \underline{\underline{L}}_{22} & \underline{\underline{0}} \\ \underline{\underline{L}}_{31} & \underline{\underline{L}}_{32} & \underline{\underline{I}}_A \end{bmatrix} \cdot \begin{bmatrix} \underline{\underline{D}}_{11} & \underline{\underline{0}} & \underline{\underline{0}} \\ \underline{\underline{0}} & \underline{\underline{D}}_{22} & \underline{\underline{0}} \\ \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{A}} \end{bmatrix} \cdot \begin{bmatrix} \underline{\underline{U}}_{11} & \underline{\underline{U}}_{12} & \underline{\underline{U}}_{13} \\ \underline{\underline{0}} & \underline{\underline{U}}_{22} & \underline{\underline{U}}_{23} \\ \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{I}}_D \end{bmatrix} \quad (15)$$

Este procedimiento por bloques es como sigue:

- Para el bloque [1,1] de $\underline{\underline{A}}^k$ y teniendo en cuenta que $\underline{\underline{B}}^k$ es simétrica y definida positiva

$$\underline{\underline{A}}_{11} = \underline{\underline{B}}^k = \underline{\underline{L}}_{11} \cdot \underline{\underline{D}}_{11} \cdot \underline{\underline{U}}_{11} = \underline{\underline{L}}_{11} \cdot \underline{\underline{D}}_{11} \cdot \left[\underline{\underline{L}}_{11} \right]^t \quad (16)$$

- Para el bloque [1,2] de $\underline{\underline{A}}^k$

$$\underline{\underline{A}}_{12} = \underline{\underline{\nabla}}g \left(\underline{\underline{x}}^{(k)} \right) \cdot \underline{\underline{\Lambda}}^{(k)} = \underbrace{\begin{bmatrix} \lambda_1 \frac{\partial g_1}{\partial x_1} & \dots & \lambda_m \frac{\partial g_m}{\partial x_1} \\ \dots & \dots & \dots \\ \lambda_1 \frac{\partial g_1}{\partial x_n} & \dots & \lambda_m \frac{\partial g_m}{\partial x_n} \end{bmatrix}}_{[n \times m]}^{(k)} = \underline{\underline{L}}_{11} \cdot \underline{\underline{D}}_{11} \cdot \underline{\underline{U}}_{12}$$

De esta última expresión obtenemos m sistema triangulares inferiores, con n incógnitas cada uno

$$\underline{\underline{L}}_{11} \cdot \underline{\underline{D}}_{11} \cdot \left\{ \underline{\underline{U}}_{12} \right\}_{(:,k)} = \left\{ \underline{\underline{A}}_{12} \right\}_{(:,k)} = \lambda_k \left\{ \frac{\partial g_k}{\partial x_1} \quad \frac{\partial g_k}{\partial x_2} \quad \dots \quad \frac{\partial g_k}{\partial x_n} \right\}^t \quad k = 1, 2, \dots, m \quad (17)$$

siendo $\left\{ \underline{\underline{U}}_{12} \right\}_{(:,k)}$ la columna k de $\underline{\underline{U}}_{12}$ y $\left\{ \underline{\underline{A}}_{12} \right\}_{(:,i)}$ la columna k de $\underline{\underline{A}}_{12}$. Resolviendo los m sistemas se tienen las m columnas de $\underline{\underline{U}}_{12}$.

- Para el bloque [1,3] de $\underline{\underline{A}}^k$

$$\underline{\underline{A}}_{13} = \underline{\underline{\nabla}}h \left(\underline{\underline{x}}^{(k)} \right) = \underbrace{\begin{bmatrix} \frac{\partial h_1}{\partial x_1} & \frac{\partial h_2}{\partial x_1} & \dots & \frac{\partial h_p}{\partial x_1} \\ \frac{\partial h_1}{\partial x_2} & \frac{\partial h_2}{\partial x_2} & \dots & \frac{\partial h_p}{\partial x_2} \\ \dots & \dots & \dots & \dots \\ \frac{\partial h_1}{\partial x_n} & \frac{\partial h_2}{\partial x_n} & \dots & \frac{\partial h_p}{\partial x_n} \end{bmatrix}}_{[n \times p]}^{(k)} = \underline{\underline{L}}_{11} \cdot \underline{\underline{D}}_{11} \cdot \underline{\underline{U}}_{13}$$

De esta última expresión se tienen p sistemas triangulares inferiores, cada uno con n incógnitas.

$$\underline{\underline{L}}_{11} \cdot \underline{\underline{D}}_{11} \cdot \left\{ \underline{\underline{U}}_{13} \right\}_{(:,k)} = \left\{ \underline{\underline{A}}_{13} \right\}_{(:,k)} = \left[\frac{\partial h_k}{\partial x_1} \quad \frac{\partial h_k}{\partial x_2} \quad \dots \quad \frac{\partial h_k}{\partial x_n} \right]^t \quad k = 1, 2, \dots, p \quad (18)$$

donde $\left\{ \underline{\underline{U}}_{13} \right\}_{(:,k)}$ es la columna k de $\underline{\underline{U}}_{13}$ y $\left\{ \underline{\underline{A}}_{13} \right\}_{(:,k)}$ la columna k de $\underline{\underline{A}}_{13}$. Resolviendo los p sistemas se tienen p columnas de $\underline{\underline{U}}_{13}$.

- Para el bloque [2,1] de $\underline{\underline{A}}^k$

$$\underline{\underline{U}}_{12} = \left[\underline{\underline{L}}_{21} \right]^t \rightarrow \left[\underline{\underline{U}}_{12} \right]^t = \underline{\underline{L}}_{21} \quad (19)$$

- Para el bloque [3,1] de $\underline{\underline{A}}^k$, análogamente,

$$\underline{\underline{U}}_{13} = \left[\underline{\underline{L}}_{31} \right]^t \rightarrow \left[\underline{\underline{U}}_{13} \right]^t = \underline{\underline{L}}_{31} \quad (20)$$

- Para el bloque [2,2] de $\underline{\underline{A}}^k$ se tiene:

$$\underline{\underline{A}}_{22} = \underline{\underline{G}}(\underline{\underline{x}}) \cdot \underline{\underline{\Lambda}} = \begin{bmatrix} \lambda_1 g_1(\underline{\underline{x}}) & 0 & \dots & 0 \\ 0 & \lambda_2 g_2(\underline{\underline{x}}) & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \lambda_m g_m(\underline{\underline{x}}) \end{bmatrix} = \underline{\underline{L}}_{21} \cdot \underline{\underline{D}}_{11} \cdot \underline{\underline{U}}_{12} + \underline{\underline{L}}_{22} \cdot \underline{\underline{D}}_{22} \cdot \underline{\underline{U}}_{22}$$

A partir de (19) y por la forma de la matriz $\underline{\underline{A}}_{22}$, la matriz

$$\underline{\underline{L}}_{22} \cdot \underline{\underline{D}}_{22} \cdot \underline{\underline{U}}_{22} = \underline{\underline{A}}_{22} - \underline{\underline{L}}_{21} \cdot \underline{\underline{D}}_{11} \cdot \left[\underline{\underline{L}}_{21} \right]^t$$

es simétrica, por tanto se factoriza $\underline{\underline{L}}_{22} \cdot \underline{\underline{D}}_{22} \cdot \underline{\underline{L}}_{22}^t$ o sea,

$$\underline{\underline{L}}_{22} \cdot \underline{\underline{D}}_{22} \cdot \left[\underline{\underline{L}}_{22} \right]^t = \underline{\underline{A}}_{22} - \underline{\underline{L}}_{21} \cdot \underline{\underline{D}}_{11} \cdot \left[\underline{\underline{L}}_{21} \right]^t \quad (21)$$

- Para el bloque [2,3] de $\underline{\underline{A}}^k$ se tiene:

$$\underline{\underline{A}}_{23} = \underline{\underline{L}}_{21} \cdot \underline{\underline{D}}_{11} \cdot \underline{\underline{U}}_{13} + \underline{\underline{L}}_{22} \cdot \underline{\underline{D}}_{22} \cdot \underline{\underline{U}}_{23} = \underline{\underline{0}}$$

y de (20) se puede escribir:

$$\underline{\underline{L}}_{22} \cdot \underline{\underline{D}}_{22} \cdot \underline{\underline{U}}_{23} = -\underline{\underline{L}}_{21} \cdot \underline{\underline{D}}_{11} \cdot \left[\underline{\underline{L}}_{31} \right]^t$$

De esta última expresión se tienen p sistemas triangulares independientes, cada uno con m incógnitas.

$$\underline{\underline{L}}_{22} \cdot \underline{\underline{D}}_{22} \cdot \left\{ \underline{\underline{U}}_{23} \right\}_{(:,k)} = -\left\{ \underline{\underline{L}}_{21} \cdot \underline{\underline{D}}_{11} \cdot \left[\underline{\underline{L}}_{31} \right]^t \right\}_{(:,k)} \quad k = 1, 2, \dots, m \quad (22)$$

donde $\left\{ \underline{\underline{U}}_{23} \right\}_{(:,k)}$ es la columna k de $\underline{\underline{U}}_{23}$ y $\left\{ \underline{\underline{L}}_{21} \cdot \underline{\underline{D}}_{11} \cdot \left[\underline{\underline{L}}_{31} \right]^t \right\}_{(:,k)}$ la columna k que resulta

del producto matricial $\underline{\underline{L}}_{21} \cdot \underline{\underline{D}}_{11} \cdot \left[\underline{\underline{L}}_{31} \right]^t$. A partir de aquí se obtiene $\underline{\underline{U}}_{23}$.

- Para el bloque [3,2] de $\underline{\underline{A}}^k$, se calcula

$$\left[\underline{\underline{L}}_{32} \right]^t = \underline{\underline{U}}_{23} \rightarrow \underline{\underline{L}}_{32} = \left[\underline{\underline{U}}_{23} \right]^t \quad (23)$$

- Para el bloque [3,3] de $\underline{\underline{A}}^k$, se tiene

$$\underline{\underline{A}}_{33} = \underline{\underline{L}}_{31} \cdot \underline{\underline{D}}_{11} \cdot \underline{\underline{U}}_{13} + \underline{\underline{L}}_{32} \cdot \underline{\underline{D}}_{22} \cdot \underline{\underline{U}}_{23} + \underline{\underline{A}}^{\sim} = \underline{\underline{0}}$$

De (20) y (23) se puede escribir

$$\underline{\underline{A}}^{\sim} = -\left[\underline{\underline{L}}_{31} \cdot \underline{\underline{D}}_{11} \cdot \left[\underline{\underline{L}}_{31} \right]^t + \underline{\underline{L}}_{32} \cdot \underline{\underline{D}}_{22} \cdot \left[\underline{\underline{L}}_{32} \right]^t \right]$$

Dado que la matriz $\underline{\underline{A}}^{\sim}$ es simétrica, usamos una factorización del tipo LDL^t

$$\underline{\underline{L}}_{\tilde{A}} \cdot \underline{\underline{D}}_{\tilde{A}} \cdot \left[\underline{\underline{L}}_{\tilde{A}} \right]^t = -\left[\underline{\underline{L}}_{31} \cdot \underline{\underline{D}}_{11} \cdot \left[\underline{\underline{L}}_{31} \right]^t + \underline{\underline{L}}_{32} \cdot \underline{\underline{D}}_{22} \cdot \left[\underline{\underline{L}}_{32} \right]^t \right] \quad (24)$$

Los tres sistemas del algoritmo FAIPA se pueden expresar de la siguiente forma:

$$\begin{bmatrix} \underline{U}_{11}^t & \underline{0} & \underline{0} \\ \underline{U}_{12}^t & \underline{L}_{22} & \underline{0} \\ \underline{U}_{13}^t & \underline{U}_{23}^t & \underline{L}_{\bar{A}} \end{bmatrix} \cdot \begin{bmatrix} \underline{D}_{11} & \underline{0} & \underline{0} \\ \underline{0} & \underline{D}_{22} & \underline{0} \\ \underline{0} & \underline{0} & \underline{D}_{\bar{A}} \end{bmatrix} \cdot \begin{bmatrix} \underline{U}_{11} & \underline{U}_{12} & \underline{U}_{13} \\ \underline{0} & \underline{U}_{22} & \underline{U}_{23} \\ \underline{0} & \underline{0} & \underline{L}_{\bar{A}}^t \end{bmatrix} \cdot \begin{Bmatrix} \underline{c}_1 \\ \underline{c}_2 \\ \underline{c}_3 \end{Bmatrix} = \begin{Bmatrix} \underline{b}_1 \\ \underline{b}_2 \\ \underline{b}_3 \end{Bmatrix}$$

donde:

	Primer sistema	Segundo sistema	Tercer sistema
c_1	d_0	d_1	\tilde{d}
c_2	λ_0	λ_1	$\tilde{\lambda}$
c_3	μ_0	μ_1	$\tilde{\mu}$
b_1	$-\nabla f(\underline{x})$	$\underline{0}$	$\underline{0}$
b_2	$\underline{0}$	$-\underline{\lambda}$	$-\underline{\Lambda} \cdot \underline{\tilde{w}}^t$
b_3	$-h(\underline{x})$	$\underline{0}$	$-\underline{\tilde{w}}^E$

Cada sistema se resuelve como tres sistemas de menor complejidad.

4.2 Estrategias de implementación paralela

Para generar la matriz \underline{A}^k se deben calcular las derivadas de g y de h con respecto a las variables de diseño. Estas matrices se dividen en bloques de filas asignadas a diferentes procesos. En otro nivel de paralelismo, los sistemas desacoplados triangular inferior, se resuelven en diferentes procesadores. Finalmente los sistemas (10), (11) y (13) se calculan también en dos procesadores.

5 EJEMPLO DE APLICACIÓN

Como ejemplo de aplicación se ha tomado una viga en estado plano de tensiones, con espesor variable, de 10 cm de largo por 10 cm de ancho, con carga externa como se muestra en la figura 1.

Se resolvió usando elementos finitos triangulares de tres nodos y con espesor variable. En la figura 2 se muestran distintas mallas usadas.

El propósito es encontrar un diseño estructural óptimo con mínimo peso y para el cual las tensiones de Von Mises no excedan 800 kg/cm². Las variables de diseño son los espesores nodales y las variables de estado los desplazamientos nodales. Los espesores están acotados por 1cm y 0.1 cm. En todos los casos el espesor inicial es 0.5 cm.

Los espesores para el diseño óptimo se muestran en la Figura 3, y los resultados principales están en la Tabla I.

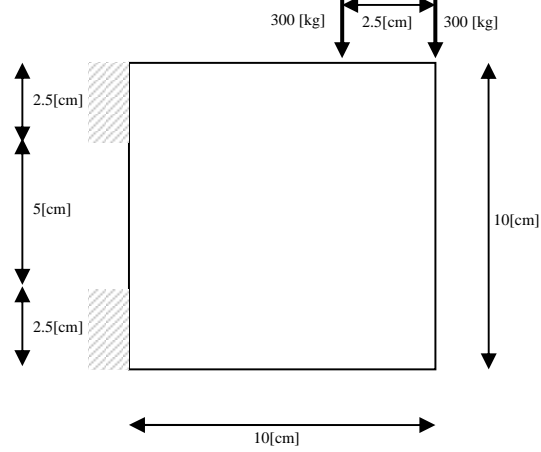
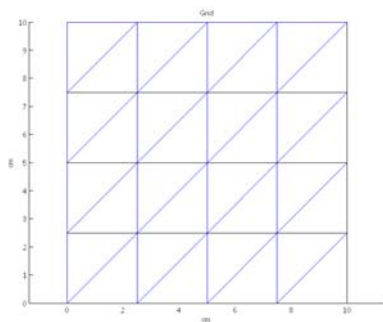
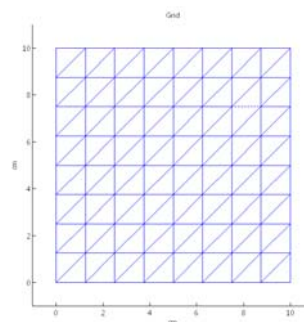


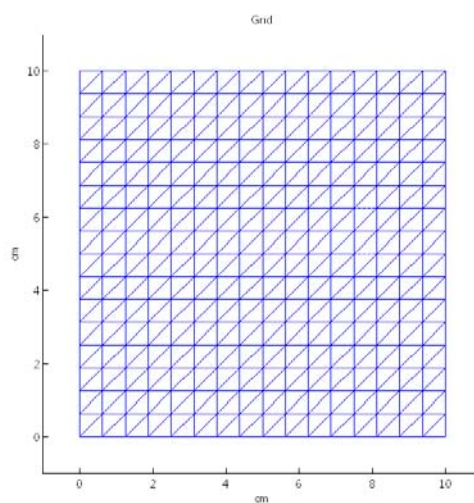
Figura 1: Problema mecánico



Grid_4x4



Grid_8x8



Grid_16x16

Figura 2: Mallas de elementos finitos

La Tabla II muestra el tiempo de procesamiento para la solución secuencial del sistema completo; para la solución secuencial realizada por bloques; y para la solución paralela por bloques. Se precisaron 109 iteraciones para convergencia para la malla de 16 x 16 elementos. Para esta malla no se obtuvo la solución clásica secuencial del sistema completo hasta convergencia. Los tiempos mostrados en la tabla corresponde a 10 iteraciones. Para comparación se muestran también los tiempos para 10 iteraciones obtenidos con la solución por bloques y paralela por bloques, además de aquellos para la convergencia completa. La solución paralela se obtuvo con 2 procesadores.

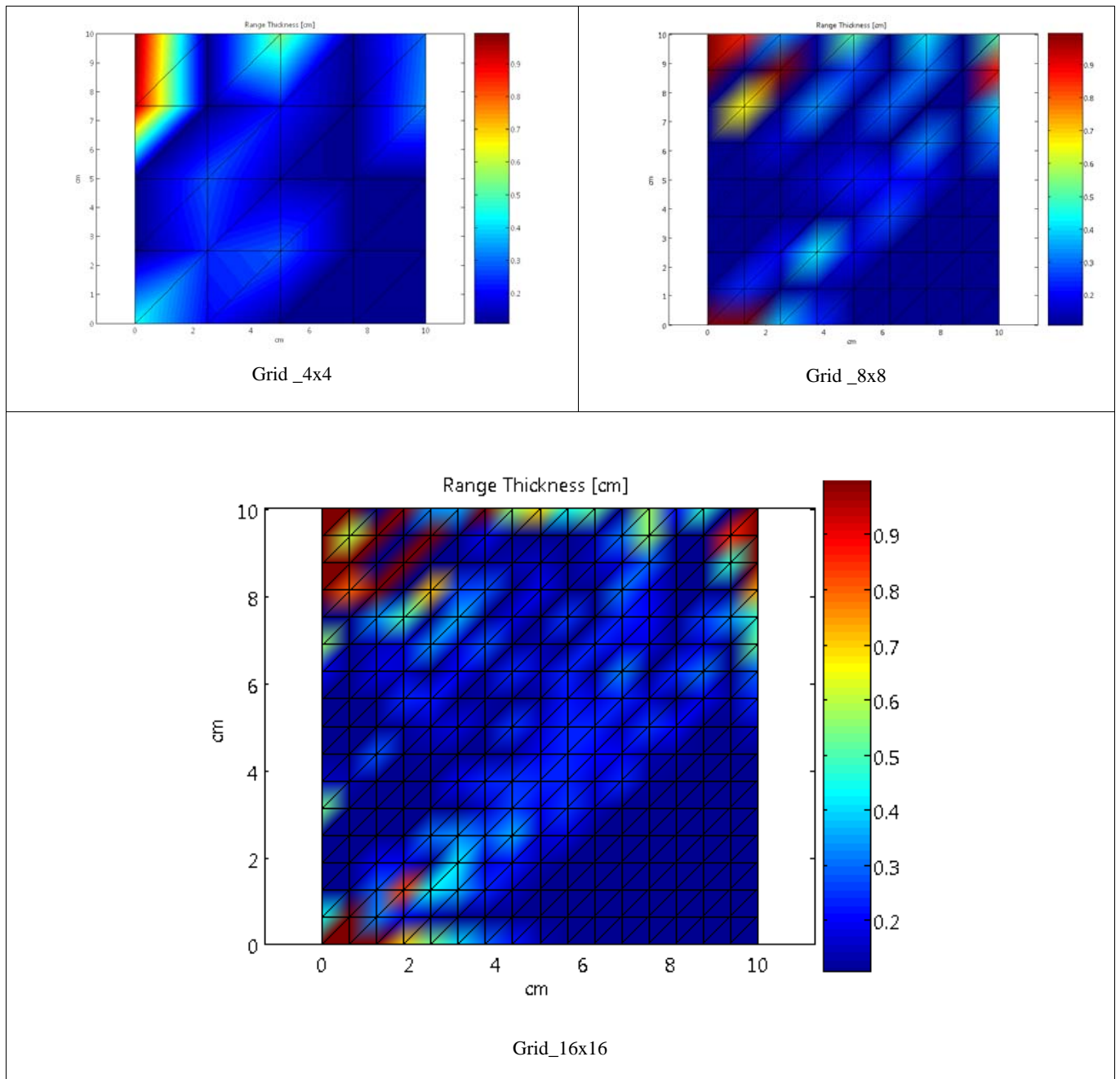


Figura 3: Espesores para el diseño óptimo

Malla	Función Objetivo		Iteraciones	Igualdades	Desigualdades
	Valor inicial	Valor final			
grid_4x4	50	21.1778	28	42	75
grid_8x8	50	20.9950	50	150	243
grid_16x16	50	22.1443	109	554	867

Tabla 1: . Principales resultados para el problema de optimización.

Malla	Variables de diseño	Tamaño sistema	Tiempo procesamiento [s]		
			LU serial	Block LDL ^t serial	Block LDL ^t paralelo
Gris_4x4	67	184	74.91	54.43	56
Gris_8x8	231	624	2125.65 (*)	1156.08 (**)	1170
Gris 16x16	843	2264	(10 iter.) 14738.43	(10 iter.) 5557.28 (109 iter.) 44505.13 (***)	(10 iter.) 3996.31 (109 iter.) 32512.41 (****)

(*) 2125.65 = 35 min 25.65 s

(**) 19 min 16.083 s

(***) 44505.13 s = 12 hs 21 min 45.13 s

(****) 32512.41 s = 9 hs 1 min 52.41 s

Tabla 2. Tiempo de procesamiento para el problema de optimización

6 CONCLUSIONES

En este trabajo se estudia la solución del método FAIPA con el objeto de reducir el tiempo de procesamiento. Los sistemas de ecuaciones para calcular las direcciones de búsqueda se pueden reducir a la factorización de una única matriz, seguido de tres sustituciones con diferentes vectores independientes. Más aún, la factorización LDL^t se divide de forma de realizar estas operaciones por bloques. Esto ha producido ganancia de tiempo de procesamiento aún secuencialmente. Para pequeños sistemas no se observa una mejora en tiempo de procesamiento, pero podrá apreciarse en problemas mayores.

AGRADECIMIENTOS

Este trabajo ha recibido apoyo de: Universidad Tecnológica Nacional; CONICET; ANPCyT, Universidad Nacional de la Plata y Universidad Nacional del Litoral.

REFERENCES

- Estupiñán, J., Oñate, E. Suarez, B., Métodos Evolutivos en la Optimización Topológica., *Monografía CIMNE N°47*. 1999.
- Golub, G. y. Van Loan, C.F., *Matrix Computations*. 1996
- Hernandez Ibáñez, S., *Métodos de Diseño Optimo de Estructuras*. Editorial Paraninfo. 1990
- Herskovits, J., Mappa, P. Goulart, E. y Mota Soares, C.M., *Mathematical programming*

- models and algorithms for engineering design optimization. *Comput. Methods Appl. Mech. Engrg*, 194: 3244-3268. 2005.
- Herskovits, J and Santos, G. On the Computer Implementation of Feasible Direction Interior Point Algorithms For Nonlinear Optimization. 1997
- Penteado Dias, G., Optimización de formas de sólidos elásticos a través de técnicas de programación matemática. *Tesis Doctoral – Universidad Federal de Río de Janeiro*. 1997.
- Sun, W. Yuan, Y., *Optimization Theory and Methods Nonlinear Programming*. 2006