

Método de las Diferencias Finitas

Ruben Weht^{1,2}

Mariano Forti^{1,3}

¹ Instituto de Tecnología Prof. Jorge Sabato

²Física del Sólido, Edificio TANDAR, weht@cnea.gov.ar, interno 7104

³División Aleaciones Especiales, Edificio 47 (microscopía), mforti@cnea.gov.ar, interno 7832

1. Problema de la chapa

Vamos a encarar el problema presentado en la Figura 1. se tiene una chapa dispuesta bajo condiciones de contorno tales que sus lados se encuentran a temperaturas fijas T_A , T_B , T_C , T_D .

Queremos obtener la distribución de temperatura dentro del recinto de la chapa, en condiciones estacionarias, es decir, cuando las temperaturas de no varían en ningún punto del recinto. Por lo tanto, podemos elegir el modelo físico a resolver. La ecuación diferencial que rige el problema es la ecuación de Poisson, que es la ecuación de transferencia térmica homogénea.

$$\frac{\partial^2 T_{(x,y)}}{\partial x^2} + \frac{\partial^2 T_{(x,y)}}{\partial y^2} = 0 \quad (1)$$

Debido a que los contornos del recinto donde debemos resolver la ecuación 1 son los adecuados, elegimos utilizar el método de **Diferencias Finitas**.

1.1. Discretización del Problema

El primer paso en este sentido consiste en discretizar el dominio de la chapa en un conjunto de puntos regularmente distribuidos. Al tratarse de un recinto cuadrado, este paso es trivial.

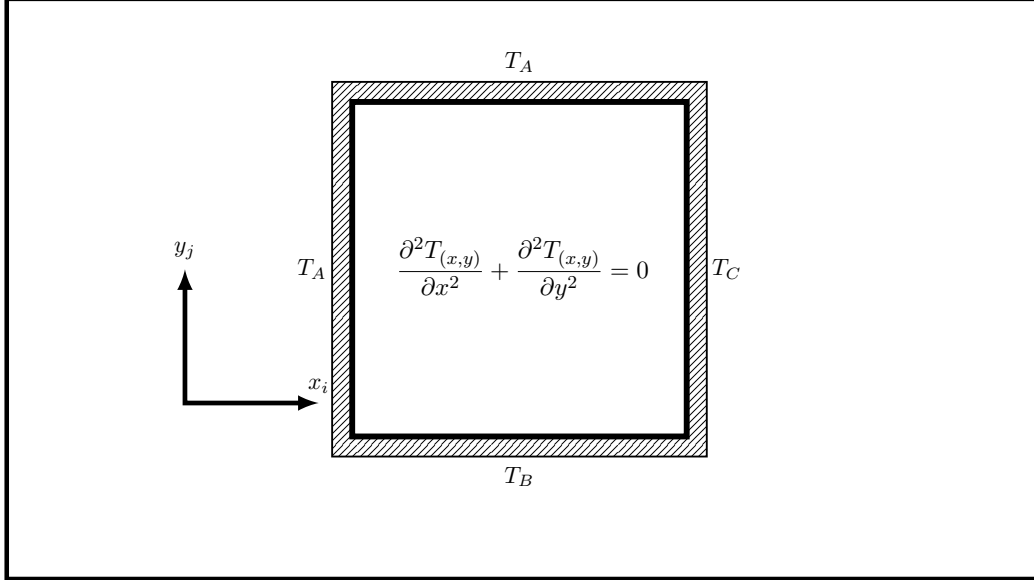


Figura 1: Presentación del Problema. Chapa bidimensional de conductividad térmica finita, cuyos bordes se encuentran a temperaturas constantes finitas T_A , T_B , T_C , T_D .

Para formar la grilla dividimos los lados sobre los ejes x e y del cuadrado en $N_x - 1$ y $N_y - 1$ segmentos iguales, determinando N_x y N_y puntos equidistantes sobre cada uno. De las intersecciones de las líneas paralelas a los lados que se extienden de los puntos formados se genera la grilla a usar, como se muestra en la Figura 2. Puede verse inmediatamente que la cantidad de nodos total de la grilla es $N_x N_y$.

Para poder asignar una temperatura a cada nodo de la grilla, debemos tomar una convención para numerarlos. La primera intuición tiene que ver con una numeración de índices según las dos direcciones cartesianas x e y . Necesitamos entonces dos índices i y j , de manera que las temperaturas $T(x_i, y_j)$ para los nodos de la grilla pueden representarse en forma matricial $T_{i,j}$.

Sin embargo, por razones que pronto quedarán claras, es conveniente tomar una numeración de un único índice k para cada nodo. Con alguna arbitrariedad, puede tomarse una numeración correlativa asignando $k = 1$ al nodo del el vértice inferior izquierdo, hacia la derecha. En cada “fin de línea” se toma el índice siguiente para el primer nodo de la línea superior. Con esta convención, puede verse que los vértices de la chapa quedan identificados en

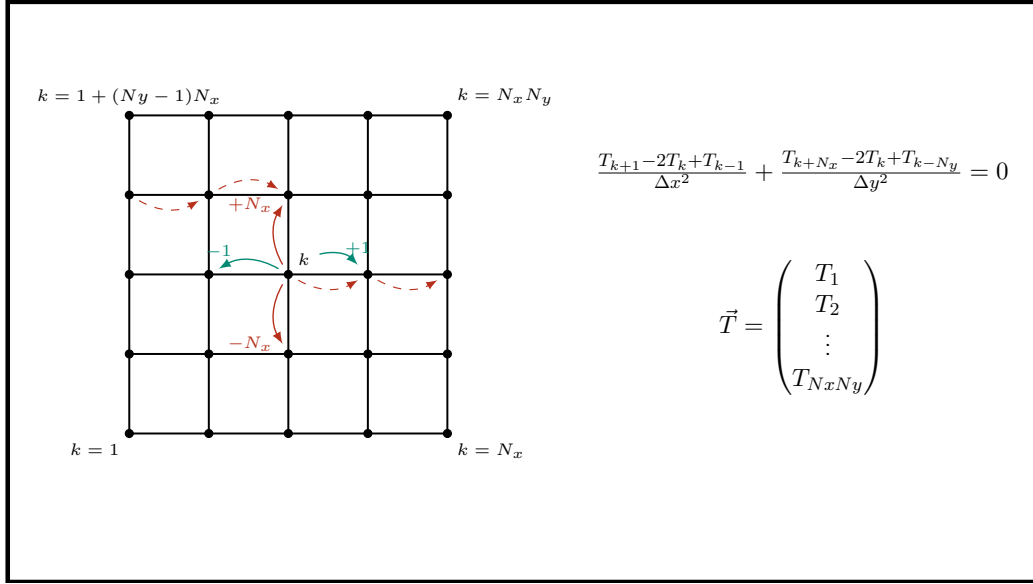


Figura 2: Grilla de discretización del dominio de la chapa

función de las dimensiones de la grilla elegida

$$\begin{aligned}
 &\text{Vertice inferior izquierdo: } k = 1 \\
 &\text{Vertice inferior derecho: } k = N_x \\
 &\text{Vertice superior izquierdo: } k = N_x N_y - N_x + 1 \\
 &\text{Vertice superior derecho: } k = N_x N_y
 \end{aligned} \tag{2}$$

Ahora, las temperaturas de la barra pueden representarse en un **vector**

$$\vec{T} = \begin{pmatrix} T_1 \\ T_2 \\ \vdots \\ T_{N_x N_y} \end{pmatrix} \tag{3}$$

Es conveniente encontrar una relación entre los sistemas de índices. como $1 \leq i \leq N_x$; $1 \leq j \leq N_y$, puede verse fácilmente que dicha relación es

$$\begin{aligned}
 k &= i + (j - 1)N_x \\
 j &= \text{floor}\left(\frac{k - 1}{N_x} + 1\right) \\
 i &= k - (j - 1)N_x
 \end{aligned} \tag{4}$$

donde la función *floor* indica que debe tomarse la parte entera inferior de su argumento. Por otro lado, las relaciones de la ecuación 4 permite relacionar ambas convenciones de índices en forma biyectiva, de manera que podamos relacionarlo con la numeración de dos índices i, j en forma unívoca.

Según el método de las diferencias finitas, debemos usar las versiones *discretas* de las derivadas segundas en la ecuación 1. Puede consultar la teórica para rescatar estas definiciones, pero aquí razonaremos en base a las mismas, que las derivadas segundas pueden calcularse a partir de las derivadas primeras.

Para empezar por el problema más sencillo, busquemos la segunda derivada parcial respecto de x como un cociente incremental de las derivadas primeras. Dado un nodo de numeración k que no se encuentre en ningún borde de la chapa, notemos según nuestra convención el punto siguiente hacia las x positivas es el número $k + 1$, mientras que el nodo hacia las x negativas es el $k - 1$. Podemos definir entonces las derivadas primera y segunda en el nodo k -ésimo como

$$\begin{aligned} \left. \frac{\partial T}{\partial x} \right|_k &= \frac{T_k - T_{k-1}}{\Delta x} &= dT_k \\ \left. \frac{\partial^2 T}{\partial x^2} \right|_k &= \frac{dT_{k+1} - dT_k}{\Delta x} = \frac{1}{\Delta x} \left(\frac{T_{k+1} - T_k}{\Delta x} - \frac{T_k - T_{k-1}}{\Delta x} \right) \\ &\Rightarrow = \frac{T_{k+1} - 2T_k + T_{k-1}}{\Delta x^2} \end{aligned} \quad (5)$$

Para poder calcular la derivada segunda en la otra dirección debemos notar que es necesario tomar cocientes incrementales de la temperatura hacia arriba y hacia abajo. Debe notar que para un nodo de índice k , el nodo de *arriba* tiene el orden $k + N_x$. Haga el ejercicio de obtener la derivada segunda respecto de y

$$\left. \frac{\partial^2 T}{\partial y^2} \right|_k = \frac{T_{k+N_x} - 2T_k + T_{k-N_y}}{\Delta y^2} \quad (6)$$

2. Matricialización del Probema

Siguiendo el método de las diferencias Finitas, sólo resta reemplazar las ecuaciones 5 y 6 en la 1. Puede verse fácilmente¹ que *para el conjunto de*

¹hágalo, ¡atrévase!

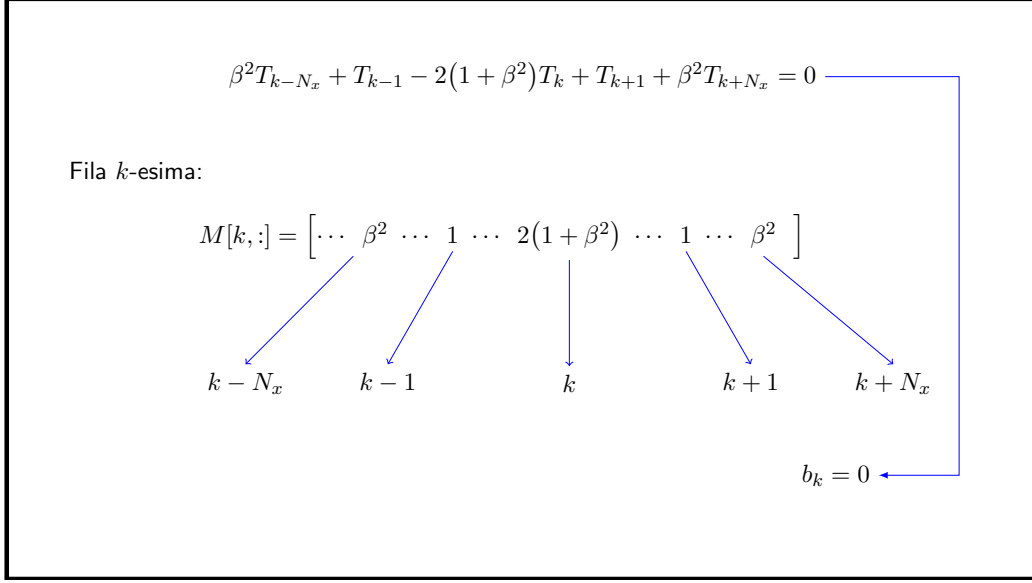


Figura 3: Ilustración de la construcción de los coeficientes de matriz para la fila k -ésima de la matriz del sistema lineal, para algún k correspondiente a un nodo del interior de la chapa.

odos del interior de la chapa se puede escribir la ecuación

$$\frac{T_{k+1} - 2T_k + T_{k-1}}{\Delta x^2} + \frac{T_{k+N_x} - 2T_k + T_{k-N_x}}{\Delta y^2} = 0 \quad (7)$$

$$\beta^2 T_{k-N_x} + T_{k-1} - 2(1 + \beta^2)T_k + T_{k+1} + \beta^2 T_{k+N_x} = 0$$

donde se ha definido

$$\beta = \frac{\Delta x}{\Delta y} \quad (8)$$

La ecuación 7 es válida para todos los nodos de índice k tal que siguiendo la biyección de la ecuación 4 se tiene que $0 < i < N_x$ y $0 < j < N_y$, donde el uso del menor estricto refleja que los nodos no están en los bordes. Para los nodos de los bordes debemos tomar una consideración especial.

Aún en este caso general, podemos ver lo siguiente. Cada nodo interno de la chapa cumple la ecuación 7. Si las temperaturas de los nodos están dispuestas en el vector \vec{T} de la ecuación 3 la ecuación diferencial surge de multiplicar un vector fila por el vector columna \vec{T} . El conjunto de ecuaciones para todos los k internos genera entonces un sistema lineal cuyos coeficientes están bien determinados en la ecuación 7, como se muestra en la Figura 3.

3. Condiciones de Contorno

3.1. Borde a Temperatura Fija

Para describir el problema en forma completa, las condiciones de contorno deben ser dadas. Comencemos por el caso más sencillo en el que se conoce el valor de la función incógnita en el borde del recinto de integración, tal como se mostró en la figura 1.

Para poder expresar estas condiciones de contorno en nuestro modelo computacional, debemos pensar primero en la numeración que recibirán los nodos de cada borde. podemos entonces tomar los vectores de índices que los describen. Retomando la numeración descrita en la Figura 2, podemos escribir los siguientes vectores

$$\begin{aligned} k_A &= [1, 1 + Nx, \dots, 1 + (N_y - 1)N_x] \\ k_B &= [Nx, 2Nx, \dots, N_y N_x] \\ k_C &= [1, 2, \dots, N_x] \\ k_D &= [1 + (N_y - 1)N_x, 2 + (N_y - 1)N_x, \dots, N_x N_y] \end{aligned} \tag{9}$$

Para cualquiera de los nodos cuyo índice k puede encontrarse en alguno de los vectores de la ecuación 9, se tiene que la ecuación que *deben cumplir* es

$$T_{k \in k_A} = T_A; T_{k \in k_B} = T_B; T_{k \in k_C} = T_C; T_{k \in k_D} = T_D \tag{10}$$

Estas últimas condiciones son triviales de conseguir si se cambian los coeficientes de las filas de la matriz correspondientes y el vector \vec{b} como se muestra en la Figura . Se muestra que como la ecuación se ha cambiado por la 10, solo el coeficiente de la diagonal es distinto de cero. El resto de los coeficientes en las columnas no diagonales es nulo ya que a la derecha del igual los nodos con $k' \neq k$ aparecen solo en forma implícita multiplicados por cero.

3.1.1. Vértices

El caso de los vértices debe ser tratado en forma separada. esto se debe a que según la definición de los bordes en la ecuación 9, los nodos en los vértices

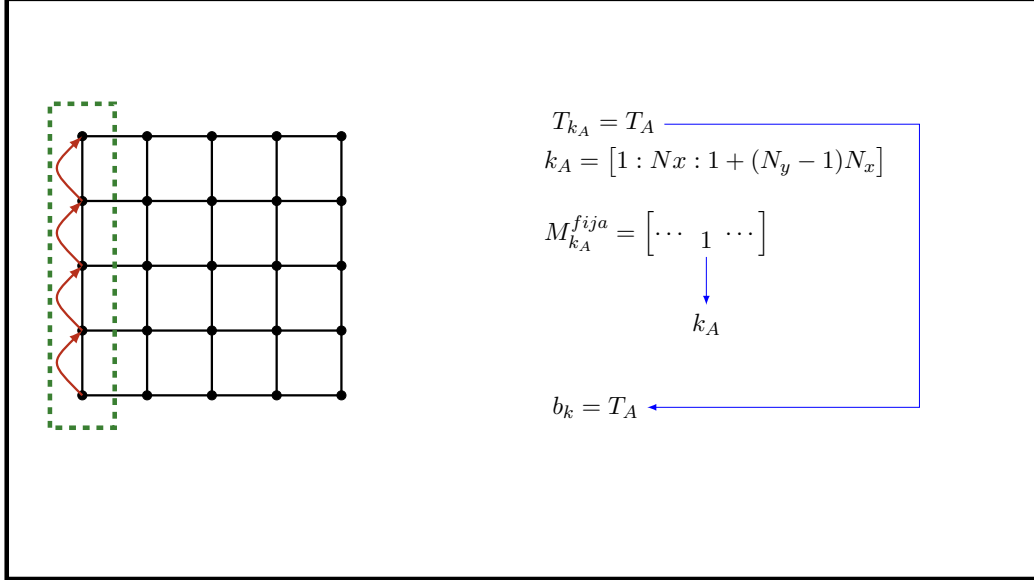


Figura 4: Coeficientes de matriz para los nodos en cualquiera de los bordes del recinto del problema.

pueden pertenecer a dos bordes simultáneamente. Por ejemplo el nodo $k = 1$ pertenece a k_A y a K_C .

Pueden tomarse varias estrategias para atacar estos problemas. Podemos tomar preferencia por alguno de los bordes. Entonces la lógica que adoptaremos debe evaluar primero si los nodos están en los bordes según la preferencia dada.

Otra alternativa es que la lógica de la solución evalúe primero si el nodo está en algún borde, se le asigne alguna condición de contorno específica. Una vez descartada la pertenencia a todos los vértices, evaluar la pertenencia a los bordes. En este último caso es sencillo asignar por ejemplo el promedio de las temperaturas de los bordes subyacentes al vértice, por ejemplo.

4. Lógica

Sabiendo los valores de los coeficientes de la matriz que representa a nuestra ecuación diferencial, podemos trazar la lógica de nuestro modelo computacional.

Cuando inicie el estudio del problema, no use tamaños de grillas grandes. Utilice la mínima grilla posible, por ejemplo $N_x = N_y = 3$ para poder interpretar los resultados en forma legible.

5. Resultados con temperaturas Fijas

Una vez se haya armado la matriz, debe resolver el sistema lineal

$$\mathbf{A}\vec{T} = \vec{b} \quad (11)$$

Utilice las subrutinas especializadas del lenguaje que haya elegido para implementar el método. En la figura 5 se muestran las órdenes en **Matlab** y en **python**. Por el otro lado, en la Figura se muestra las curvas de nivel obtenidas imponiendo temperaturas arbitrarias en los bordes. Al mismo tiempo se muestran los ajustes lineales de los tiempos de ejecución de las subrutinas de resolución del sistema lineal.

Se usaron programas redactados en **python** y **FORTTRAN** como ejemplo de los dos tipos de lenguajes compilables y no compilables. Por un lado, los tiempos en **FORTTRAN** son menores que en **python**. Por otro lado, los tiempos medidos para la ejecución de la solución del sistema en **python** ajustan mal al modelo lineal, lo que se evidencia en el mayor error en los parámetros de ajuste. Esto se debe a que el programa **FORTTRAN** se ha compilado para que la ejecución se implemente en un ejecutable binario que interactúa con la CPU ‘con menos intermediarios’, por decirlo de alguna manera. Por otro lado, el intérprete **python** debe pasar el programa a lenguaje de máquina en tiempo de ejecución, lo que evidentemente insume un tiempo considerable.

6. Cálculo de Flujos

Una vez obtenida la distribución de temperaturas, podemos medir, por ejemplo, los flujos de calor que de ella se derivan. Como sabemos, los flujos de calor son proporcionales a la derivada primera de la temperatura. En los nodos interiores, es posible aproximar esta derivada por un cociente incremental centrado como se esquematiza en la figura 7. Sin embargo, en los bordes es necesario tomar algunos cuidados. Por ejemplo, es posible optar por tomar cocientes incrementales hacia la derecha para los nodos en el borde izquierdo, etc.

Resolver:	Matlab	<code>>> T = A \ b</code>
	python	<code>>>> T=numpy.linalg.solve(A,b)</code>
Graficar:	Matlab	<code>>> x=[0:dx:1];</code> <code>>> y=[0:dy:1];</code> <code>>> [X,Y]=meshgrid(x,y);</code> <code>>> countourf(X,Y,Tmatriz)</code>
	python	<code>>>> x=numpy.linspace([0,1,dx])</code> <code>>>> y=numpy.linspace([0,1,dy])</code> <code>>>> [X,Y]=numpy.meshgrid(x,y)</code> <code>>>> ax=matplotlib.pyplot.contourf(X,Y,Tmatriz)</code>

Figura 5: Órdenes Matlab y python para resolver sistemas lineales

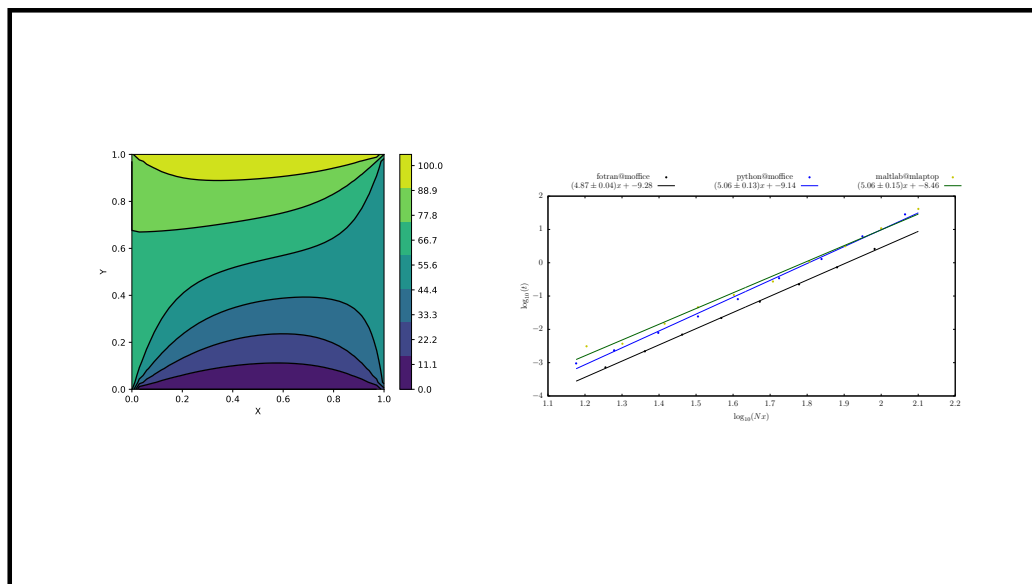


Figura 6: Resultado en el caso de bordes a temperatura fija

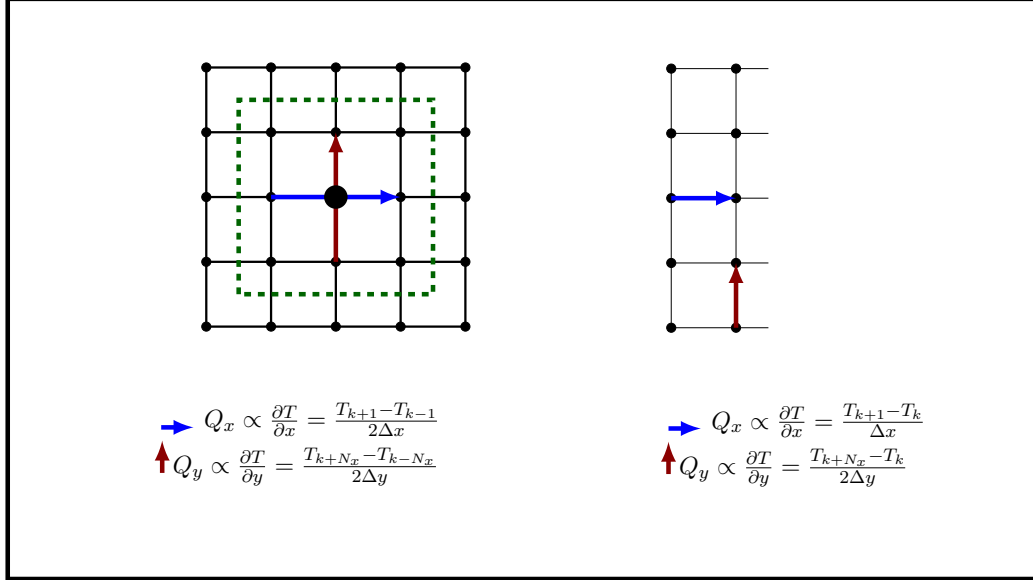


Figura 7: Derivadas Primeras para los nodos interiores y del borde.

Pueden utilizarse varios métodos para visualizar los campos vectoriales como el flujo, pero en general es necesario ejecutar una instrucción cuyos argumentos serán las posiciones a las que corresponden cada vector, y las componentes x e y para cada punto del recinto calculado. El resultado se muestra en la Figura 8

7. Condiciones de contorno 2: Flujo dado

Hasta aquí resolvimos el problema con condiciones de contorno de temperaturas fijas. Si queremos incorporar una condición de contorno para derivadas (flujos) sobre alguno de los bordes debemos tener en cuenta el siguiente truco. Consideremos como ejemplo el flujo entrante hacia la chapa sobre el borde izquierdo. En un punto de ese borde, por ejemplo el punto k_A podemos aproximar el flujo entrante de calor a una cantidad proporcional a la derivada primera que calculamos con un cociente incremental centrado en el mismo punto. Para eso, como se ilustra en la Figura 9 debemos considerar algún punto auxiliar \tilde{k} fuera de la chapa. A no desesperar, que como se ve en la Ecuación 12, este punto extra solo nos sirve para despejar la temperatura T_{k_A} en función de las temperaturas del interior de la chapa y el flujo Q_{xA} que viene dado por la condición de contorno.

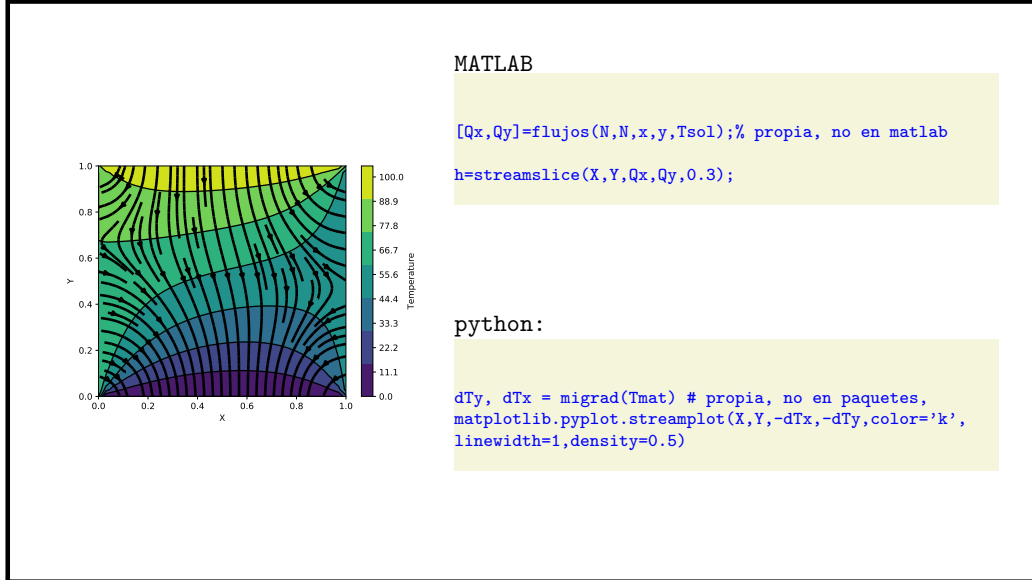


Figura 8: Resultado final incluyendo el flujo de calor para las condiciones de borde de temperatura

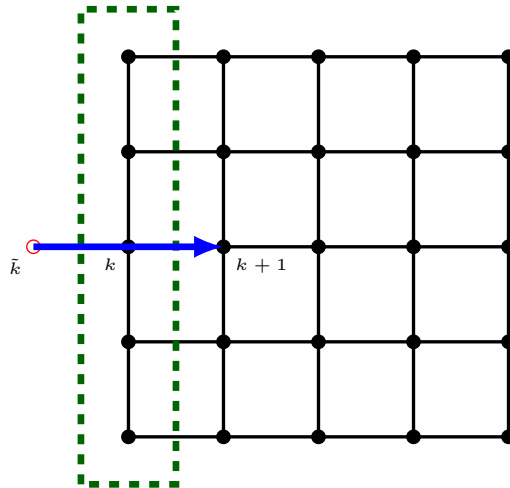


Figura 9: Para aproximar la temperatura en el punt k -ésimo tomamos un punto auxiliar \tilde{k} afuera del recinto de integración

$$\begin{aligned}
& T_{\tilde{k}} = T_{k_A+1} - 2\Delta x q_{xA} \quad k_A = 1 : N_x : N_x(N_y - 1)N_x \\
& \quad \text{reemplazando en la ecuación general:} \\
& \beta^2 T_{k-N_x} + T_{k-1} - 2(1 + \beta^2)T_k + T_{k+1} + \beta^2 T_{k+N_x} = 0 \\
& \quad \text{reordenando:} \\
& \beta^2 T_{k-N_x} - 2(1 + \beta^2)T_k + 2T_{k+1} + \beta^2 T_{k+N_x} = 2\Delta x q_{xA} \\
& \text{Fila } k_A\text{-esima:} \\
& M[k, :] = \left[\dots \quad \beta^2 \quad \dots \quad 0 \quad 2(1 + \beta^2) \quad 2 \quad \dots \quad \beta^2 \right] \\
& \quad \quad \quad \swarrow \quad \swarrow \quad \downarrow \quad \searrow \quad \searrow \\
& \quad \quad \quad k - N_x \quad k - 1 \quad k \quad k + 1 \quad k + N_x \\
& \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad b_k = 2\Delta x q_{xA}
\end{aligned}$$

Figura 10: Al reemplazar la temperatura del nodo auxiliar en la ecuación para los nodos internos, tenemos la nueva ecuación para los nodos del borde.

$$\begin{aligned}
Q_x \propto q_{xA} &= \left. \frac{\partial T}{\partial x} \right|_{k_A} \\
&= \frac{T_{k_A+1} - T_{\tilde{k}}}{2\Delta x} \\
& \quad k_A = 1 : N_x : (N_y - 1)N_x + 1
\end{aligned} \tag{12}$$

Una vez hecho esto, el “truco” es detectar en la ecuación homogénea cuál es el coeficiente de matriz que no podemos incluir. Evidentemente, es el coeficiente $(k, k - 1)$ el que nos daría problemas en nuestro modelo. Sin embargo con nuestra descripción de \tilde{k} podemos reemplazar T_{k-1} por $T_{\tilde{k}}$ de la Ecuación (12) en la ecuación (7).

Tenemos entonces que los nuevos coeficientes de matriz pueden reconocerse fácilmente como se ve en la Figura 10. Notar que los coeficientes que cambian son solo el $k_A - 1$ y el $k_A + 1$ de manera que solo tiene un valor distinto de cero el coeficiente de matriz que acopla al punto k_A con un punto interno de la chapa.

Método de las Diferencias Finitas

Modelización de Propiedades y Procesos 2020

Ruben Weht^{1,2} Mariano Forti^{1,3}

¹Instituto de Tecnología Prof. Jorge Sabato

²Física del Sólido, Edificio TANDAR, weht@cnea.gov.ar, interno 7104

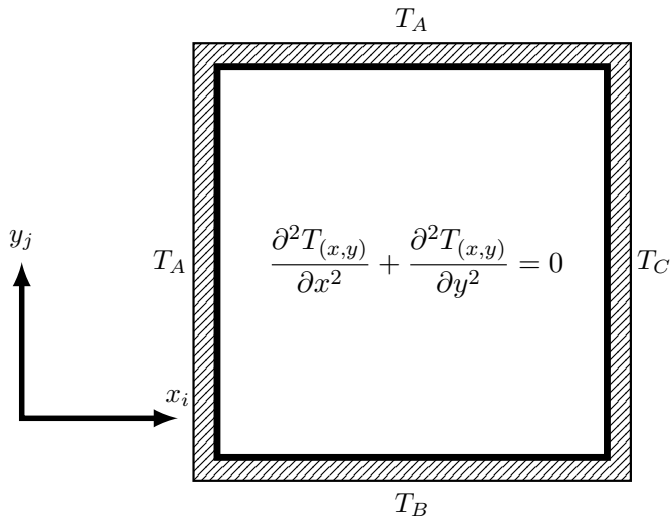
³División Aleaciones Especiales, Edificio 47 (microscopía), mforti@cnea.gov.ar, interno 7832



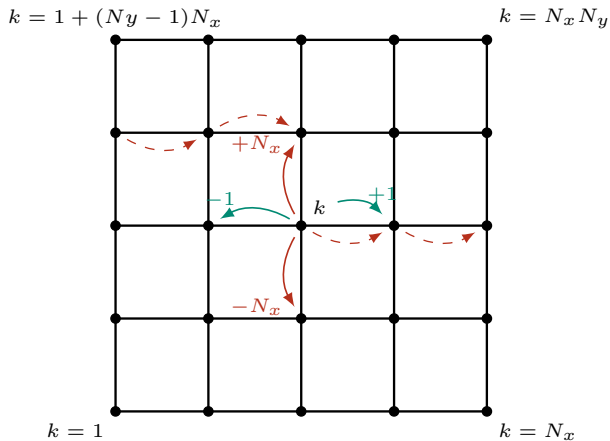
Método de las Diferencias Finitas

Problema de la chapa

Presentación del Problema



Discretización del Problema



$$\frac{T_{k+1} - 2T_k + T_{k-1}}{\Delta x^2} + \frac{T_{k+N_x} - 2T_k + T_{k-N_y}}{\Delta y^2} = 0$$

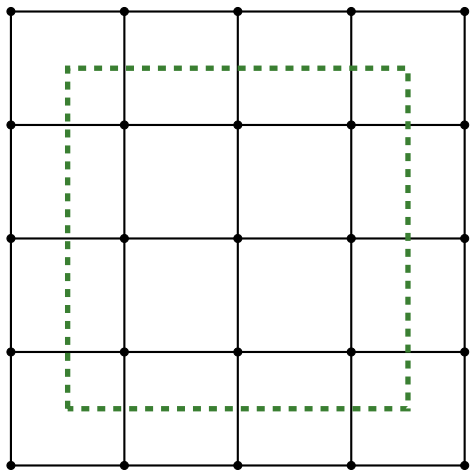
$$\vec{T} = \begin{pmatrix} T_1 \\ T_2 \\ \vdots \\ T_{NxNy} \end{pmatrix}$$

Método de las Diferencias Finitas

Matricialización del Problema

Matricialización de la Ecuación Diferencial

$$\beta^2 T_{k-N_x} + T_{k-1} - 2(1 + \beta^2)T_k + T_{k+1} + \beta^2 T_{k+N_x} = 0$$



$$[\mathbf{M}] \begin{pmatrix} T_1 \\ T_2 \\ \vdots \\ T_{N_x N_y} \end{pmatrix} = \vec{b}$$

Coeficientes de Matriz

$$\beta^2 T_{k-N_x} + T_{k-1} - 2(1 + \beta^2)T_k + T_{k+1} + \beta^2 T_{k+N_x} = 0$$

Fila k -ésima:

$$M[k, :] = \left[\dots \quad \beta^2 \quad \dots \quad 1 \quad \dots \quad 2(1 + \beta^2) \quad \dots \quad 1 \quad \dots \quad \beta^2 \right]$$

$k - N_x$

$k - 1$

k

$k + 1$

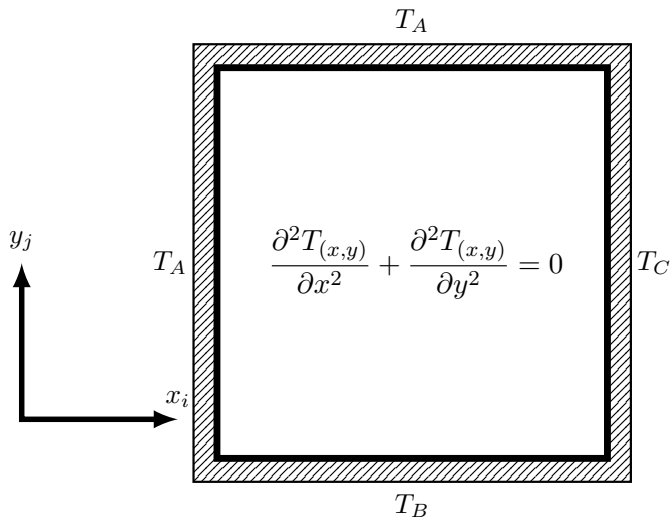
$k + N_x$

$$b_k = 0$$

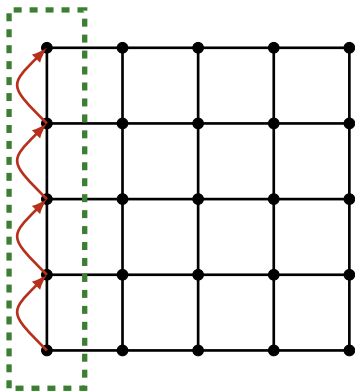
Método de las Diferencias Finitas

Condiciones de Contorno

Bordes a temperatura Fija



Borde a temperatura fija



$$T_{k_A} = T_A$$

$$k_A = ? \text{ Algun Rango ...}$$

$$M_{k_A}^{fija} = \begin{bmatrix} \cdots & 1 & \cdots \end{bmatrix}$$

↓
 k_A

$$b_k = T_A$$

Método de las Diferencias Finitas

Lógica

Lógica del programa

- 1 Recorrer todos los valores posibles para el índice k , es decir los nodos

Lógica del programa

- 1 Recorrer todos los valores posibles para el índice k , es decir los nodos
- 2 decidir si el nodo está en un borde o en el interior de la chapa

Lógica del programa

- 1 Recorrer todos los valores posibles para el índice k , es decir los nodos
- 2 decidir si el nodo está en un borde o en el interior de la chapa
- 3 llenar la fila correspondiente de la matriz y al vector \vec{b} en consecuencia.

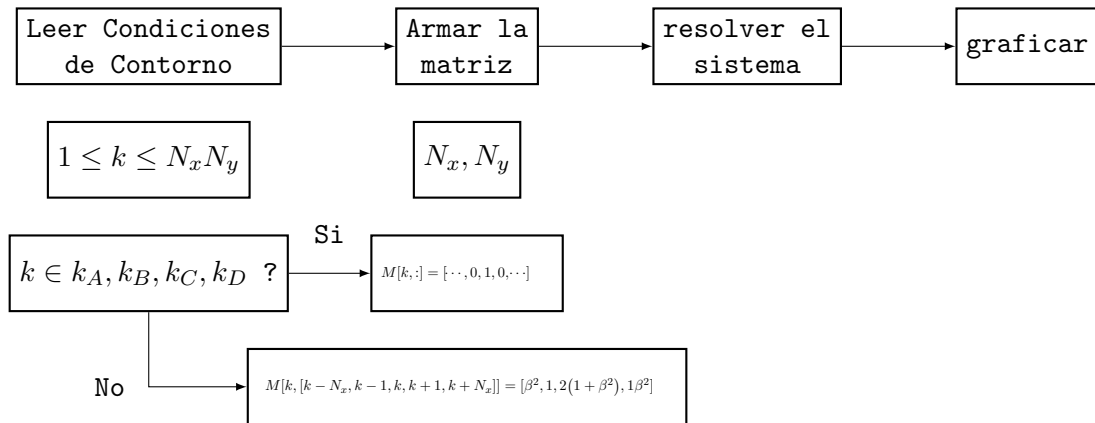
Lógica del programa

- 1 Recorrer todos los valores posibles para el índice k , es decir los nodos
- 2 decidir si el nodo está en un borde o en el interior de la chapa
- 3 llenar la fila correspondiente de la matriz y al vector \vec{b} en consecuencia.
- 4 resolver el sistema lineal

Lógica del programa

- 1 Recorrer todos los valores posibles para el índice k , es decir los nodos
- 2 decidir si el nodo está en un borde o en el interior de la chapa
- 3 llenar la fila correspondiente de la matriz y al vector \vec{b} en consecuencia.
- 4 resolver el sistema lineal
- 5 Graficar !

Lógica del modelo



Método de las Diferencias Finitas

Resultados con temperaturas Fijas

Códigos de solución

Resolver:

Matlab

```
>> T = A \ b
```

python

```
>>> T=numpy.linalg.solve(A,b)
```

Graficar:

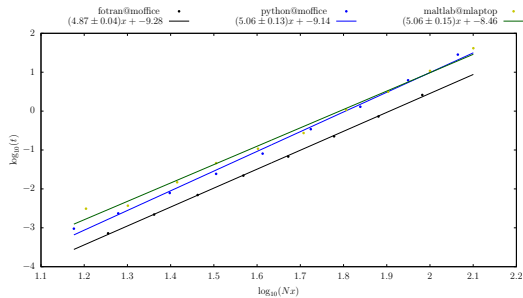
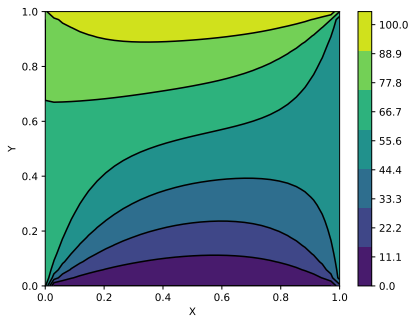
Matlab

```
>> x=[0:dx:1];  
>> y=[0:dy:1];  
>> [X,Y]=meshgrid(x,y);  
>> countourf(X,Y,Tmatriz)
```

python

```
>>> x=numpy.linspace([0,1,dx])  
>>> y=numpy.linspace([0,1,dy])  
>>> [X,Y]=numpy.meshgrid(x,y)  
>>> ax=matplotlib.pyplot.countourf(X,Y,Tmatriz)
```

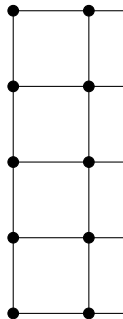
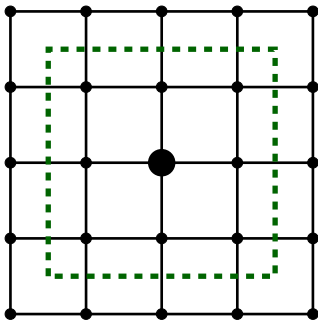
Resultado primero



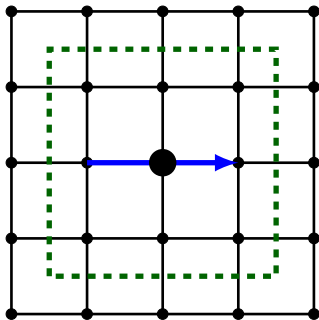
Método de las Diferencias Finitas

Cálculo de Flujos

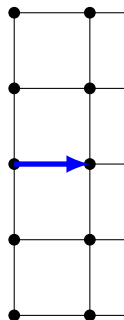
Cálculo de los Flujos de Calor



Cálculo de los Flujos de Calor

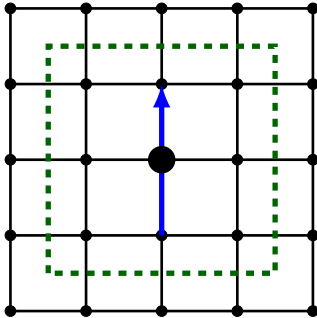


$$\rightarrow Q_x \propto \frac{\partial T}{\partial x} = \frac{T_{k+1} - T_{k-1}}{2\Delta x}$$

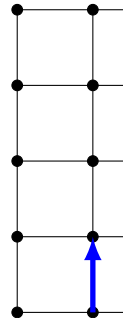


$$\rightarrow Q_x \propto \frac{\partial T}{\partial x} = \frac{T_{k+1} - T_k}{\Delta x}$$

Cálculo de los Flujos de Calor

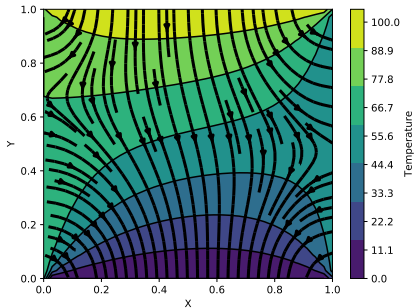


$$\uparrow Q_y \propto \frac{\partial T}{\partial y} = \frac{T_{k+N_x} - T_{k-N_x}}{2\Delta y}$$



$$\uparrow Q_y \propto \frac{\partial T}{\partial y} = \frac{T_{k+N_x} - T_k}{2\Delta y}$$

Resultado Final para Temperaturas Fijas



MATLAB

```
[Qx,Qy]=flujos(N,N,x,y,Tsol);% propia, no en matlab  
h=streamslice(X,Y,Qx,Qy,0.3);
```

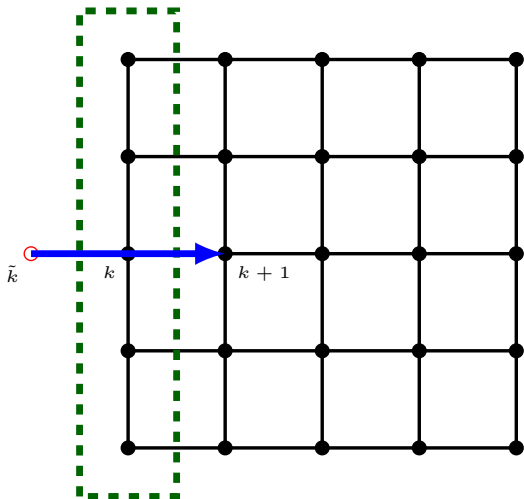
python:

```
dTy, dTx = migrad(Tmat) # propia, no en paquetes,  
matplotlib.pyplot.streamplot(X,Y,-dTx,-dTy,color='k',  
linewidth=1,density=0.5)
```

Método de las Diferencias Finitas

Condiciones de contorno 2: Flujo dado

Flujo en el borde izquierdo



$$\begin{aligned} Q_x \propto q_{xA} &= \left. \frac{\partial T}{\partial x} \right|_{k_A} \\ &= \frac{T_{k_A+1} - T_{\tilde{k}}}{2\Delta x} \\ k_A &= 1 : N_x : (N_y - 1)N_x + 1 \end{aligned}$$

Coeficientes de Matriz

$$\tilde{T}_k = T_{k_A+1} - 2\Delta x q_{xA} \quad k_A = 1 : N_x : N_x(N_y - 1)N_x$$

reemplazando en la ecuación general:

$$\beta^2 T_{k-N_x} + T_{k-1} - 2(1 + \beta^2)T_k + T_{k+1} + \beta^2 T_{k+N_x} = 0$$

reordenando:

$$\beta^2 T_{k-N_x} - 2(1 + \beta^2)T_k + 2T_{k+1} + \beta^2 T_{k+N_x} = 2\Delta x q_{xA}$$

Fila k_A -esima:

$$M[k, :] = \left[\dots \quad \beta^2 \quad \dots \quad 0 \quad 2(1 + \beta^2) \quad 2 \quad \dots \quad \beta^2 \right]$$

$$\begin{array}{ccccccccc} & \swarrow & & \swarrow & \downarrow & \searrow & & \searrow & \\ & k - N_x & & k - 1 & k & k + 1 & & k + N_x & \end{array}$$

$$b_k = 2\Delta x q_{xA}$$