



**Universidad Tecnológica Nacional**

**Facultad Regional Concepción del Uruguay**

**Ingeniería en Sistemas de Información**

# **Matemática Superior**

**Trabajo Práctico Final: Ecuación del calor en 2D**

**Alumno:**

- Gastón Gomariz Moser

**Docentes:**

- Dr. Omar Faure
- Ing. Gastón Quiroga

# Índice

<b>Introducción</b>	<b>3</b>
<b>Objetivos</b>	<b>4</b>
<b>Desarrollo del modelo matemático</b>	<b>5</b>
Consideraciones iniciales	5
Ecuación de la temperatura en el siguiente instante	6
Numeración de los nodos	7
Condiciones de borde	8
Forma matricial	9
Análisis del error	11
<b>Implementación en MATLAB</b>	<b>13</b>
Datos iniciales	13
Ejecución	13
<b>Conclusión</b>	<b>16</b>
<b>Bibliografía</b>	<b>17</b>

# Introducción

A medida que avanza la tecnología, las computadoras desarrollan componentes más y más potentes, los cuales requerirán mayores cantidades de energía para operar. Debido a las leyes de la física, esto trae como consecuencia que el dispositivo alcance temperaturas mucho más elevadas, por lo que suele recurrirse a distintos métodos para mitigarlas. Una de las soluciones más frecuentes se trata de emplear disipadores, los cuales consisten de una serie de placas, por lo general hechas de aluminio, a través de las cuales se hace circular el calor hacia el exterior.

# Objetivos

El objetivo de este trabajo práctico es poder desarrollar un modelo matemático utilizando el método de diferencias finitas que nos permita estudiar y evaluar la circulación de calor a lo largo del tiempo a través de una placa de aluminio ficticia de 2 dimensiones, así como un programa en MATLAB que nos permita calcular y visualizar la evolución de la temperatura en dicha placa.

# Desarrollo del modelo matemático

Sea una placa de 2 dimensiones de longitud horizontal  $L$ , longitud vertical  $H$ , hecha de un material homogéneo de conductividad  $\kappa$ , densidad  $\rho$ , calor específico  $C$ , y transmitancia térmica  $h_c$  busquemos un modelo matemático que nos permita conocer su temperatura en un instante dado.

## Consideraciones iniciales

Antes de comenzar con el desarrollo del modelo, debemos establecer algunas aproximaciones que serán importantes para este:

- **Aproximación de la derivada primera:** Podemos hacer una aproximación sencilla de la derivada primera mediante una serie de Taylor:

Sea  $f$  una función de un conjunto  $U \subseteq \mathbb{R} \rightarrow \mathbb{R}$  la cual es al menos clase  $C^1$ , y  $x_0 \in U$ , podemos considerar como una buena aproximación a:

$$\frac{df(x_0)}{dx} \approx \frac{f(x_0 + \Delta_x) - f(x_0)}{\Delta_x}$$

Esta será la ecuación (1).

- **Aproximación de la derivada segunda:** De manera similar a la anterior, podemos también aproximar la derivada segunda de la función.

Sea  $f$  una función de un conjunto  $U \subseteq \mathbb{R} \rightarrow \mathbb{R}$  la cual es al menos clase  $C^2$ , y  $x_0 \in U$ , podemos considerar como una buena aproximación a:

$$\frac{d^2 f(x_0)}{dx^2} \approx \frac{f(x_0 + \Delta_x) - 2 * f(x_0) + f(x_0 - \Delta_x)}{\Delta_x^2}$$

Esta será la ecuación (2).

- **Evolución de la temperatura:** Para conocer la variación de la temperatura en la placa a lo largo del tiempo, necesitamos una función que tome como argumentos a la posición horizontal  $x$ , y la posición vertical  $y$  de un segmento dentro de esta, así como el instante de evaluación  $t$ .

Llamamos entonces  $u$  a la función de un conjunto  $U \subseteq \mathbb{R}^3 \rightarrow \mathbb{R}$ , la cual es al menos clase  $C^2$ , y cuyos argumentos son  $x$ ,  $y$ , y  $t$ . La ecuación que da la variación del calor en la placa a lo largo del tiempo será:

$$\frac{\partial u}{\partial t} = \frac{\kappa}{\rho * C} * \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right)$$

Esta será la ecuación (3).

## Ecuación de la temperatura en el siguiente instante

Comenzamos definiendo  $u_{ij k}$  como la temperatura en el nodo  $i, j$  de la placa en el instante  $k$ , a fin de simplificar la notación de las ecuaciones.

Luego, para obtener la ecuación que nos permite aproximar esta temperatura, reemplazamos las ecuaciones (1) y (2) en (3):

$$\frac{\partial u}{\partial t} = \frac{\kappa}{\rho * C} * \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right)$$

$$\frac{u_{ij k+1} - u_{ij k}}{\Delta_t} = \frac{\kappa}{\rho * C} * \left( \frac{u_{i+1 j k} - 2 * u_{ij k} + u_{i-1 j k}}{\Delta_x^2} + \frac{u_{i j+1 k} - 2 * u_{ij k} + u_{i j-1 k}}{\Delta_y^2} \right)$$

Si  $u_{ij k}$  es la temperatura del nodo  $i, j$  en el presente,  $u_{ij k+1}$  será la temperatura del mismo nodo en la siguiente iteración. Por lo tanto, podemos utilizar la ecuación anterior para conocer la temperatura en el próximo instante con datos de la iteración actual. Entonces, si para simplificar la fórmula consideramos  $\Delta_x = \Delta_y$ , queda:

$$\frac{u_{ij k+1} - u_{ij k}}{\Delta_t} = \frac{\kappa}{\rho * C * \Delta_x^2} * (u_{i+1 j k} + u_{i-1 j k} - 4 * u_{ij k} + u_{i j+1 k} + u_{i j-1 k})$$

Además, definimos  $r = \frac{\kappa * \Delta_t}{\rho * C * \Delta_x^2}$ . De esta manera, seguimos despejando y queda:

$$u_{ij k+1} - u_{ij k} = r * (u_{i+1 j k} + u_{i-1 j k} - 4 * u_{ij k} + u_{i j+1 k} + u_{i j-1 k})$$

$$u_{ij k+1} = r * (u_{i+1 j k} + u_{i-1 j k} - 4 * u_{ij k} + u_{i j+1 k} + u_{i j-1 k}) + u_{ij k}$$

$$u_{ij k+1} = r * (u_{i+1 j k} + u_{i-1 j k} + u_{i j+1 k} + u_{i j-1 k}) + (1 - 4 * r) * u_{ij k}$$

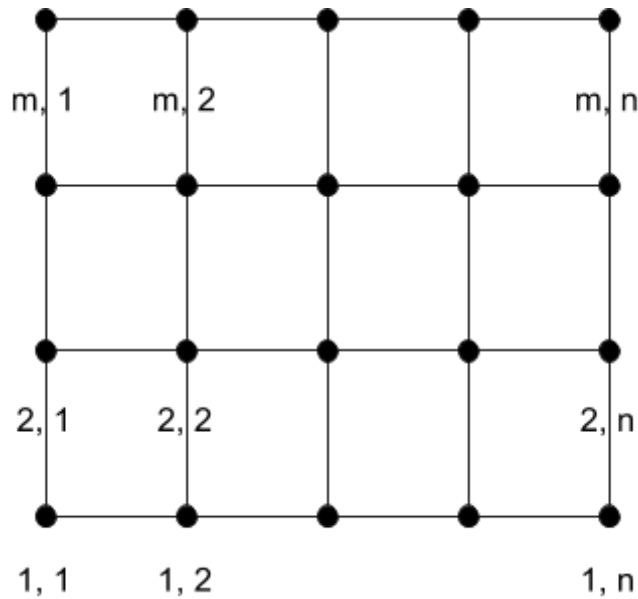
Esta será la ecuación (4).

Con esta fórmula podemos calcular la temperatura en un punto en el instante  $k+1$  conociendo la del mismo nodo y los adyacentes durante el instante  $k$ .

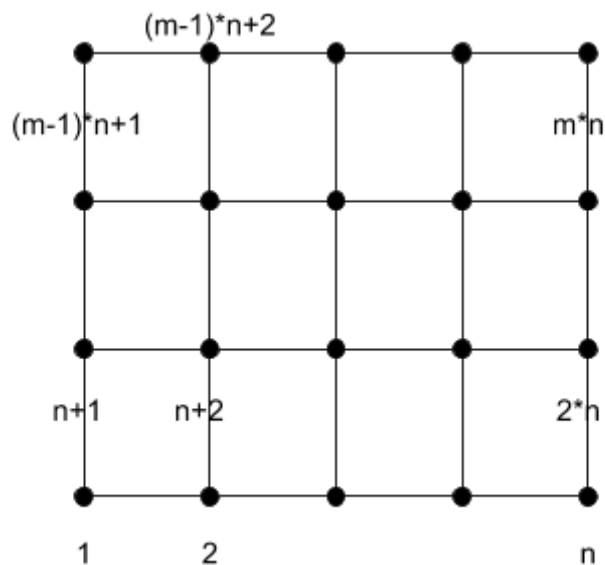
Podemos plantear entonces un sistema de ecuaciones que represente la evolución de la temperatura en toda la placa a lo largo de todo el tiempo. Sin embargo, antes de hacerlo debemos establecer algunas consideraciones respecto a las ecuaciones y su estructura.

## Numeración de los nodos

Primero, debido a que estamos trabajando con el método de diferencias finitas, podemos pensar en la placa no como un arreglo de nodos de 2 dimensiones, si no como un vector en el que los puntos están enumerados de manera secuencial. Para ello los recorreremos primero de izquierda a derecha y luego de abajo hacia arriba, del modo que se ilustra mediante los siguientes diagramas:



Nodos de la placa enumerados en dos dimensiones



Nodos de la placa enumerados en una dimensión

De esta manera, podemos pensar en el vector  $u^k \in \mathbb{R}^{m*n}$ , formado por los elementos  $u_1^k, u_2^k, \dots, u_{m*n}^k$  como la temperatura de todos los nodos de la placa en el instante  $k$ . Podemos extender esto para el tiempo y plantear la matriz  $U \in \mathbb{R}^{m*n \times l}$  como la matriz formada por todos los vectores  $u^1, u^2, \dots, u^l$ , la cual representa el estado de la placa durante cada iteración del modelo, siendo  $n$  el número total de nodos horizontales,  $m$  el número total de nodos verticales, y  $l$  el número total de iteraciones.

## Condiciones de borde

La segunda consideración que debemos hacer para este modelo es lo que ocurre en los bordes de la placa. Las ecuaciones planteadas anteriormente nos permiten conocer la temperatura en los nodos interiores. Sin embargo, esto no quiere decir que la temperatura de los nodos en las fronteras de la placa vaya a evolucionar de la misma forma, debido a que se encuentran en condiciones diferentes. Por lo tanto, debemos plantear también cuatro ecuaciones, una para cada frontera, con el objetivo de que los datos obtenidos mediante el modelo sean más cercanos a la realidad:

- I. **Ingreso de temperatura:** Al inicio del proceso, la placa se encontrará a temperatura ambiente, e irá ingresando calor a lo largo de la recta  $x = 0$ . El calor que ingresa varía según el instante que se analice, por lo que estará dado por una función  $f: U \subseteq \mathbb{R} \rightarrow \mathbb{R}$ , cuyo parámetro será el instante de tiempo  $t$ . Por lo tanto, la condición (I) será:

$$u = f(t) \text{ en } x = 0$$

- II. **Conservación de calor en el extremo horizontal:** Dado que la longitud horizontal es mayor que la vertical, consideramos que la pérdida de calor hacia el exterior por la frontera  $x = L$  es despreciable y no la tendremos en cuenta. Entonces, la condición (II) será:

$$\frac{\partial u}{\partial \eta} = 0 \text{ en } x = L$$

- III. **Simetría vertical:** Si consideramos que los extremos verticales de la placa se encuentran en  $y = H/2$  y en  $Y = -H/2$  podemos establecer una simetría entre ambas mitades verticales a fin de simplificar el modelo al sólo considerar el segmento comprendido entre  $y = 0$  e  $y = H/2$ . Para ello, imponemos una restricción en la que no haya transferencia de calor entre ambas secciones. De esta forma, la condición (III) será:

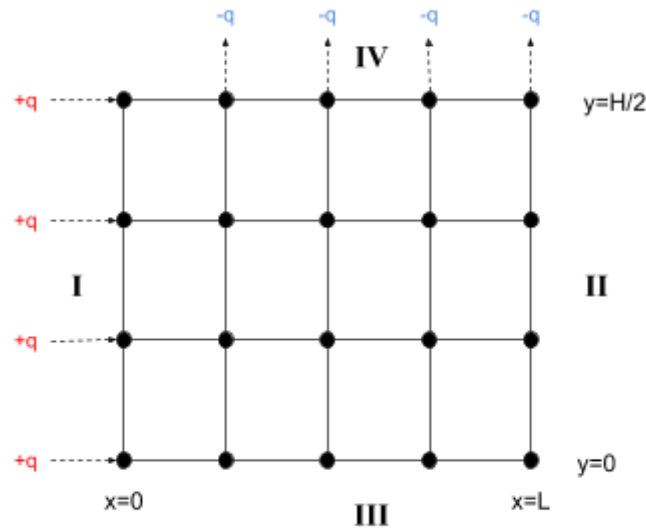


$$\frac{\partial u}{\partial \eta} = 0 \text{ en } y = 0$$

**IV. Pérdida de calor hacia el exterior:** En contraparte a lo mencionado en la condición (II), debemos tener en cuenta la salida de calor a través del borde  $y = H/2$ . Así, la condición (IV) será:

$$\frac{\partial u}{\partial \eta} - \frac{h_c}{\kappa}(u - T_a) = 0 \text{ en } y = H/2$$

Si observamos la placa con estas condiciones, esta quedará:



## Forma matricial

Al conocer ya las condiciones de borde, podemos pensar el cálculo de la temperatura en un instante  $k+1$  como el producto de matrices  $u^{k+1} = M * u^k + f(k)$ , siendo  $f(k)$  el vector que contiene los valores de la función de temperatura en  $x = 0$ . Debido a esto, sólo queda encontrar la matriz  $M$  para poder llevar a cabo las operaciones.

Para facilitar el diseño de la matriz, definimos  $a = 1 - 4 * r$ . Si consideramos, por ejemplo, una aproximación de la placa con 4 filas y 5 columnas de nodos,  $M$  será:



## Análisis del error

Por distintas razones, este modelo presenta un error respecto a la solución analítica de la situación.

Primero debido a que trabajamos con el método de las diferencias finitas, asignándole a cada segmento de la placa una temperatura homogénea siendo que esto no es necesariamente cierto. Algo similar ocurre con el tiempo, ya que consideramos las iteraciones como saltos en el intervalo temporal analizado. Además, tenemos el error adicional de trabajar con aproximaciones de las derivadas.

Podemos, sin embargo, hacer una aproximación del error de cada iteración a fin de ver cómo podemos reducirlo. Si consideramos a  $A$  como la solución analítica,  $N$  como la solución numérica obtenida por nuestro modelo matemático, y  $\varepsilon$  como el error de la solución numérica, entonces queda:

$$\begin{aligned}A &= N + \varepsilon \\ \varepsilon &= A - N \\ \varepsilon &= A - M * u^k\end{aligned}$$

Podemos ver que el error crece si el producto  $M * u^k$  es muy elevado, por lo que podemos pensar que para reducir el error tenemos que evitar que este producto crezca demasiado. Podemos lograr esto al observar los autovalores de  $M$ , ya que si estos son menores que 1, el resultado será convergente.

Esto puede presentar un problema, ya que encontrar los autovalores de  $M$  cuando esta es muy amplia resulta en enormes cantidades de cálculos, lo cual consume mucho tiempo y recursos.

Sin embargo, podemos recurrir al teorema de Gerschgorin para evitar tener que calcular estos valores. Este teorema dice que los autovalores de una matriz se encuentran dentro de la unión de circunferencias, cada una de las cuales tiene centro en el elemento de la diagonal principal de cada fila y radio igual a la sumatoria de los elementos restantes de dicha fila. De esta forma podemos asegurarnos que el cálculo de la siguiente iteración será convergente siempre que esas circunferencias tengan centro cercano al origen y radio menor que 1.

Si analizamos la matriz propuesta en la sección anterior, vemos que existen 5 casos para la aproximación de los autovalores mediante este teorema:

1. **Centro en 0 y radio = 0:** Esto ocurre en las filas correspondientes a la restricción (I), como por ejemplo en la fila 1. Este caso no representa ningún problema.
2. **Centro en  $r$  y radio =  $(1 - r)$ :** Esto ocurre mayormente en las filas correspondientes a las restricciones (II) y (III). Si planteamos la inecuación  $|1 - r| > 1$  correspondiente al radio, llegamos a que  $r > 0$ . De esta manera tenemos el mínimo valor que puede tomar  $r$ .
3. **Centro en 0 y radio = 1:** Este caso sólo se presenta en la fila 5 de la matriz. Al igual que el caso 1, este no presenta ningún problema.
4. **Centro en  $(1 - 4*r)$  y radio =  $(4*r)$ :** Esta situación se da en la mayoría de las filas. Si planteamos la inecuación  $1 - 4*r \geq 0$ , llegamos a que  $r \leq 1/4$ . Con esto obtenemos que el máximo valor que puede tomar  $r$  es  $1/4$ , ya que así la circunferencia tendrá centro 0 y radio 1.
5. **Centro en  $(1 - 4*r)$  y radio =  $(3*r)$ :** Este caso se da en las filas correspondientes a la restricción (IV) sin modificar la matriz M. Esto también resulta en que  $r \leq 1/4$ , con la diferencia que en este caso el radio máximo será de  $3/4$ .

Habiendo hecho este análisis, podemos concluir que el producto  $M * u^k$  será convergente siempre que  $r \in (0; 1/4)$ .

Como mencionamos durante el desarrollo de la ecuación (4), consideramos a  $r$  como:

$$r = \frac{\kappa * \Delta_t}{\rho * C * \Delta_x^2}$$

En donde  $\kappa$ ,  $\rho$ , y  $C$  son constantes físicas correspondientes al material, de modo que no podemos hacerlos variar sin tener que cambiar de placa. Sin embargo,  $\Delta_t$  y  $\Delta_x$  son valores elegidos arbitrariamente para el modelo matemático, indicando los saltos en el período temporal y el espacio de la placa respectivamente. Entonces, si buscamos reducir el error lo más posible, debemos asignarles valores para que  $r$  se encuentre en el rango mencionado anteriormente.

# Implementación en MATLAB

Para poder poner en práctica el modelo matemático desarrollado a lo largo del trabajo, programaremos un script en MATLAB versión R2017b que nos permitirá obtener la evolución del calor en la placa a lo largo del tiempo al pasarle las constantes del problema, así como generar una animación que nos permite visualizar la evolución de la temperatura en la placa. El código de este script se encuentra subido en un [repositorio de GitHub](#).

## Datos iniciales

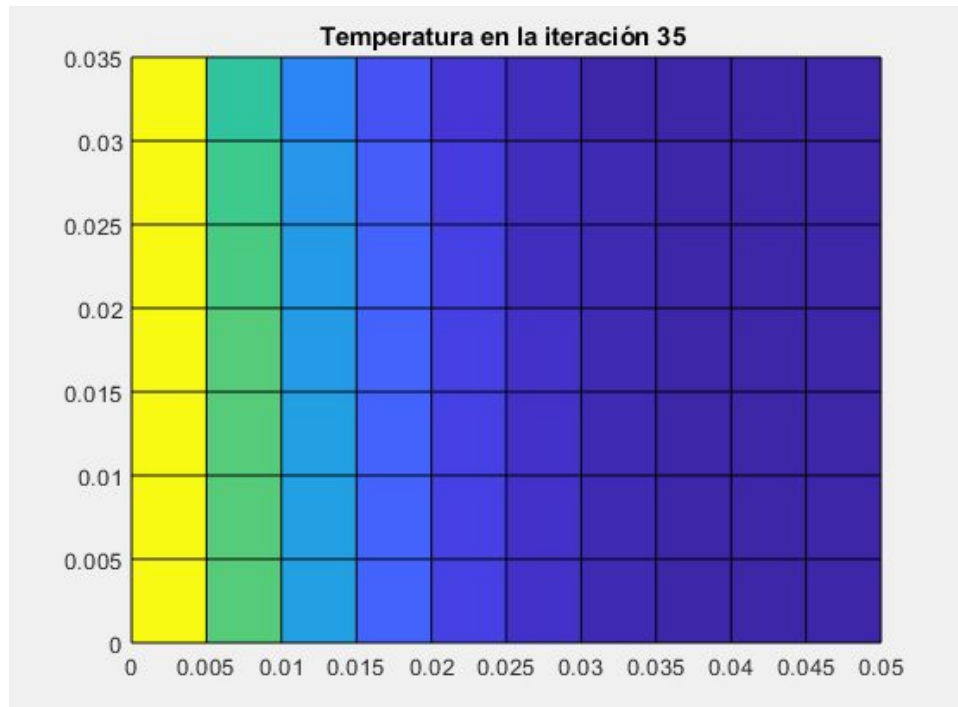
Para la ejecución de este script, otorgamos a las constantes los siguientes valores:

- Conductividad térmica:  $\kappa = 204,3 \text{ (J/m} \cdot \text{s)}$
- Calor específico:  $C = 910 \text{ (J/Kg} \cdot \text{K)}$
- Densidad:  $\rho = 2700 \text{ (Kg/m}^3\text{)}$
- Transmitancia térmica:  $h_c = 200 \text{ (W/m}^2 \cdot \text{K)}$
- Longitud:  $L = 0,06 \text{ (m)}$
- Altura:  $H = 0,04 \text{ (m)}$
- Delta x:  $\Delta_x = 0,005 \text{ (m)}$
- Tiempo de evaluación:  $t_{fin} = 20 \text{ (s)}$
- Delta t:  $\Delta_t = 0,05 \text{ (s)}$
- Temperatura del ambiente:  $T_{ambiente} = 293,15 \text{ (K)}$
- Temperatura máxima de la superficie:  $T_{superficie} = 343,15 \text{ (K)}$
- Función de entrada de calor:  
$$f(t) = T_{ambiente} + (T_{superficie} - T_{ambiente}) * \sin(t / (10 * \pi)) \text{ (K)}$$

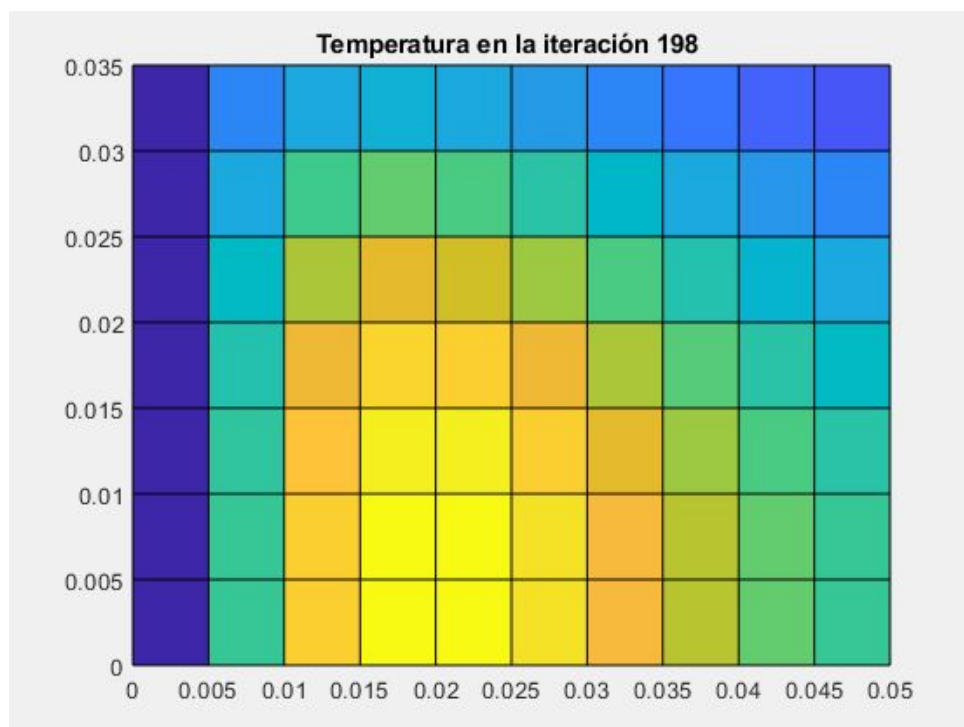
## Ejecución

A continuación ejecutaremos el código del programa, adjuntando capturas de pantalla sobre la animación de la evolución de la placa y explicando brevemente lo que ocurre. Podemos distinguir distintas etapas en la ejecución del script:

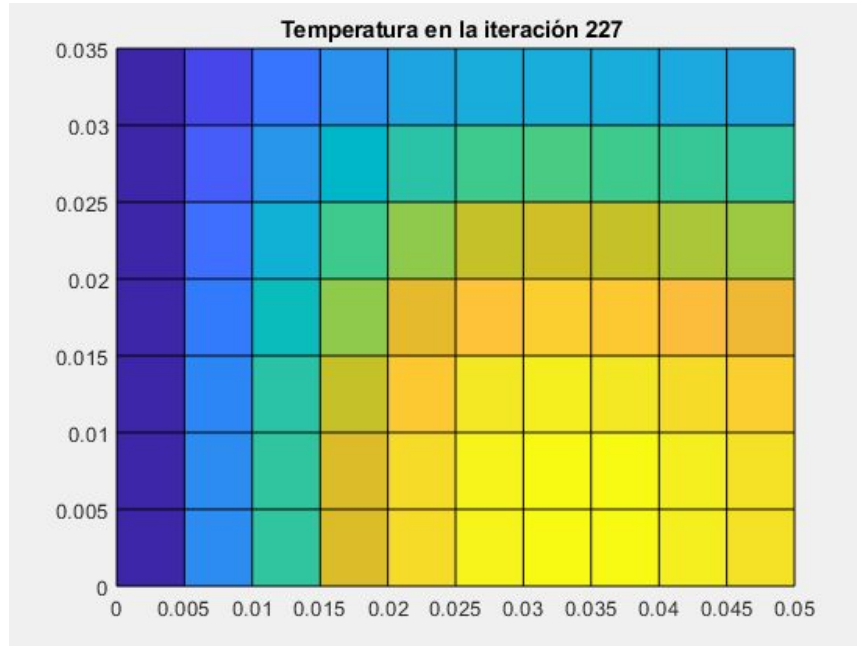
- **Calentamiento:** Al inicio de la ejecución, la placa se encuentra a temperatura ambiente, pero podemos ver cómo se va calentando a medida que aumenta la temperatura en la frontera izquierda debido a la condición (I).



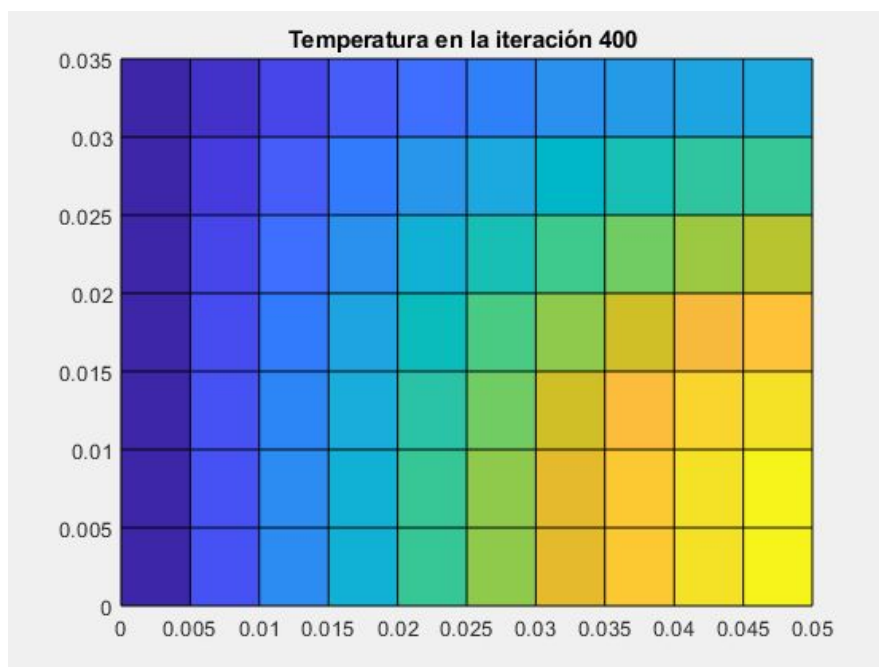
- **Fin del ingreso de calor:** Aquí vemos cómo la función  $f$  deja de introducir calor, por lo que la zona de mayor temperatura comienza a desplazarse hacia el medio de la placa. También podemos ver cómo la parte superior se encuentra más fría, lo cual tiene sentido debido a la condición (IV).



- **Enfriamiento del lado izquierdo:** Conforme continúa la ejecución, podemos notar que los nodos más cercanos a la izquierda comienzan a enfriarse por lo que el foco de concentración de calor se desplaza aún más hacia abajo a la derecha debido a las condiciones (II) y (III).



- **Fin de la ejecución:** Al terminar de correr el script, vemos cómo el foco de calor se ha desplazado hacia la esquina inferior derecha, lo cual tiene sentido debido a que es la sección con menor flujo de temperatura, mientras que el resto de la placa se encuentra más fría debido a la condición (IV).



## Conclusión

Pese a tratarse de una situación completamente ficticia debido a las dimensiones de la placa, consideramos que el modelo desarrollado realiza un buen cálculo de la temperatura.

Por otra parte, cabe destacar que estos resultados no son 100% correctos debido a lo mencionado durante el análisis del error y a las aproximaciones realizadas por la propia computadora, ya que resulta imposible para esta calcular funciones no polinómicas sobre intervalos continuos.

También merece la pena mencionar que el modelo actual resulta mucho más óptimo que versiones anteriores, gracias al uso del producto matricial para el cálculo de la temperatura en lugar de llamadas a ciclos. Esto se debe a que las operaciones se realizan en paralelo, en lugar de ejecutar todos los cálculos de manera secuencial en un solo hilo, lo cual resulta en un menor tiempo de construcción de la matriz  $U$ .



## Bibliografía

- La ecuación del calor en 2D - O.R. Faure, 2019
- Ecuaciones en derivadas parciales, versión 1.0 - O.R. Faure & V.C. Rougier, 2011
- Aproximaciones en Diferencias, versión 1.0 - O.R. Faure & V.C. Rougier, 2010
- Solución Numérica de Ecuaciones en Derivadas Parciales mediante el Método de Elementos Finitos - O.R. Faure, 2019
- Finite Difference Methods in 2d Heat Transfer - V. Vuorinen, 2016
- Lecture 2: Finite Difference Methods in Heat Transfer - V. Vuorinen, 2016