



DEPARTAMENTO DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

Trabajo Práctico N° 1

“Con 15 θ s discretizo alto horno”

Segundo cuatrimestre de 2015

Métodos Numéricos

Integrante	LU	Correo electrónico
Frizzo, Franco	013/14	francofrizzo@gmail.com
Martínez, Manuela	160/14	martinez.manuela.22@gmail.com
Rabinowicz, Lucía	105/14	lu.rabinowicz@gmail.com



Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires

Ciudad Universitaria - (Pabellón I/Planta Baja)
Intendente Güiraldes 2160 - C1428EGA
Ciudad Autónoma de Buenos Aires - Rep. Argentina
Tel/Fax: (54 11) 4576-3359
<http://www.fcen.uba.ar>

Resumen

En el presente trabajo, se estudia la utilización de métodos numéricos para calcular las temperaturas en el interior de la pared de un alto horno de fusión de metales, con el objetivo de evaluar así el riesgo de colapso de la misma. Para este fin, el dominio del problema es discretizado, obteniendo un sistema de ecuaciones lineales, en cuya resolución se emplean los métodos de Eliminación Gaussiana y Factorización LU. Además, se proponen técnicas para la estimación de una isotermia crítica en el interior de la pared y para el cálculo de un índice de peligrosidad que permita determinar el riesgo de colapso. Luego se realizan pruebas experimentales, para así extraer conclusiones tanto acerca del rendimiento temporal de los algoritmos implementados como del comportamiento del sistema y la precisión de las técnicas de evaluación propuestas.

Palabras clave: Eliminación Gaussiana, Factorización LU, sistemas de ecuaciones lineales, alto horno, cálculo de isotermas

Índice

1. Introducción	4
2. Desarrollo	5
2.1. Conceptos teóricos	5
2.1.1. Eliminación Gaussiana	7
2.1.2. Factorización LU	8
2.2. Modelado del sistema	8
2.3. Aplicabilidad de los métodos elegidos	11
2.4. Interpretación de los resultados	14
2.4.1. Estimación de la posición de la isoterma crítica	14
2.4.2. Cálculo del índice de peligrosidad	16
2.5. Implementación	16
3. Experimentación y discusión	17
3.1. Pruebas sobre el rendimiento de los métodos	17
3.1.1. Experimento 1: Tiempo de ejecución según granularidad de la discretización	17
3.1.2. Experimento 2: Tiempo según número de instancias	19
3.2. Pruebas sobre el comportamiento del sistema	20
3.2.1. Experimento 3: Isoterma según temperatura en pared externa	20
3.2.2. Experimento 4: Isoterma según granularidad	22
3.2.3. Experimento 5: Índice de peligrosidad según ancho de la pared	27
4. Conclusiones	29
Apéndices	30
A. Enunciado del trabajo práctico	30
A.1. Introducción	30
A.2. El Modelo	30
A.3. Enunciado	31
A.4. Programa y formato de archivos	33
A.5. Fechas de entrega	34
Referencias	35

1. Introducción

Un *alto horno* es un horno cilíndrico en cuyo interior se realiza la fusión de metales a temperaturas muy elevadas. Dado que las paredes del horno se encuentran sometidas a estas condiciones extremas, resulta importante conocer, por cuestiones de seguridad, si su estructura corre riesgo de colapsar a causa del calor. Este problema físico es el que se pretende resolver, a través de un modelo matemático, en el transcurso de este trabajo.

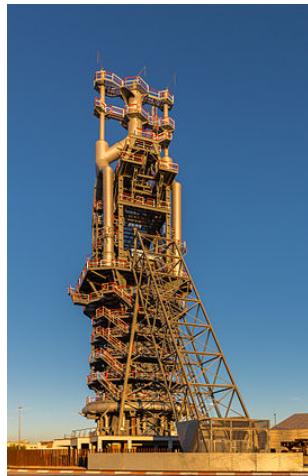


Figura 1: Alto horno.

La temperatura en el interior del horno es constante y conocida; además, se cuenta con sensores ubicados a intervalos regulares en el exterior de la pared, que proporcionan información sobre la temperatura (variable) de la misma. A partir de esta información, se pretende en primer lugar estimar el valor de la temperatura en los puntos correspondientes al interior de la pared.

Para encarar el problema de forma computacional, su dominio será discretizado mediante un modelo matemático; es decir, se seleccionará un conjunto finito de puntos del mismo en el que calcular los resultados. Esto permitirá obtener los mismos a través de la resolución de un sistema de ecuaciones lineales. Por este motivo, serán de interés los métodos que permitan encontrar soluciones a este tipo de sistemas. Se utilizarán dos de ellos: el método de Eliminación Gaussiana y el de Factorización LU.

Luego, teniendo en cuenta los resultados obtenidos, se buscará interpretarlos para evaluar la peligrosidad de la estructura. En particular, se considerará una temperatura crítica, con el objetivo de calcular la posición de la isoterma correspondiente a la misma, es decir, el conjunto de puntos de la pared que se encuentran a dicha temperatura. La cercanía de esta isoterma a la pared externa del horno será la que determinará la existencia de una situación de riesgo de colapso de la estructura. En este sentido, se estudiará también la elaboración de un criterio que permita decidir el índice de peligrosidad de la misma, en función al resultado obtenido.

2. Desarrollo

2.1. Conceptos teóricos

La resolución de *sistemas de ecuaciones lineales* es un problema recurrente en el análisis numérico, ya que estos son útiles a la hora de modelar matemáticamente el comportamiento de problemas provenientes de diversas disciplinas, como la física y la ingeniería, para ser tratados en forma computacional. Es por eso que resulta de interés desarrollar algoritmos que permitan, de manera tan eficiente como sea posible, encontrar soluciones a estos sistemas.

Un sistema de m ecuaciones lineales con n incógnitas tiene la forma

$$\begin{cases} a_{1,1}x_1 + a_{1,2}x_2 + \dots + a_{1,n}x_n = b_1 \\ a_{2,1}x_1 + a_{2,2}x_2 + \dots + a_{2,n}x_n = b_2 \\ \vdots \qquad \vdots \qquad \vdots \qquad \vdots \qquad \vdots \\ a_{m,1}x_1 + a_{m,2}x_2 + \dots + a_{m,n}x_n = b_m \end{cases}$$

donde $a_{j,k}$ y b_k (para $j = 1, \dots, m$; $k = 1, \dots, n$) son constantes reales. Por *resolver* el sistema se entiende hallar un conjunto de valores para las incógnitas x_1, \dots, x_n de forma que todas las ecuaciones se satisfagan simultáneamente.

Una manera frecuente de representar un sistema de ecuaciones lineales, que será utilizada en este trabajo, es la matricial, que consiste en expresarlo en la forma

$$\mathbf{A}x = b$$

donde

$$\mathbf{A} = \begin{bmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \dots & a_{m,n} \end{bmatrix} \in \mathbb{R}^{m \times n} \quad x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \in \mathbb{R}^n \quad b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix} \in \mathbb{R}^m$$

Dado un sistema de ecuaciones lineales, existe un conjunto de operaciones que, aplicadas sobre las ecuaciones, generan sistemas *equivalentes*, es decir, que tienen las mismas soluciones. Más específicamente, estas operaciones son:

- (a) Multiplicar una ecuación por una constante $\lambda \in \mathbb{R}$ no nula.
- (b) Sumar a una ecuación el resultado de multiplicar otra por una constante $\lambda \in \mathbb{R}$.
- (c) Intercambiar el orden de dos ecuaciones.

Lo mismo vale para sistemas en forma matricial, si se consideran las operaciones análogas entre las filas de la matriz.

Más aún, puede probarse que a partir de todo sistema puede obtenerse otro equivalente que se encuentra en forma *triangular o reducida* (ver [1, p. 358]). Este resultado es importante porque, dado un sistema en forma triangular (superior o inferior), encontrar sus soluciones es sumamente sencillo, utilizando los algoritmos de *sustitución hacia adelante* y el de *sustitución hacia atrás*.

Cabe destacar que un sistema de ecuaciones lineales puede no tener soluciones, tener solución única, o tener infinitas soluciones. En adelante se considerarán únicamente sistemas cuadrados (con igual número de ecuaciones que de incógnitas) cuya solución sea única, quedando los demás

casos fuera del alcance de este trabajo. Bajo esta hipótesis, un sistema $\mathbf{A}x = b$ cumple la importante propiedad de que la matriz \mathbf{A} es *inversible*. En el caso de que \mathbf{A} se encuentre en forma triangular, esto equivale a decir que no hay elementos nulos en su diagonal.

El algoritmo de *sustitución hacia adelante* considera un sistema de ecuaciones $\mathbf{L}x = b$, donde \mathbf{L} es una matriz triangular inferior, y devuelve un vector s solución del sistema. Lo hace aprovechando el hecho de, dado que la primer ecuación depende solamente de la primera incógnita, el valor de esta última puede ser despejado de forma inmediata; luego, sustituyendo este valor en la segunda ecuación, puede despejarse la segunda incógnita, y así sucesivamente hasta resolver el sistema completo. El algoritmo de *sustitución hacia atrás* es análogo al anterior, pero considera un sistema de ecuaciones $\mathbf{U}x = b$, con \mathbf{U} triangular superior, y realiza el proceso recorriendo las filas de la matriz en orden inverso, desde abajo hacia arriba. Ambos algoritmos tienen una complejidad cuadrática en la cantidad de incógnitas.

Algoritmo 1: Sustitución hacia adelante

Datos: $\mathbf{A} \in \mathbb{R}^{n \times n}, b \in \mathbb{R}^n$

Resultado: $s \in \mathbb{R}^n$, solución del sistema $\mathbf{A}x = b$, considerando a \mathbf{A} como una matriz triangular inferior, sin elementos nulos en la diagonal

```

1 para  $j \leftarrow 1$  a  $n$  hacer
2   |   suma  $\leftarrow 0$ 
3   |   para  $k \leftarrow 1$  a  $j$  hacer
4   |     |   suma  $\leftarrow suma + s_k \cdot a_{j,k}$ 
5   |   fin
6   |    $s_j \leftarrow \frac{b_j - suma}{a_{j,j}}$ 
7 fin

```

Algoritmo 2: Sustitución hacia atrás

Datos: $\mathbf{A} \in \mathbb{R}^{n \times n}, b \in \mathbb{R}^n$

Resultado: $s \in \mathbb{R}^n$, solución del sistema $\mathbf{A}x = b$, considerando a \mathbf{A} como una matriz triangular superior, sin elementos nulos en la diagonal

```

1 para  $j \leftarrow n$  a 1 hacer
2   |   suma  $\leftarrow 0$ 
3   |   para  $k \leftarrow j + 1$  a  $n$  hacer
4   |     |   suma  $\leftarrow suma + s_k \cdot a_{j,k}$ 
5   |   fin
6   |    $s_j \leftarrow \frac{b_j - suma}{a_{j,j}}$ 
7 fin

```

La sencillez con la que pueden resolverse sistemas cuando se encuentran en forma triangular es el principio en que se cimientan los dos métodos que se utilizarán en este trabajo, *Eliminación Gaussiana* y *Factorización LU*. La idea central de ambos es llevar el sistema de ecuaciones que pretende resolverse a otro equivalente en forma triangular, para así poder aplicar los algoritmos recién expuestos y encontrar su solución.

2.1.1. Eliminación Gaussiana

El método de *Eliminación Gaussiana* itera sobre las columnas de la matriz, colocando ceros en todas las posiciones que se encuentran por debajo de la diagonal. Para hacer esto, en la k -ésima iteración, se resta a todas las filas a partir de la $k+1$ un múltiplo de la fila k -ésima, con un factor elegido convenientemente.

Más formalmente, supongamos que se pretende resolver el sistema $\mathbf{A}x = b$. Con la notación $\mathbf{A}^{(k)}$ se hará referencia a la matriz que se obtiene luego de aplicar k iteraciones del algoritmo sobre \mathbf{A} , mientras que $\mathbf{A}_{k,*}$ denotará la k -ésima fila de la matriz \mathbf{A} .

Inicialmente, se tiene que $\mathbf{A}^{(0)} = \mathbf{A}$. Luego, tras cada iteración, se obtiene:

$$\mathbf{A}^{(k)} = \begin{bmatrix} \mathbf{A}_{1,*}^{(k-1)} \\ \vdots \\ \mathbf{A}_{k,*}^{(k-1)} \\ \mathbf{A}_{k+1,*}^{(k-1)} - m_{k+1,k} \mathbf{A}_{k,*}^{(k-1)} \\ \vdots \\ \mathbf{A}_{n,*}^{(k-1)} - m_{n,k} \mathbf{A}_{k,*}^{(k-1)} \end{bmatrix}$$

donde $m_{i,k} = \frac{a_{i,k}}{a_{k,k}}$. Esta elección de los multiplicadores tiene la particularidad de anular todos los elementos de la columna k por debajo de la diagonal. Esto asegura que, al completar el algoritmo, luego de iterar sobre todas las columnas, la matriz obtenida será triangular superior.

En su forma más básica, el método falla si, al comienzo de alguna iteración, es nulo el elemento de la diagonal correspondiente a la columna sobre la que se debe trabajar, es decir, si para algún $k = 1, \dots, n$ se cumple $a_{k,k}^{(k-1)} = 0$. Para salvar esta dificultad, se utiliza una técnica conocida como *pivoteo*, que consiste en alterar el orden de las filas o de las columnas de la matriz. No obstante, bajo determinadas hipótesis, puede garantizarse que el algoritmo es aplicable sin necesidad de pivoteo; este será el caso en el contexto particular de este trabajo, por lo que no se entrará en más detalles acerca de esta técnica.

Algoritmo 3: Eliminación Gaussiana

Datos: $\mathbf{A} \in \mathbb{R}^{n \times n}, b \in \mathbb{R}^n$

Resultado: $s \in \mathbb{R}^n$, solución del sistema $\mathbf{A}x = b$

```

1 para  $k \leftarrow 1$  a  $n$  hacer
2   para  $j \leftarrow k+1$  a  $n$  hacer
3      $m \leftarrow \frac{a_{j,k}}{a_{k,k}}$ 
4     para  $\ell \leftarrow k$  a  $n$  hacer
5        $a_{j,\ell} \leftarrow a_{j,\ell} - m \cdot a_{k,\ell}$ 
6     fin
7      $b_j \leftarrow b_j - m \cdot b_k$ 
8   fin
9 fin
10  $s \leftarrow$  Sustitución hacia atrás( $\mathbf{A}, b$ )

```

2.1.2. Factorización LU

El método de *Factorización LU*, por su parte, aprovecha el hecho de que bajo ciertas condiciones, una matriz \mathbf{A} puede factorizarse como el producto de otras dos, en la forma $\mathbf{A} = \mathbf{LU}$, donde \mathbf{L} es triangular inferior con unos en la diagonal, y \mathbf{U} es triangular superior. De esta manera, el sistema $\mathbf{Ax} = b$ puede reescribirse como $\mathbf{LUx} = b$, y luego ser resuelto en dos etapas sencillas: si llamamos $y = \mathbf{Ux}$, podemos resolver primero el sistema $\mathbf{Ly} = b$ (aplicando sustitución hacia adelante, pues \mathbf{L} es triangular inferior), y una vez conocido el valor de y , resolver $\mathbf{Ux} = y$ (aplicando esta vez sustitución hacia atrás), obteniendo así la solución del sistema original.

El pseudocódigo que se presenta como Algoritmo 4 muestra el procedimiento a seguir para encontrar la factorización LU de una matriz, asumiendo que dicha factorización existe. El mismo genera como salida una única matriz \mathbf{B} que contiene tanto los coeficientes no nulos de \mathbf{L} como los de \mathbf{U} ; esta es una optimización destinada a la implementación, con el objetivo de mejorar su eficiencia evitando el almacenamiento de información redundante.

Algoritmo 4: Factorización LU

Datos: $\mathbf{A} \in \mathbb{R}^{n \times n}$

Resultado: $\mathbf{B} \in \mathbb{R}^{n \times n}$, tal que los elementos ubicados en la diagonal y por encima de ella forman una matriz triangular superior \mathbf{U} , y los ubicados por debajo de la misma, completados con unos en la diagonal, forman una matriz triangular inferior \mathbf{L} , con $\mathbf{LU} = \mathbf{A}$

```

1 para  $k \leftarrow 1$  a  $n$  hacer
2   | para  $j \leftarrow k + 1$  a  $n$  hacer
3     |   |  $m \leftarrow \frac{a_{j,k}}{a_{k,k}}$ 
4     |   |  $b_{j,k} \leftarrow m$ 
5     |   | para  $\ell \leftarrow k + 1$  a  $n$  hacer
6     |     |   |  $b_{j,\ell} \leftarrow a_{j,\ell} - m \cdot a_{k,\ell}$ 
7     |   | fin
8   | fin
9 fin

```

En cuanto a las hipótesis que permiten asegurar la existencia de la factorización LU de una matriz, resultará de utilidad el siguiente resultado, cuya demostración puede encontrarse en [1, p. 403].

Proposición 1. Si una matriz $\mathbf{A} \in \mathbb{R}^{n \times n}$ es tal que el sistema $\mathbf{Ax} = b$ ($b \in \mathbb{R}^n$) puede ser resuelto aplicando Eliminación Gaussiana sin pivoteo, entonces \mathbf{A} puede ser factorizada de la forma $\mathbf{A} = \mathbf{LU}$, donde \mathbf{U} es una matriz triangular superior y \mathbf{L} es una matriz triangular inferior con unos en la diagonal.

2.2. Modelado del sistema

Para resolver el problema presentado en la Introducción, se considerará una sección horizontal del horno, como puede verse en la Figura 2. Se denominará $r_e \in \mathbb{R}_{>0}$ al radio exterior y $r_i \in \mathbb{R}_{>0}$ al radio interior de la pared del horno. Para cada punto, se considerarán sus coordenadas polares (r, θ) con origen en el centro del horno, llamando $T(r, \theta)$ a la temperatura en dicho punto. Se

considerará conocido el valor (constante) de la temperatura en el interior del horno,

$$T_i = T(r, \theta) \quad (\forall r \in [0, r_i], \theta \in [0, 2\pi])$$

como así también el correspondiente a los puntos de la pared externa,

$$T_e(\theta) = T(r_e, \theta)$$

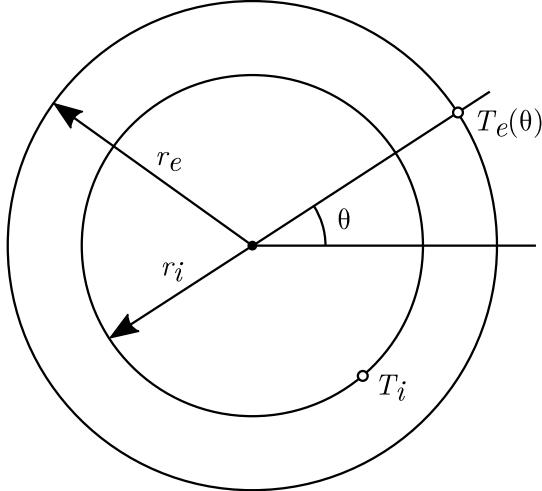


Figura 2: Sección horizontal del horno.

En el estado estacionario, la temperatura T satisface la ecuación del calor:

$$\frac{\partial^2 T(r, \theta)}{\partial r^2} + \frac{1}{r} \frac{\partial T(r, \theta)}{\partial r} + \frac{1}{r^2} \frac{\partial^2 T(r, \theta)}{\partial \theta^2} = 0 \quad (1)$$

No obstante, esta ecuación plantea un modelo continuo para estudiar el problema. Para poder tratar computacionalmente el mismo, se hace necesario discretizar su dominio, considerando una cantidad finita de puntos del mismo. Con este fin, se adoptará una partición del sector interno de la pared del horno, tal y como ilustra la Figura 3, en $m + 1$ radios equidistantes

$$r_i = r_0 < r_1 < \dots < r_m = r_e, \quad \text{con } r_j - r_{j-1} = \Delta r \text{ para } j = 1, \dots, m + 1$$

y en n ángulos iguales

$$0 = \theta_0 < \theta_1 < \dots < \theta_n = 2\pi, \quad \text{con } \theta_k - \theta_{k-1} = \Delta\theta \text{ para } k = 1, \dots, n$$

Nuestro objetivo es calcular el valor de la temperatura en todos los puntos de la discretización $t_{j,k} = T(r_j, \theta_k)$ para $j = 0, \dots, m$ y $k = 0, \dots, n - 1$. Gracias a los datos disponibles sobre las condiciones de borde, conocemos el valor de la temperatura en los puntos de las paredes externa e interna: $t_{m,k} = T_e(\theta_k)$ y $t_{0,k} = T_i$, para todo $k = 0, \dots, n - 1$.

Para discretizar el problema continuo planteado por la ecuación (1), y así obtener el valor en los puntos internos de la pared, pueden utilizarse las siguientes fórmulas de diferencias finitas

$$\begin{aligned} \frac{\partial^2 T(r, \theta)}{\partial r^2}(r_j, \theta_k) &\cong \frac{t_{j-1,k} - 2t_{jk} + t_{j+1,k}}{(\Delta r)^2} \\ \frac{\partial T(r, \theta)}{\partial r}(r_j, \theta_k) &\cong \frac{t_{j,k} - t_{j-1,k}}{\Delta r} \\ \frac{\partial^2 T(r, \theta)}{\partial \theta^2}(r_j, \theta_k) &\cong \frac{t_{j,k-1} - 2t_{jk} + t_{j,k+1}}{(\Delta\theta)^2} \end{aligned}$$

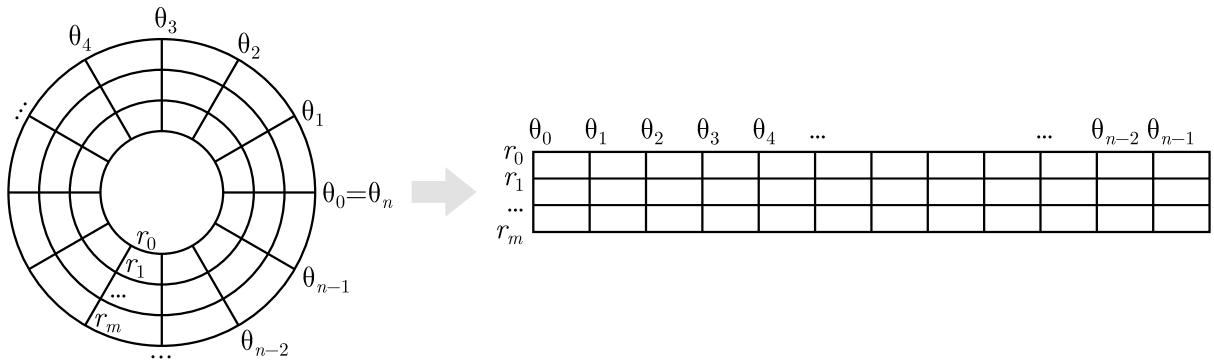


Figura 3: Discretización del dominio del problema.

que, reemplazadas en la ecuación (1), permiten obtener una formulación de la temperatura en estos puntos en función de sus cuatro puntos vecinos más próximos. Más específicamente, para $1 \leq j < m$, $0 \leq k < n$, se tiene que:

$$\begin{aligned} \Phi(j, k) = & \left(\frac{1}{(\Delta r)^2} - \frac{1}{r \Delta r} \right) t_{j-1, k} + \left(\frac{1}{r^2 (\Delta \theta)^2} \right) t_{j, k-1} + \\ & \left(-\frac{2}{(\Delta r)^2} + \frac{1}{r \Delta r} - \frac{2}{r^2 (\Delta \theta)^2} \right) t_{j, k} + \left(\frac{1}{r^2 (\Delta \theta)^2} \right) t_{j, k+1} + \left(\frac{1}{(\Delta r)^2} \right) t_{j+1, k} = 0 \quad (2) \end{aligned}$$

De esta manera, podemos reunir la información de la que disponemos sobre el problema discreto en un sistema de $(m+1)n$ ecuaciones con $(m+1)n$ incógnitas, cada una de ellas el valor de la temperatura en un punto de la discretización, es decir, $t_{j,k}$ para algún $j = 0, \dots, m$, $k = 0, \dots, n-1$. En otras palabras, el problema de calcular la temperatura en todos los puntos de la discretización se traduce en la resolución del siguiente sistema de ecuaciones lineales:

$$\begin{cases} t_{0,k} = T_i & \text{para } 0 \leq k < n \\ 0 = \Phi(j, k) & \text{para } 1 \leq j < m, 0 \leq k < n \\ t_{m,k} = T_e(\theta_k) & \text{para } 0 \leq k < n \end{cases} \quad (3)$$

Como puede verse, cada ecuación del sistema está asociada directamente al cálculo de la temperatura en uno de los puntos de la discretización, es decir, a una de las incógnitas.

A partir de ahora, se considerará la representación matricial del sistema, $\mathbf{A}x = b$, con $\mathbf{A} \in \mathbb{R}^{(m+1)n \times (m+1)n}$ y $b \in \mathbb{R}^{(m+1)n}$. Para construir la matriz \mathbf{A} , se ordenan tanto las incógnitas $t_{j,k}$ como las ecuaciones asociadas a ellas, primero de acuerdo al radio (j) y luego de acuerdo al ángulo (k). Esto significa que los coeficientes correspondientes a la incógnita $t_{j,k}$ se encontrarán en la columna $(m+1)j + k + 1$, y a la ecuación asociada al cálculo de esta incógnita le corresponderá la fila con el mismo índice.

Puede notarse que las primeras n filas y las últimas n filas de la matriz, correspondientes a los puntos de la pared interna y externa del horno, respectivamente, contendrán un 1 en la diagonal y 0 en todas las demás posiciones, dado que allí la temperatura ya es conocida y no depende de ningún otro punto. En las demás filas, correspondientes a los puntos internos de la pared del horno, la temperatura depende solamente del valor de la misma en los cuatro vecinos más próximos; esto provoca que los coeficientes no nulos se encuentren confinados a un entorno relativamente reducido de la diagonal de la matriz. Más precisamente, todos los coeficientes que

se encuentran fuera de una banda de ancho n hacia ambos lados de la diagonal son nulos, i.e. $a_{p,q} = 0$ siempre que $|p - q| > n$. En otras palabras, la matriz construida es *banda n, n* ; este hecho puede observarse claramente en la Figura 4.

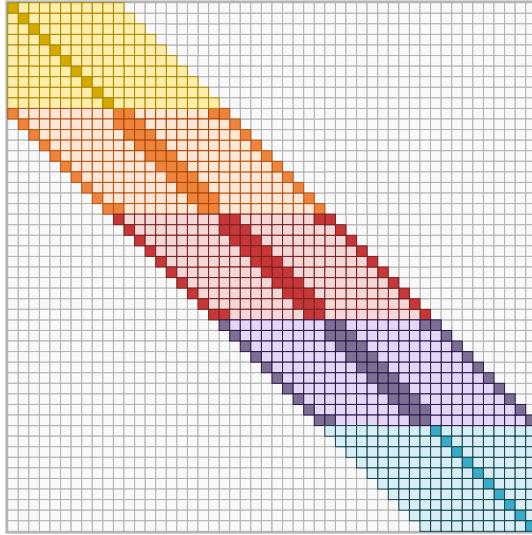


Figura 4: Matriz asociada al sistema con $m = 4$ y $n = 10$. Aparecen resaltados los coeficientes no nulos.

2.3. Aplicabilidad de los métodos elegidos

Los métodos numéricos ya presentados no son aplicables a la hora de resolver cualquier sistema de ecuaciones lineales, sino que requieren que este satisfaga determinadas hipótesis. Por este motivo, es necesario demostrar que estas se cumplen en el caso particular del sistema que modela el problema a resolver en este trabajo.

El objetivo es demostrar que tanto el método de Eliminación Gaussiana (sin pivoteo) como el de Factorización LU pueden aplicarse para resolver el sistema planteado en la sección anterior. Sin embargo, de lo enunciado en la Proposición 1 (sección 2.1) se desprende que basta con probar la aplicabilidad del primero de ellos. A continuación se procederá a introducir una definición y demostrar un lema, que luego se utilizarán durante la demostración.

Definición 1. Sea $\mathbf{A} \in \mathbb{R}^{n \times n}$. \mathbf{A} se dice *diagonal dominante* (por filas, de manera no estricta) si cada uno de los elementos de su diagonal es mayor o igual, en módulo, que la suma de los módulos de todos los demás coeficientes presentes en la misma fila. Es decir, si

$$|a_{j,j}| \geq \sum_{\substack{k=1 \\ k \neq j}}^n |a_{j,k}| \quad \text{para todo } j = 1, \dots, n$$

Lema 1. Sea $\mathbf{A}^{(0)} = \mathbf{A} \in \mathbb{R}^{n \times n}$ una matriz diagonal dominante, con $\mathbf{A}_{1,1} \neq 0$, y $\mathbf{A}^{(1)}$ el resultado de aplicar un paso de Eliminación Gaussiana (sin pivoteo) sobre \mathbf{A} . Entonces $\mathbf{A}^{(1)}$ es diagonal dominante.

Demostración. Se considerará la fila j -ésima de $\mathbf{A}^{(1)}$, para ver que $|a_{j,j}^{(1)}| \geq \sum_{k=1}^n |a_{j,k}^{(1)}|$. Se tiene

que

$$\left| a_{j,j}^{(1)} \right| = \left| a_{j,j} - \frac{a_{j,1}}{a_{1,1}} a_{1,j} \right| \quad \text{y} \quad \sum_{\substack{k=1 \\ k \neq j}}^n \left| a_{j,k}^{(1)} \right| = \sum_{\substack{k=2 \\ k \neq j}}^n \left| a_{j,k} - \frac{a_{j,1}}{a_{1,1}} a_{1,k} \right|$$

Luego,

$$\begin{aligned} \sum_{\substack{k=1 \\ k \neq j}}^n \left| a_{j,k}^{(1)} \right| &= \sum_{\substack{k=2 \\ k \neq j}}^n \left| a_{j,k} - \frac{a_{j,1}}{a_{1,1}} a_{1,k} \right| \\ &\leq \sum_{\substack{k=2 \\ k \neq j}}^n |a_{j,k}| + \left| \frac{a_{j,1}}{a_{1,1}} \right| \sum_{\substack{k=2 \\ k \neq j}}^n |a_{1,k}| \\ &\leq (|a_{j,j}| - |a_{j,1}|) + \left| \frac{a_{j,1}}{a_{1,1}} \right| (|a_{1,1}| - |a_{1,j}|) \\ &= |a_{j,j}| - \left| \frac{a_{j,1}}{a_{1,1}} \right| |a_{1,j}| \\ &\leq \left| a_{j,j} - \frac{a_{j,1}}{a_{1,1}} a_{1,j} \right| = \left| a_{j,j}^{(1)} \right| \end{aligned} \quad \square$$

Contando con estos resultados, se probará que el sistema construido en la sección anterior, tal y como fue expuesto en (3), puede ser resuelto aplicando Eliminación Gaussiana sin pivoteo. La demostración tendrá dos etapas:

1. Demostrar que la matriz $\mathbf{A} \in \mathbb{R}^{(m+1)n \times (m+1)n}$ asociada al sistema es diagonal dominante.
2. Demostrar, a partir de este hecho y en forma inductiva, que todas las iteraciones del algoritmo de Eliminación Gaussiana están bien definidas, sin utilizar pivoteo.

Para ver que la matriz \mathbf{A} es diagonal dominante, se verificará separadamente que la definición se cumple para las filas correspondientes a los bordes de la pared del horno (primeras n filas y últimas n filas) y para las correspondientes a los puntos internos.

- (I) Para $p = 1, \dots, n$ y para $p = mn + 1, \dots, (m+1)n$, la fila p -ésima corresponde a un punto del borde de la pared del horno. Como ya se señaló anteriormente, en estas filas hay un 1 en los lugares correspondientes a la diagonal, mientras que los demás coeficientes son nulos. Por lo tanto, se sigue trivialmente que, para $p = 1, \dots, n, mn + 1, (m+1)n$:

$$|a_{p,p}| = 1 \geq 0 = \sum_{\substack{q=1 \\ q \neq p}}^{(m+1)n} |a_{p,q}|$$

- (II) Para $p = n+1, \dots, mn$, la fila p -ésima corresponde a un punto interno de la pared. Es decir, representa a una ecuación del tipo $\Phi(j, k) = 0$, con $j = 1, \dots, m-1$ y $k = 0, \dots, n-1$. Luego, esta fila contiene solo 5 coeficientes no nulos.

El coeficiente que aparece en la diagonal es el correspondiente a la variable $t_{j,k}$. Entonces:

$$\begin{aligned}|a_{p,p}| &= \left| -\frac{2}{(\Delta r)^2} + \frac{1}{r \Delta r} - \frac{2}{r^2 (\Delta \theta)^2} \right| \\&= \left| -\frac{1}{(\Delta r)^2} + \frac{1}{r \Delta r} - \frac{2}{r^2 (\Delta \theta)^2} - \frac{1}{(\Delta r)^2} \right| \\&= \left| -\frac{r - \Delta r}{r (\Delta r)^2} - \frac{2}{r^2 (\Delta \theta)^2} - \frac{1}{(\Delta r)^2} \right| \\&= \left| \frac{r - \Delta r}{r (\Delta r)^2} + \frac{2}{r^2 (\Delta \theta)^2} + \frac{1}{(\Delta r)^2} \right|\end{aligned}$$

Dado que $r_i, \Delta r > 0$ y $j \geq 1$, se cumple que

$$r - \Delta r = (r_i + j \Delta r) - \Delta r = r_i + (j - 1) \Delta r > 0$$

de donde se desprende que todos los términos que aparecen dentro del módulo son positivos, y por lo tanto

$$|a_{p,p}| = \frac{r - \Delta r}{r (\Delta r)^2} + \frac{2}{r^2 (\Delta \theta)^2} + \frac{1}{(\Delta r)^2}$$

Por otro lado, si se suman los módulos de los demás coeficientes de la fila, se obtiene:

$$\begin{aligned}\sum_{\substack{q=1 \\ q \neq p}}^{(m+1)n} |a_{p,q}| &= \left| \frac{1}{(\Delta r)^2} - \frac{1}{r \Delta r} \right| + 2 \left| \frac{1}{r^2 (\Delta \theta)^2} \right| + \left| \frac{1}{(\Delta r)^2} \right| \\&= \left| \frac{r - \Delta r}{r (\Delta r)^2} \right| + \frac{2}{r^2 (\Delta \theta)^2} + \frac{1}{(\Delta r)^2} \\&= \frac{r - \Delta r}{r (\Delta r)^2} + \frac{2}{r^2 (\Delta \theta)^2} + \frac{1}{(\Delta r)^2} = |a_{p,p}|\end{aligned}$$

De todo lo anterior, se sigue que \mathbf{A} es diagonal dominante de forma no estricta, como se quería demostrar.

Observación 1. De la demostración anterior se desprende que para todo $p = 1, \dots, (m+1)n$, $|a_{p,p}| > 0$; es decir, la diagonal de \mathbf{A} no contiene ceros.

Por último, se debe probar que las $(m+1)n$ iteraciones del algoritmo de Eliminación Gaussiana necesarias para resolver el sistema pueden aplicarse, y que no es necesario utilizar pivoteo. La idea será demostrar que, para todo $u \leq (m+1)n$, pueden aplicarse u pasos de Eliminación Gaussiana sobre \mathbf{A} sin pivoteo, y que además, la matriz $\mathbf{A}^{(u)}$ que resulta de este proceso es diagonal dominante, y su fila $u+1$ es no nula. La demostración será por inducción en u .

- (I) **Caso base.** Como $a_{1,1} = 1$ no es nulo, se puede aplicar la primera iteración de Eliminación Gaussiana sin realizar pivoteo. Dado que \mathbf{A} es diagonal dominante, por el Lema 1, la matriz $\mathbf{A}^{(1)}$ obtenida también será diagonal dominante. Además, se sabe que $a_{1,2} = 0$, por lo que la primera iteración del algoritmo no afectará al coeficiente $a_{2,2}$, que era necesariamente no nulo (ver Observación 1). Luego, la segunda fila de $\mathbf{A}^{(1)}$ será no nula.

- (II) **Paso inductivo.** Por hipótesis inductiva, la matriz $\mathbf{A}^{(u-1)}$, obtenida tras aplicar $u-1$ pasos de Eliminación Gaussiana sobre \mathbf{A} , es diagonal dominante y su u -ésima fila es no nula. Esto implica que $a_{u,u}^{(u-1)} \neq 0$.

Nótese que es posible escribir a $\mathbf{A}^{(u-1)}$ por bloques de la siguiente manera:

$$\mathbf{A}^{(u-1)} = \begin{pmatrix} \mathbf{U}_{u-1} & \mathbf{B} \\ \mathbf{0} & \tilde{\mathbf{A}}_{u-1} \end{pmatrix}$$

donde las matrices $\mathbf{U}_{u-1} \in \mathbb{R}^{(u-1) \times (u-1)}$ y $\tilde{\mathbf{A}}_{u-1} \in \mathbb{R}^{((m+1)n-u+1) \times ((m+1)n-u+1)}$ son trivialmente diagonal dominantes.

Se puede ver que realizar el paso u -ésimo del algoritmo de Eliminación Gaussiana sobre \mathbf{A} equivale a realizar el primer paso de este algoritmo sobre $\tilde{\mathbf{A}}_{u-1}$, manteniendo intacto el resto de la matriz. Dado que $a_{u,u}^{(u-1)} \neq 0$, esto puede realizarse sin pivoteo, y del Lema 1 se sigue que el resultado de este proceso será diagonal dominante, lo cual implica directamente que también lo será $\mathbf{A}^{(u)}$.

Solo resta probar que la fila $u+1$ de la matriz $\mathbf{A}^{(u)}$ es no nula.

- (a) Si la fila $u+1$ corresponde a uno de los puntos externos de la pared del horno, es decir, si $1 \leq u+1 \leq n$ o $mn < u+1 \leq (m+1)n$, entonces tendrá un 1 en la diagonal y 0 en las demás posiciones, por lo que no se verá afectada por el algoritmo de Eliminación Gaussiana.
- (b) En caso contrario, la fila corresponde a una de los puntos internos de la pared, representando a una de las ecuaciones del tipo $\Phi(j, k) = 0$, para ciertos $j = 1, \dots, m-1$ y $k = 0, \dots, n-1$. Puntualmente, por la forma en la que se construye la matriz del sistema, el elemento $a_{u+1,u+1+n}$ es el coeficiente de la variable $t_{j,k+1}$ en dicha ecuación, que es necesariamente no nulo. Por otra parte, como la matriz \mathbf{A} es *banda* n, n , todos los elementos de la columna $u+1$ que se encuentran por encima de $a_{u+1,u+1+n}$ deben ser nulos. Esto quiere decir que ninguna de las iteraciones ya ejecutadas de Eliminación Gaussiana modificaron este coeficiente, y por lo tanto, sigue siendo no nulo.

2.4. Interpretación de los resultados

2.4.1. Estimación de la posición de la isotermia crítica

Además de contar con métodos para la resolución del sistema, la elaboración del trabajo requirió la elaboración de criterios para interpretar los resultados obtenidos. Uno de los aspectos a tener en cuenta fue la necesidad de aproximar el conjunto de puntos en el interior del horno que se encontraban a una temperatura dada, es decir, la posición de la isotermia correspondiente.

Como criterio general, se decidió evaluar independientemente los valores encontrados para cada uno de los ángulos de la discretización, procurando estimar, para cada uno de estos radios, el radio del punto cuya temperatura fuera la buscada. Más formalmente, dada una temperatura c , se buscó calcular un vector $\iota = (\iota_0, \dots, \iota_{n-1})$, donde cada ι_k es tal que $T(i_k, \theta_k) = c$.

Un análisis sencillo de las funciones involucradas permitió arribar a la conclusión de que, a lo largo de cada ángulo, la temperatura se comporta de forma monótona, por lo que en general el valor ι_k está bien definido; más aún teniendo en cuenta que, dado el contexto de aplicación,

cabe esperar temperaturas muy superiores en el interior que en el exterior del horno. En el caso de que la temperatura se mantuviera constante a lo largo del ángulo, obteniendo más de un punto válido para la posición de la isoterma, se decidió que se elegiría el más externo de ellos, para no afectar así las estimaciones posteriores acerca del riesgo de la estructura, que dependen de la cercanía de la isoterma a la pared.

Se evaluaron tres técnicas diferentes para la estimación de los valores ι_k , que se enumeran a continuación:

1. Para cada ángulo θ_k , elegir entre los radios de la discretización aquel cuya temperatura sea más próxima a la temperatura crítica c .

Este criterio es sumamente sencillo, pero se ve limitado al tomar en cuenta únicamente puntos de la discretización; esto podría causar que se viera seriamente afectado si la misma tuviera una baja granularidad en cuanto a cantidad de radios considerados. Peor aún, la isoterma real podría estar más cerca de la pared externa del horno que la estimada con este método, evitando, especialmente en conjunción con el problema recién mencionado, que se detecten situaciones de riesgo.

2. Para cada ángulo θ_k , determinar el más externo de los radios de la discretización cuya temperatura sea mayor o igual que la temperatura crítica c , y elegir luego como ι_k el radio de la discretización inmediatamente exterior. Esto equivale a seleccionar el más interno de los radios de la discretización tal que su temperatura y las de todos los radios más externos sea menor a c , es decir

$$\iota_k = \min \{r_j \mid 0 \leq j \leq m \wedge t_{j',k} < c (\forall j' = j, \dots, m)\}$$

Al igual que el anterior, este método adolece del problema de solo tener en cuenta puntos de la discretización. La diferencia es que, si se asume la hipótesis ya discutida de que la temperatura será decreciente hacia el exterior del horno, esta técnica puede asegurar que la isoterma nunca se encontrará más cerca de la pared externa que el resultado arrojado, garantizando que se detectarán todas las posibles situaciones de riesgo. No obstante, si la granularidad es baja y el método resulta ser sensible a la misma, podrían generarse falsos positivos, dando lugar a concluir erróneamente que la estructura corre peligro.

3. Para cada ángulo θ_k , considerar r_j , el más externo de los radios de la discretización con temperatura mayor o igual a c . Luego, sabiendo que la isoterma buscada se encontrará entre este radio y el inmediatamente exterior, aproximar la variación de la temperatura en los puntos intermedios mediante una función lineal y utilizar esta aproximación para estimar la posición de la isoterma, tomando

$$\iota_k = r_j + \Delta r \left(\frac{c - t_{j,k}}{t_{j+1,k} - t_{j,k}} \right)$$

Esta opción tiene en cuenta el hecho de que la temperatura es una función continua, y que en un estado estacionario tiende a estabilizarse y no a presentar saltos bruscos, por lo que se puede esperar que una función lineal resulte una aproximación aceptable de su comportamiento. A diferencia de las otras dos técnicas planteadas, esta contempla la posibilidad de que la isoterma se encuentre a mitad de camino entre dos radios de la discretización, y se considera esperable que arroje resultados razonables incluso con

menores granularidades. Esto le permite detectar situaciones de riesgo que con el primer método pasarían desapercibidas, pero evitando la generación de falsos positivos que podría provocar el segundo método. Por estas razones, esta fue la técnica elegida para realizar las estimaciones, por considerarla superior a las otras dos propuestas.

2.4.2. Cálculo del índice de peligrosidad

A la hora de utilizar el resultado del cálculo de la posición de la isoterma para determinar el riesgo de colapso de la estructura, se decidió adoptar como parámetro la mínima distancia entre la isoterma y la pared externa del horno. Se favoreció este criterio, por encima de otras opciones, como el cálculo de una distancia promedio, por considerar la posibilidad de que un pico de temperatura en un sector puntual de la pared sea suficiente para generar una situación de peligro, que podría pasar desapercibida en el caso de utilizar un índice más ponderado.

El índice de peligrosidad que se calcula es un valor normalizado con respecto al radio externo del horno, que se obtiene mediante la fórmula

$$\text{IP} = \frac{\hat{r}}{r_e} \quad \text{donde } \hat{r} = \max_{k=1, \dots, n-1} \{r_k\}$$

2.5. Implementación

Los dos métodos utilizados fueron implementados en lenguaje C++. Para la representación interna de las matrices se utiliza el tipo `vector`, proporcionado por la biblioteca estándar de C++. Se utilizó el mecanismo de clases del lenguaje para mejorar la organización del código, proporcionando métodos que implementan los diversos algoritmos ya presentados: sustitución hacia adelante y hacia atrás, Eliminación Gaussiana y Factorización LU.

Adicionalmente, se incorporaron a la implementación funciones para estimar, en función de la salida del algoritmo, la posición de la isoterma según las tres técnicas ideadas, como así también para calcular el índice de peligrosidad.

Un aspecto que se presentó durante la etapa de implementación fue la posibilidad de optimizar su rendimiento aprovechando la información disponible sobre el sistema a resolver. Puntualmente, el hecho de que la matriz con la que se trabaja es *banda* permite mejorar tanto la complejidad temporal como espacial del programa, evitando almacenar y procesar una gran cantidad de coeficientes de la matriz cuando ya se sabe que estos son nulos. No obstante, se decidió omitir esta posible optimización, ya que se contaba con un tiempo limitado y se consideró preferible priorizar la realización de pruebas experimentales.

3. Experimentación y discusión

A continuación, se presentan las pruebas experimentales que se realizaron, junto con los resultados obtenidos y una discusión de los mismos. Estas pueden agruparse en dos grandes grupos: por un lado, aquellas llevadas a cabo con el fin de evaluar y comparar el rendimiento de los diferentes métodos ya presentados; por otra parte, las que se efectuaron para estudiar el comportamiento del sistema y los indicadores diseñados ante variaciones en los datos de entrada.

Todos los experimentos se ejecutaron en las computadoras del laboratorio 4 del Departamento de Computación (FCEN - UBA).

3.1. Pruebas sobre el rendimiento de los métodos

3.1.1. Experimento 1: Tiempo de ejecución según granularidad de la discretización

Presentación

El objetivo de este experimento fue observar la relación entre los tiempos de ejecución y la granularidad de la discretización del dominio del problema, para los dos métodos implementados, manteniendo constantes los demás datos de entrada. El experimento se realizó dos veces, uno para cada uno de los métodos numéricos. Manteniendo las temperaturas constantes, variamos cantidad de particiones de los ángulos dejando la de los radios intacta. Calculamos la solución del sistema para cada una de las distintas granularidades y comparamos los tiempos que toman las diferentes ejecuciones. Repetimos esto mismo, ahora variando las particiones de los radios. Luego tomamos el otro método numérico y volvemos a iniciar el experimento.

Datos de entrada

Para ambos casos, se construyeron escenarios de prueba con los siguientes parámetros: $r_i = 30$, $r_e = 60$, $T_i = 1500$, $c = 500$. La temperatura externa se consideró constante, con $T_e(\theta) = 50$ para todo θ . En ambos casos las pruebas se realizaron utilizando primero el método de Eliminación Gaussiana y luego el de Factorización LU, y los tiempos de ejecución se tomaron en ciclos de reloj, utilizando las funciones provistas por la biblioteca estándar de C++. Para las pruebas se omitió realizar la estimación de la posición de la isoterma, limitándose a resolver el sistema por el método elegido ya que el cálculo de la isoterma es idéntico en ambos algoritmos y no afectará al análisis del tiempo de ejecución.

- **Caso A:** Se realizaron pruebas variando la cantidad de ángulos de la discretización, con $m + 1 = 30$ y $n = 10, 30, 50, 70, 90$. Se puede encontrar en los archivos adjuntos llamados `exp1a-{n}.in` la instancia pasada por parámetro.
- **Caso B:** Se realizaron pruebas variando la cantidad de radios de la discretización, con $n = 60$ y $m + 1 = 10, 30, 50, 70$. Se puede encontrar en los archivos adjuntos llamados `exp1b-{m+1}.in` la instancia pasada por parámetro.

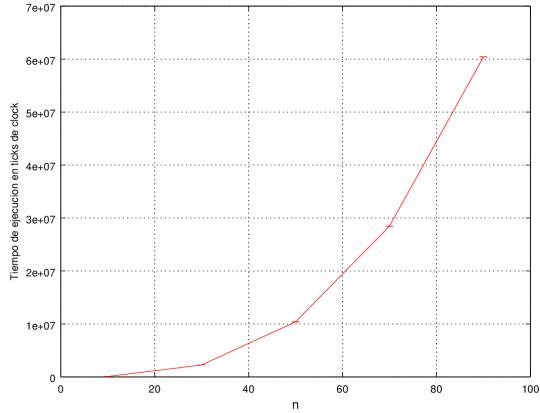
Hipótesis

La complejidad del algoritmo es cúbico en el tamaño de la entrada. Como solo variamos la granularidad del radio o del ángulo (dejando el otro constante) entonces en ambos casos

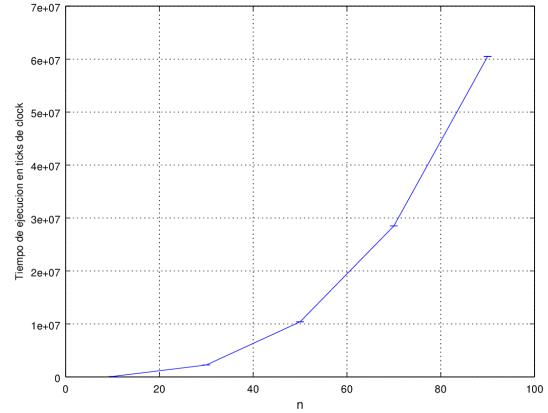
suponemos que el tiempo de ejecución va a ser cubico con relación a la granularidad de la entrada.

Resultados

Caso A

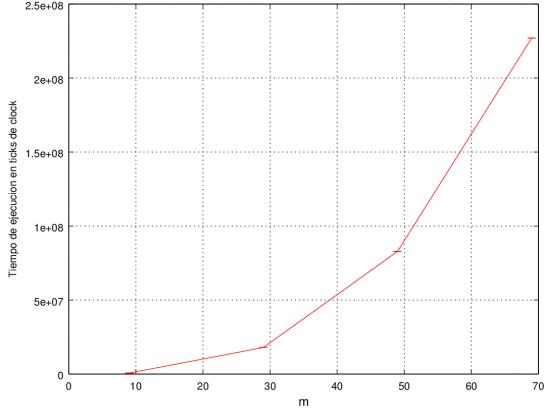


Eliminación Gaussiana

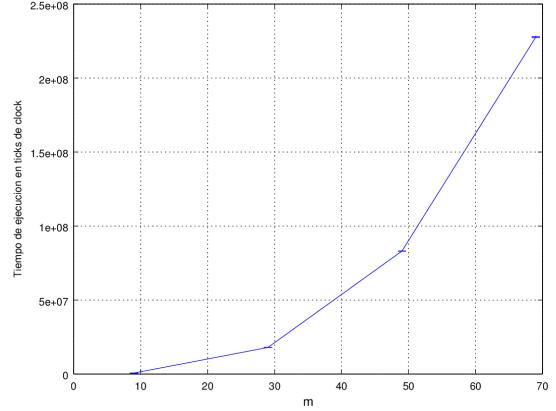


Factorización LU

Caso B



Eliminación Gaussiana



Factorización LU

Resultados arrojados por el experimento 1

Discusión

En todos los gráficos correspondientes a este experimento, es claro el crecimiento del tiempo de ejecución a medida que la granularidad aumenta, tanto en los ángulos como en los radios. Esto se debe a la complejidad de los algoritmos utilizados. Además, se puede observar que modificar la granularidad de radios o de ángulos es indistinto.

También, es importante notar como el tiempo de ejecución sobre una sola instancia no cambia demasiado al modificar el método de resolución. Sabemos que, cuando se trata de un solo caso, factorización LU tiene complejidad mayor por constantes que eliminación gaussiana, lo que se puede visualizar en los casos con mayor granularidad de la discretización.

3.1.2. Experimento 2: Tiempo según número de instancias

Presentación

En este experimento se generan distintas instancias para un mismo sistema, con el objetivo de comparar el tiempo de ejecución al resolverlas mediante el método de eliminación gaussiana y la factorización LU.

Para ello, se crean sistemas con diferentes cantidades de instancias y se comparan los tiempos de ejecución de los mismos para cada uno de los métodos. Con el fin de acercarse a los valores reales y descartar posibles falsos resultados, se ejecuta la resolución de un mismo sistema siete veces y se calcula el promedio de los tiempos medidos.

Datos

Se consideró una serie de 6 instancias del problema con los siguientes parámetros: $r_i = 10$, $r_e = 60$, $m + 1 = 25$, $n = 40$, $T_i = 1500$, $c = 500$. Para las temperaturas externas se tomaron valores arbitrarios entre 50 y 200°C, diferentes en cada una de las instancias. Se puede encontrar en el archivo adjunto llamados `exp2-{nInst}.in` las instancias pasadas por parámetro.

Se procesaron sucesivamente 1, 2, 3, 4, 5 y 6 instancias en corridas únicas del programa, utilizando primero el método de Eliminación Gaussiana y luego el de Factorización LU. Los tiempos de ejecución se tomaron en segundos, utilizando las funciones provistas por la biblioteca estándar de C++. Para las pruebas se omitió realizar la estimación de la posición de la isoterma, limitándose a resolver el sistema por el método elegido ya que el cálculo de la isoterma es idéntico en ambos algoritmos y no afectará al análisis del tiempo de ejecución.

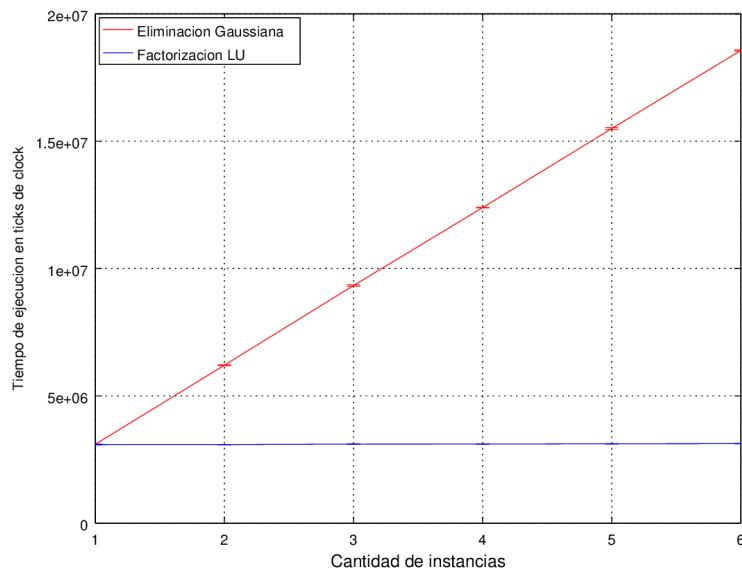
Hipótesis

Conjeturamos que la diferencia entre los tiempos de ejecución de los dos métodos aumenta con la cantidad de instancias.

Al considerar un mismo sistema con diferentes instancias el método de eliminación gaussiana resulta más lento ya que resuelve cada una de ellas individualmente. En cambio, el método de factorización LU reutiliza la matriz L ya calculada para resolver cada una de las instancias.

Destacamos que al considerar pocas instancias, factorización LU es más lenta que eliminación gaussiana, por una constante. Esto sucede ya que no se puede sacar ventaja de guardar la matriz L .

Resultados



Resultados arrojados por el experimento 2. Tiempo de ejecución en función del número de instancias del problema.

Discusión

Se observa que cuando hay pocas instancias, buscar la isoterma con el método de eliminación gaussiana o el de factorización LU es muy similar.

A medida que aumentan las instancias las diferencias entre ambos métodos aumentan, ya que la factorización LU se mantiene casi inmutable al aumentar las instancias mientras que la eliminación gaussiana aumenta notoriamente.

Se puede observar que lo que tarda la eliminación gaussiana cuando tiene 2 instancias es aproximadamente el doble de lo que tarda cuando es una única instancia. Cuando son 3 instancias, tarda el triple de lo que tarda cuando es una única instancia.

Sean n instancias entonces lo que tarda la eliminación gaussiana es lo que tarda cuando es una única instancia n veces. Esto es así ya que en este método numérico, para cada una de las instancias recalcula toda la matriz. En cambio la factorización LU, calcula la matriz una única vez.

3.2. Pruebas sobre el comportamiento del sistema

3.2.1. Experimento 3: Isoterma según temperatura en pared externa

Presentación

En este experimento vamos a observar como varía la isoterma cuando modificamos la temperatura de la pared externa del horno. Para ello tomamos dos sistemas con temperaturas externas constantes pero distintos entre sí. Graficamos para cada uno de ellos la isoterma 500.

Datos

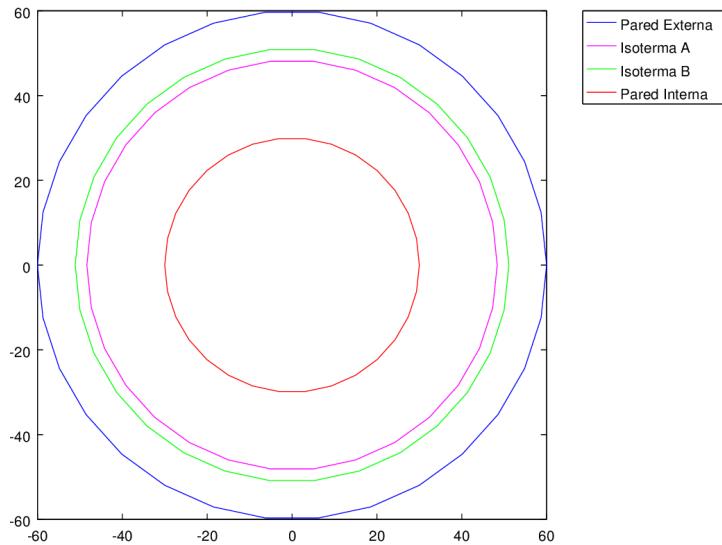
Se evaluaron dos escenarios de prueba, ambos con los siguientes parámetros: $r_i = 30$, $r_e = 60$, $m+1 = 30$, $n = 30$, $T_i = 1500$, $c = 500$. Se utilizaron temperaturas externas constantes en todos los puntos (r_e, θ) de la discretización, con $T_e(\theta) = 50^\circ\text{C}$ en uno de los casos y $T_e(\theta) = 200^\circ\text{C}$ en el otro. Se puede encontrar en los archivos adjuntos llamados `exp3-{Te(θ)}.in` la instancia pasada por parámetro.

Hipótesis

Suponemos que cuando mayor sea la temperatura, la isoterma estará mas cerca de la pared externa del alto horno.

Resultados

En el gráfico puede observarse la ubicación estimada de la isoterma para cada uno de los casos, en magenta para $T_e(\theta) = 50^\circ\text{C}$ (Isoterma A) y en verde para $T_e(\theta) = 200^\circ\text{C}$ (Isoterma B).



Resultados arrojados por el experimento 3. Posiciones de las isothermas de 500°C.

Discusión

En este experimento se puede ver que la isoterma magenta se encuentra más próxima a la pared interna del horno ya que corresponde a la instancia que tiene menor temperatura en la pared externa (50°C). La isoterma verde se encuentra más cerca de la pared externa debido a que las temperaturas son más altas (200°C).

En uno de los casos la temperatura externa es la máxima posible y en el otro la mínima. Podemos observar que nunca, con esta discretización, para ningún valor de temperaturas externas válidas, la isoterma puede pasar más cerca de la pared externa del alto horno que lo que pasa la

isoterma marcada en el gráfico con verde. Tampoco es posible que la isoterma pase más cerca de la pared interna del horno de lo que pasa la isoterma marcada en el gráfico con color magenta.

3.2.2. Experimento 4: Isoterma según granularidad

Presentación

Para realizar este experimento, se calcula la isoterma en un sistema donde la temperatura de la pared externa es constante, y la granularidad es muy alta. Llamaremos a esta, la isoterma de contraste. Se planea observar como varía la isoterma cuando se disminuye esta granularidad modificando únicamente la cantidad de radios a tener en cuenta (sin cambiar los ángulos), o modificando la cantidad de ángulos (sin cambiar los radios). Luego, se repite este proceso pero variando los valores de las temperaturas de la pared externa del horno. Estos valores son generados utilizando la función seno.

Datos

(I) Temperaturas constantes

Para ambos casos se consideraron instancias de prueba con los siguientes parámetros: $r_i = 30$, $r_e = 60$, $T_i = 1500$, $c = 500$. Se utilizó una temperatura externa constante en todos los puntos (r_e, θ) de la discretización, ($T_e(\theta) = 50$).

En primer lugar, se calculó la solución del sistema y la posición estimada de la isoterma para una discretización considerablemente granular, con $m+1 = 70$ y $n = 90$, para utilizar como caso de contraste.

- **Caso A:** Se mantuvo constante la cantidad de radios de la discretización $m + 1 = 70$, y se tomaron instancias con diferentes cantidades de ángulos, para $n = 3, 5, 8, 10, 30, 50, 70$. Los gráficos que incluimos representan los resultados obtenidos para $n = 5, 10, 50$, reflejando las temperaturas calculadas y la ubicación estimada de la isoterma (en verde), comparada con la obtenida para el caso de contraste (en magenta). Se puede encontrar en los archivos adjuntos llamados `exp4-const-ang-{n}.in` la instancia pasada por parámetro.
- **Caso B:** Se mantuvo constante la cantidad de ángulos de la discretización $n = 90$, y se tomaron instancias con diferentes cantidades de radios, para $m+1 = 3, 5, 8, 10, 30, 50$. Los gráficos que incluimos representan los resultados obtenidos para $m+1 = 3, 8, 30$, reflejando las temperaturas calculadas y la ubicación estimada de la isoterma (en verde), comparada con la obtenida para el caso de contraste (en magenta). Se puede encontrar en los archivos adjuntos llamados `exp4-const-rad-{m+1}.in` la instancia pasada por parámetro.

(II) Temperaturas variables

Para ambos casos se consideraron instancias de prueba con los siguientes parámetros: $r_i = 30$, $r_e = 60$, $T_i = 1500$, $c = 500$. Se generó una instancia de prueba con una temperatura externa variable entre 50 y 200°C en todos los puntos (r_e, θ) de la discretización. Para la generación de las instancias de prueba se consideró la función $T_e(\theta) = 125 + 75 \sin(2\theta)$ para que la variación de temperaturas sea suave. En primer lugar, se calculó la solución del

sistema y la posición estimada de la isoterma para una discretización considerablemente granular, con $m + 1 = 70$ y $n = 90$, para utilizar como caso de contraste. Reproducimos los gráficos que representan las temperaturas calculadas para todos los puntos de la discretización y la ubicación estimada de la isoterma.

- **Caso A:** Se mantuvo constante la cantidad de radios de la discretización $m + 1 = 70$, y se tomaron instancias con diferentes cantidades de ángulos, para $n = 3, 5, 8, 10, 30, 50, 70$. Los gráficos que incluimos representan los resultados obtenidos para $n = 5, 10, 50$, reflejando las temperaturas calculadas y la ubicación estimada de la isoterma (en verde), comparada con la obtenida para el caso de contraste (en magenta). Se puede encontrar en los archivos adjuntos llamados `exp4-seno-ang-{n}.in` la instancia pasada por parámetro.
- **Caso B:** Se mantuvo constante la cantidad de ángulos de la discretización $n = 90$, y se tomaron instancias con diferentes cantidades de radios, para $m + 1 = 3, 5, 8, 10, 30, 50$. Los gráficos que incluimos representan los resultados obtenidos para $m + 1 = 3, 8, 30$, reflejando las temperaturas calculadas y la ubicación estimada de la isoterma (en verde), comparada con la obtenida para el caso de contraste (en magenta). Se puede encontrar en los archivos adjuntos llamados `exp4-seno-rad-{m+1}.in` la instancia pasada por parámetro.

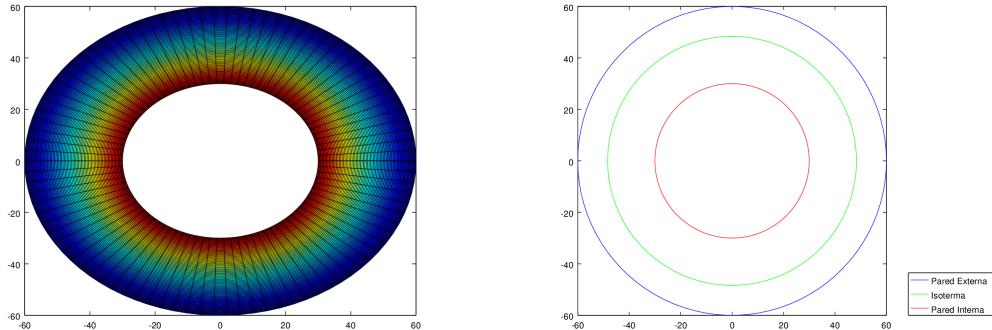
Hipótesis

Se espera que, en ambos casos, cuanto menor sea la granularidad de la discretización, más alejada estará la isoterma calculada de la de contraste. Si tomamos menos particiones sobre radios, la distancia entre ellos aumenta. Como el cálculo de la isoterma supone que la temperatura entre dos de ellos, uno más lejano y otro más cercano a la pared, decrece en forma lineal, lo cual podría generar que el error aumente. Si se toman menos particiones sobre ángulos, existen puntos que no van a ser considerados en el cálculo de la isoterma, por lo tanto, el error también es mayor.

Resultados

Para estos gráficos se tomará el color rojo como las temperaturas más calientes siendo el rojo más oscuro la temperatura de 1500°C y azul las temperaturas más frías siendo el azul más oscuro 50°C.

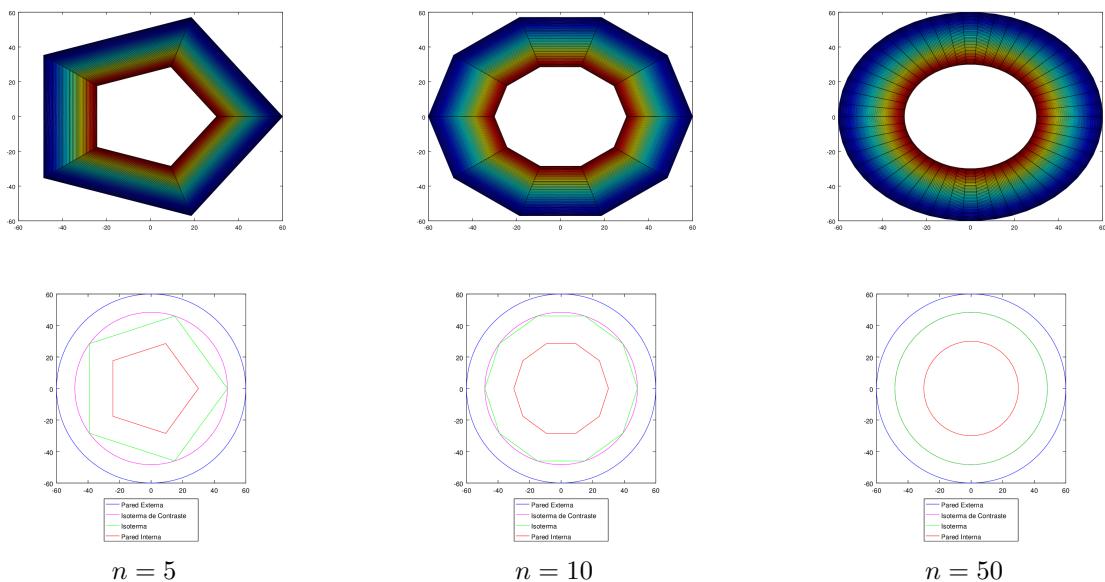
(i) Temperaturas constantes

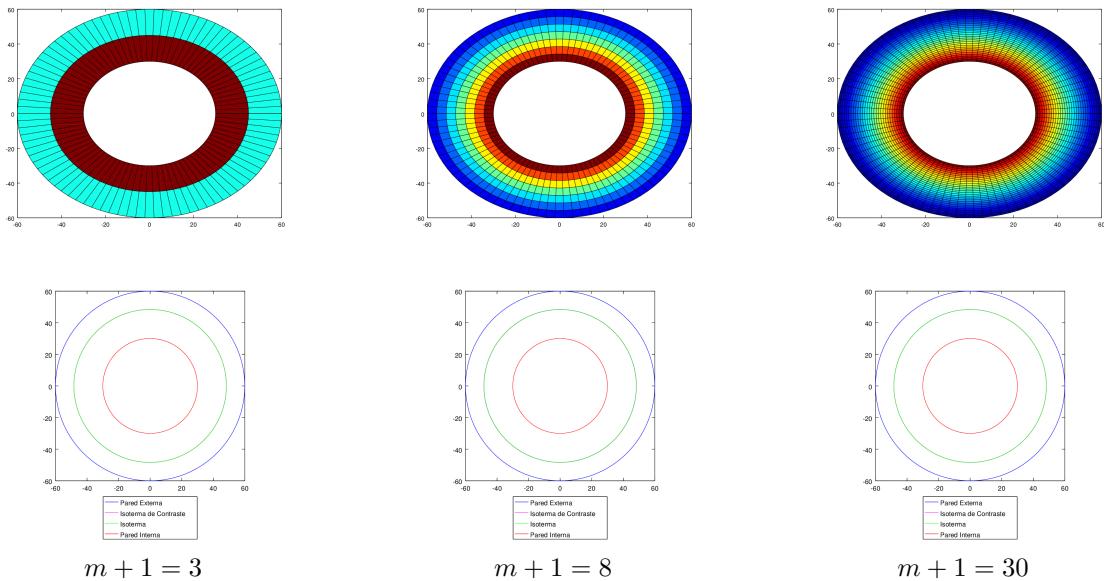
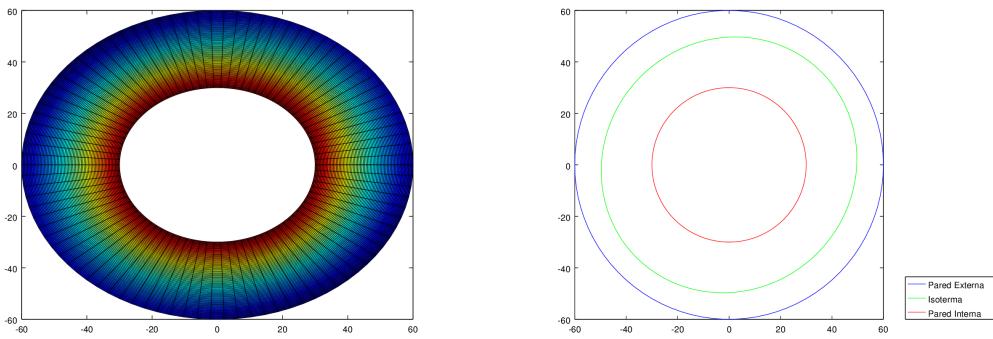
Caso de contraste ($m + 1 = 70$ y $n = 90$)

Temperaturas obtenidas

Posición estimada de la isoterma 500°C

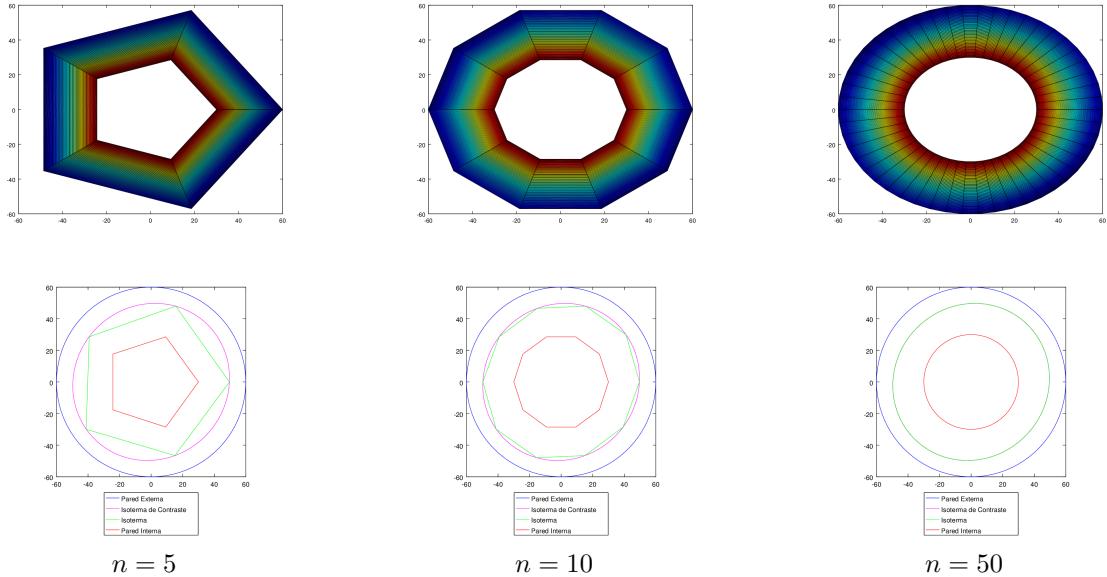
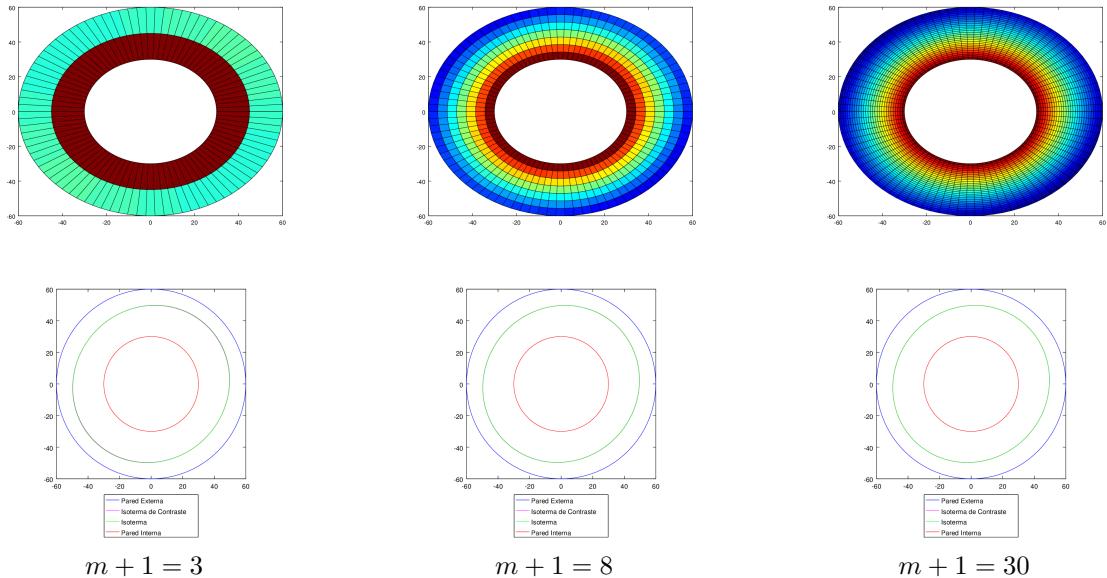
Caso A

 $n = 5$ $n = 10$ $n = 50$

Caso B**(II) Temperaturas variables****Caso de contraste ($m+1 = 70$ y $n = 90$)**

Temperaturas obtenidas

Posición estimada de la isotermia 500°C

Caso A**Caso B****Discusión**

Como se puede observar en los resultados, variar la granularidad con respecto a los ángulos no afecta al resultado de la posición de la isoterma. Ésta sigue siendo igual que la de contraste. Cuanto mayor cantidad de ángulos se tengan, más información podrá verse pero sin afectar el resultado total. En cambio, cuando modificamos la granularidad con respecto a los radios la isoterma calculada se aleja de la de contraste.

Se deja abierta para experimentos futuros la posibilidad de observar qué tanto se modifica dependiendo de las temperaturas de la pared externa del horno. Para ello se podrían tomar

valores de temperaturas externas mayores que 200°C y menores que 50°C a modo de análisis.

3.2.3. Experimento 5: Índice de peligrosidad según ancho de la pared

Presentación

Se busca estudiar la variación de la medida de peligrosidad con respecto al grosor de la pared del alto horno, sin modificar las temperaturas de la pared externa. Para ello se modificará el radio interno dejando constante el externo. Se realizará de esta manera ya que para lo que se quiere observar es indistinto cual de los dos variar.

Datos

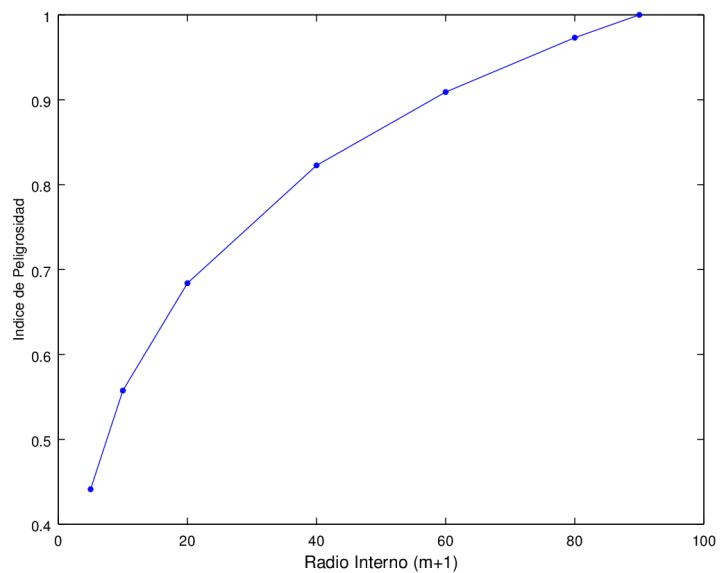
Para este experimento se mantuvieron constantes los parámetros, $r_e = 90$, $m + 1 = 30$, $n = 80$, $T_i = 1500$, $c = 500$, como así también la temperatura externa $T_e(\theta)$, para la cual se tomaron valores arbitrarios entre 50°C y 200°C. Se generaron instancias con los siguientes valores de r_i : 5, 10, 20, 40, 60, 80, 90. Se puede encontrar en los archivos adjuntos llamados `exp5-{r_i}.in` la instancia pasada por parámetro.

Hipótesis

La hipótesis de este experimento es que a medida que el grosor de la pared disminuye, el índice de peligrosidad aumenta, dado que al dejar las temperaturas constantes, la isoterma se encontrará cada vez más cerca de la pared externa. Esto sucede porque se acerca la pared interna (donde se encuentra la temperatura más alta del sistema) a la pared externa.

Resultados

En el gráfico pueden observarse los valores obtenidos para el índice de peligrosidad en función de los diferentes valores de r_i .



Resultados arrojados por el experimento 5. Índice de peligrosidad en función del valor de r_i .

Discusión

En este experimento se puede ver que a medida que disminuye el grosor de la pared del alto horno, la isoterma se encuentra cada vez mas cerca de la pared externa. Dado que el índice de peligrosidad depende precisamente de la proximidad de la isoterma a la pared externa, cuando disminuye el grosor de la pared, el índice de peligrosidad aumenta, verificando así lo que se planteó en la hipótesis.

4. Conclusiones

Además de exponer otra de las tantas posibilidades de aplicación a problemas físicos de la resolución de sistemas de ecuaciones lineales, la realización de este trabajo permitió, a partir de las pruebas experimentales llevadas a cabo, extraer las siguientes conclusiones con respecto a los dos aspectos del problema cubiertos por estas pruebas: el rendimiento de los métodos y el comportamiento del sistema.

Las comparaciones realizadas entre los métodos estudiados con respecto a su rendimiento temporal muestran que este fue muy similar entre ambos cuando solo se realizó el cálculo para una única instancia. Sin embargo, al realizar el cálculo repetidamente para varias instancias, manteniendo constantes los parámetros del problema pero variando los datos de entrada, el método de Factorización LU se mostró claramente superior al de Eliminación Gaussiana. Dado que este escenario podría presentarse en el caso de aplicación estudiado, por ejemplo, al analizar la evolución temporal del sistema, la utilización del método de Factorización LU podría ser conveniente.

Cabe destacar también que el aumento de la granularidad de la discretización repercute negativamente en el rendimiento de ambos métodos por igual. Esto indica que debe estudiarse con cuidado la granularidad necesaria, considerando las limitaciones particulares del contexto a la hora de realizar los cálculos.

En cuanto al comportamiento del sistema, se pudieron verificar las hipótesis de que la cercanía de la isoterma en el interior de la pared del horno es directamente proporcional a la temperatura externa e inversamente proporcional al grosor de la pared del horno. Esto arroja información que puede ser útil a la hora de prevenir situaciones de riesgo en el contexto de aplicación real. Por otra parte, se comprobó que la precisión de la estimación de la posición de la isoterma es más sensible a cambios en la granularidad de radios que de ángulos; no obstante, sería útil continuar experimentando para verificar el efecto que una menor granularidad de ángulos puede tener en el índice de peligrosidad y en la detección de situaciones de riesgo.

Apéndices

A. Enunciado del trabajo práctico

A.1. Introducción

Consideremos la sección horizontal de un horno de acero cilíndrico, como en la Figura 5. El sector A es la pared del horno, y el sector B es el horno propiamente dicho, en el cual se funde el acero a temperaturas elevadas. Tanto el borde externo como el borde interno de la pared forman círculos. Suponemos que la temperatura del acero dentro del horno (o sea, dentro de B) es constante e igual a 1500°C . Tenemos sensores ubicados en la parte externa del horno para medir la temperatura de la pared externa del mismo, que habitualmente se encuentra entre 50°C y 200°C . El problema que debemos resolver consiste en estimar la isoterma de 500°C dentro de la pared del horno, para estimar la resistencia de la misma. Si esta isoterma está demasiado cerca de la pared externa del horno, existe peligro de que la estructura externa de la pared colapse.

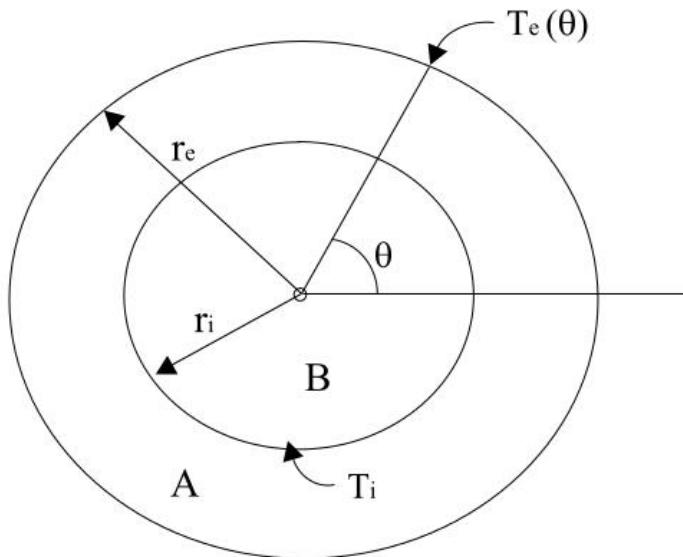


Figura 5: Sección circular del horno.

El objetivo del trabajo práctico es implementar un programa que calcule la isoterma solicitada, conociendo las dimensiones del horno y las mediciones de temperatura en la pared exterior.

A.2. El Modelo

Sea $r_e \in \mathbb{R}$ el radio exterior de la pared y sea $r_i \in \mathbb{R}$ el radio interior de la pared. Llamemos $T(r, \theta)$ a la temperatura en el punto dado por las coordenadas polares (r, θ) , siendo r el radio y θ el ángulo polar de dicho punto. En el estado estacionario, esta temperatura satisface la ecuación del calor:

$$\frac{\partial^2 T(r, \theta)}{\partial r^2} + \frac{1}{r} \frac{\partial T(r, \theta)}{\partial r} + \frac{1}{r^2} \frac{\partial^2 T(r, \theta)}{\partial \theta^2} = 0 \quad (4)$$

Si llamamos $T_i \in \mathbb{R}$ a la temperatura en el interior del horno (sector B) y $T_e : [0, 2\pi] \rightarrow \mathbb{R}$ a la función de temperatura en el borde exterior del horno (de modo tal que el punto (r_e, θ) tiene temperatura $T_e(\theta)$), entonces tenemos que

$$T(r, \theta) = T_i \text{ para todo punto } (r, \theta) \text{ con } r \leq r_i \quad (5)$$

$$T(r_e, \theta) = T_e(\theta) \text{ para todo punto } (r_e, \theta) \quad (6)$$

El problema en derivadas parciales dado por la primera ecuación con las condiciones de contorno presentadas recientemente, permite encontrar la función T de temperatura en el interior del horno (sector A), en función de los datos mencionados en esta sección.

Para resolver este problema computacionalmente, discretizamos el dominio del problema (el sector A) en coordenadas polares. Consideramos una partición $0 = \theta_0 < \theta_1 < \dots < \theta_n = 2\pi$ en n ángulos discretos con $\theta_k - \theta_{k-1} = \Delta\theta$ para $k = 1, \dots, n$, y una partición $r_i = r_0 < r_1 < \dots < r_m = r_e$ en $m + 1$ radios discretos con $r_j - r_{j-1} = \Delta r$ para $j = 1, \dots, m$.

El problema ahora consiste en determinar el valor de la función T en los puntos de la discretización (r_j, θ_k) que se encuentren dentro del sector A. Llamemos $t_{jk} = T(r_j, \theta_k)$ al valor (desconocido) de la función T en el punto (r_j, θ_k) .

Para encontrar estos valores, transformamos la ecuación (4) en un conjunto de ecuaciones lineales sobre las incógnitas t_{jk} , evaluando (4) en todos los puntos de la discretización que se encuentren dentro del sector A. Al hacer esta evaluación, aproximamos las derivadas parciales de T en (4) por medio de las siguientes fórmulas de diferencias finitas:

$$\frac{\partial^2 T(r, \theta)}{\partial r^2}(r_j, \theta_k) \cong \frac{t_{j-1,k} - 2t_{jk} + t_{j+1,k}}{(\Delta r)^2} \quad (7)$$

$$\frac{\partial T(r, \theta)}{\partial r}(r_j, \theta_k) \cong \frac{t_{j,k} - t_{j-1,k}}{\Delta r} \quad (8)$$

$$\frac{\partial^2 T(r, \theta)}{\partial \theta^2}(r_j, \theta_k) \cong \frac{t_{j,k-1} - 2t_{jk} + t_{j,k+1}}{(\Delta \theta)^2} \quad (9)$$

Es importante notar que los valores de las incógnitas son conocidos para los puntos que se encuentran sobre el borde exterior de la pared, y para los puntos que se encuentren dentro del sector B. Al realizar este procedimiento, obtenemos un sistema de ecuaciones lineales que modela el problema discretizado. La resolución de este sistema permite obtener una aproximación de los valores de la función T en los puntos de la discretización.

A.3. Enunciado

Se debe implementar un programa en C o C++ que tome como entrada los parámetros del problema $(r_i, r_e, m + 1, n, \text{valor de la isoterma buscada}, T_i, T_e(\theta))$ que calcule la temperatura dentro de la pared del horno utilizando el modelo propuesto en la sección anterior y que encuentre

la isoterma buscada en función del resultado obtenido del sistema de ecuaciones. El método para determinar la posición de la isoterma queda a libre elección de cada grupo y debe ser explicado en detalle en el informe.

El programa debe formular el sistema obtenido a partir de las ecuaciones (4)-(9) y considerar dos métodos posibles para su resolución: mediante el algoritmo clásico de Eliminación Gaussiana y la Factorización LU. Finalmente, el programa escribirá en un archivo la solución obtenida con el formato especificado en la siguiente sección.

Como ya se ha visto en la materia, no es posible aplicar los métodos propuestos para la resolución a cualquier sistema de ecuaciones. Sin embargo, la matriz del sistema considerado en el presente trabajo cumple con ser diagonal dominante (no estricto) y que, ordenando las variables y ecuaciones convenientemente, es posible armar un sistema de ecuaciones cuya matriz posee la propiedad de ser *banda*. Luego, se pide demostrar (o al menos dar un esquema de la demostración) el siguiente resultado e incluirlo en el informe:

Proposición 2. *Sea $A \in \mathbb{R}^{n \times n}$ la matriz obtenida para el sistema definido por (4)-(9). Demostrar que es posible aplicar Eliminación Gaussiana sin pivoteo.¹*

La solución del sistema de ecuaciones permitirá saber la temperatura en los puntos de la discretización. Sin embargo, nuestro interés es calcular la isoterma 500, para poder determinar si la estructura se encuentra en peligro. Luego, se pide lo siguiente:

- Dada la solución del sistema de ecuaciones, proponer una forma de estimar en cada ángulo de la discretización la posición de la isoterma 500.
- En función de la aproximación de la isoterma, proponer una forma (o medida) a utilizar para evaluar la peligrosidad de la estructura en función de la distancia a la pared externa del horno.

En función de la experimentación, se busca realizar dos estudios complementarios: por un lado, analizar cómo se comporta el sistema y, por otro, cuáles son los requerimientos computacionales de los métodos. Se pide como mínimo realizar los siguientes experimentos:

1. Comportamiento del sistema.
 - Considerar al menos dos instancias de prueba, generando distintas discretizaciones para cada una de ellas y comparando la ubicación de la isoterma buscada respecto de la pared externa del horno. Se sugiere presentar gráficos de temperatura o curvas de nivel para los mismos, ya sea utilizando las herramientas provistas por la cátedra o implementando sus propias herramientas de graficación.
 - Estudiar la proximidad de la isoterma buscada respecto de la pared exterior del horno en función de distintas granularidades de discretización y las condiciones de borde.
2. Evaluación de los métodos.
 - Analizar el tiempo de cómputo requerido para obtener la solución del sistema en función de la granularidad de la discretización. Se sugiere presentar los resultados mediante gráficos de tiempo de cómputo en función de alguna de las variables del problema.

¹Sugerencia: Notar que la matriz es diagonal dominante (no estrictamente) y analizar qué sucede al aplicar un paso de Eliminación Gaussiana con los elementos de una fila.

- Considerar un escenario similar al propuesto en el experimento 1. pero donde las condiciones de borde (i.e., T_i y $T_e(\theta)$) cambian en distintos instantes de tiempo. En este caso, buscamos obtener la secuencia de estados de la temperatura en la pared del horno, y la respectiva ubicación de la isoterma especificada. Para ello, se considera una secuencia de $ninst$ vectores con las condiciones de borde, y las temperaturas en cada estado es la solución del correspondiente sistema de ecuaciones. Se pide formular al menos un experimento de este tipo, aplicar los métodos de resolución propuestos de forma conveniente y compararlos en términos de tiempo total de cómputo requerido para distintos valores de $ninst$.

De manera opcional, aquellos grupos que quieran ir un poco más allá pueden considerar trabajar y desarrollar alguno(s) de los siguientes puntos extra:

- Notar que el sistema resultante tiene estructura *banda*. Proponer una estructura para aprovechar este hecho en términos de la *complejidad espacial* y como se adaptarían los algoritmos de Eliminación Gaussiana y Factorización LU para reducir la cantidad de operaciones a realizar.
- Implementar dicha estructura y las adaptaciones necesarias para el algoritmo de Eliminación Gaussiana.
- Implementar dicha estructura y las adaptaciones necesarias para el algoritmo de Factorización LU.

Finalmente, se deberá presentar un informe que incluya una descripción detallada de los métodos implementados y las decisiones tomadas, el método propuesto para el cálculo de la isoterma buscada y los experimentos realizados, junto con el correspondiente análisis y siguiendo las pautas definidas en el archivo `pautas.pdf`.

A.4. Programa y formato de archivos

Se deberán entregar los archivos fuentes que contengan la resolución del trabajo práctico. El ejecutable tomará tres parámetros por línea de comando, que serán el archivo de entrada, el archivo de salida, y el método a ejecutar (0 Eliminación Gaussiana, 1 LU).

El archivo de entrada tendrá la siguiente estructura:

- La primera línea contendrá los valores r_i , r_e , $m + 1$, n , iso , $ninst$, donde iso representa el valor de la isoterma buscada y $ninst$ es la cantidad de instancias del problema a resolver para los parámetros dados.
- A continuación, el archivo contendrá $ninst$ líneas, cada una de ellas con $2n$ valores, los primeros n indicando los valores de la temperatura en la pared interna, i.e., $T_i(\theta_0), T_i(\theta_1), \dots, T_i(\theta_{n-1})$, seguidos de n valores de la temperatura en la pared externa, i.e., $T_e(\theta_0), T_e(\theta_1), \dots, T_e(\theta_{n-1})$.

El archivo de salida obligatorio tendrá el vector solución del sistema reportando una componente del mismo por línea. En caso de $ninst > 1$, los vectores serán reportados uno debajo del otro.

Junto con el presente enunciado, se adjunta una serie de scripts hechos en `python` y un conjunto instancias de test que deberán ser utilizados para la compilación y un testeo básico de la implementación. Se recomienda leer el archivo `README.txt` con el detalle sobre su utilización.

A.5. Fechas de entrega

- *Formato Electrónico:* Jueves 3 de Septiembre de 2015, hasta las 23:59 hs, enviando el trabajo (informe + código) a la dirección `metnum.lab@gmail.com`. El subject del email debe comenzar con el texto `[TP1]` seguido de la lista de apellidos de los integrantes del grupo.
- *Formato físico:* Viernes 4 de Septiembre de 2015, de 17:30 a 18:00 hs.

Importante: El horario es estricto. Los correos recibidos después de la hora indicada serán considerados re-entrega. Los grupos deben ser de 3 o 4 personas, sin excepción. Es indispensable que los trabajos pasen satisfactoriamente los casos de test provistos por la cátedra.

Referencias

- [1] Richard L. Burden y J. Douglas Faires. *Numerical Analysis*. Ninth Edition. Boston, USA: Brooks/Cole Publishing Company, 2010.