

Proyecto 3

1 Objetivo general

Al finalizar el proyecto, el estudiante habrá aplicado los conocimientos obtenidos en lecciones del curso para implementar un método de solución de ecuaciones diferenciales parciales, particularmente en el caso de propagación de calor.

2 Objetivos específicos

1. Utilizar programación en C/C++ para implementar estructuras de datos y algoritmos relacionados con métodos numéricos para solucionar ecuaciones diferenciales parciales.
2. Emplear la biblioteca OpenMP para paralelizar la ejecución de los algoritmos numéricos.

3 Procedimiento

El objetivo del proyecto es estimar la distribución de temperatura en una placa delgada, rectangular, aislada excepto en sus bordes, donde cada uno de ellos está en contacto con una masa con un perfil térmico dado por el usuario. Además, el usuario podrá seleccionar si alguno de los bordes está aislado.

Para ello debe utilizarse el método de Liebmann para la solución de la ecuación diferencial parcial elíptica, en una estructura “piramidal”, tal y como se explicó en clase: primero se resuelve el problema usando un único elemento, promediando las condiciones de frontera; luego, se divide el área en 4 secciones para, a partir de la inicialización del primer paso, proseguir con el método de Liebmann; así, de forma iterativa se utiliza el método de Liebmann hasta que converja, para pasar a un siguiente tamaño, moviéndose en una especie de pirámide de la cúspide hacia abajo. La idea es que cada capa calculada sirva como inicialización de la siguiente capa de mayor tamaño, favoreciendo una convergencia más rápida.

En su implementación deberá utilizar además la biblioteca **OpenMP** para reducir el tiempo de solución por medio de paralelización del algoritmo. Debe notar que la paralelización debe ser diseñada cuidadosamente, puesto que, de otro modo el costo de paralelización (*overhead*) puede ser mayor que la ganancia de utilizar múltiples hilos. Use la regla empírica de que cada hilo deberá resolver una tarea relativamente compleja, y debe evitarse la creación de hilos para tareas cortas. Observe que solo tiene sentido paralelizar el

metodo de Liebmann para una “capa” de la pirámide, cuando esta capa es suficientemente grande.

El perfil de temperatura resultante debe mostrarse como en la figura 1, donde la escala de colores se extiende/comprime para cubrir todo rango de temperaturas encontradas en la solución. Debe mostrar la escala de temperaturas utilizada.

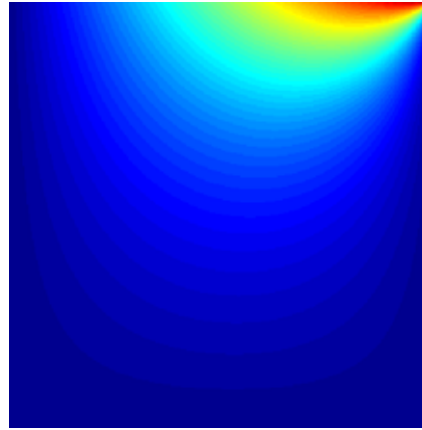


Figura 1: Imagen mostrando el perfil térmico

Además, el usuario deberá indicar si adicionalmente quiere calcular el flujo de calor como variable secundaria, que es un campo vectorial. En caso de que el flujo de calor sea calculado y la visualización esté activa, sobreponga sobre su perfil térmico pequeñas flechas sobre una rejilla que ilustren la solución. Note que no es posible poner todos los vectores calculados en la visualización, así que debe seleccionar un tamaño de rejilla razonable para submuestrear la solución calculada para su visualización. El resultado debe ser similar a la función `quiver` de GNU/Octave.

El programa debe poder ejecutarse desde consola, y el usuario debe poder anular cualquier salida gráfica con parámetros correspondientes de la aplicación para así poder medir el tiempo de ejecución. Alguna opción de consola debe permitir medir ese tiempo de ejecución con suficiente precisión.

Debe documentar los encabezados de su código utilizando la sintaxis de Doxygen.

4 Interfaz

Utilice la biblioteca `program_options` de boost, para interpretar *al menos* la siguiente lista de opciones:

- `--help` Imprime esta lista de opciones
- `-t <val>` Indica temperatura en borde superior
- `-b <val>` Indica temperatura en borde inferior
- `-l <val>` Indica temperatura en borde izquierdo
- `-d <val>` Indica temperatura en borde derecho

- i [tblr] Aísla los bordes indicados (t=arriba, b=abajo, l=izquierda, r=derecha). El usuario debe indicar uno o más bordes junto a la opción -i. Si el usuario especifica -i y no indica ningún borde, el programa debe detenerse e indicar claramente el error.
- p <archivo> Indica el nombre del archivo con perfil térmico. El detalle del contenido del archivo de perfil térmico se brinda más abajo.
- h <val> Número de píxeles horizontales en la solución
- v <val> Número de píxeles verticales en la solución
- q Desactive toda forma de visualización. Si esta opción no está presente, el programa sí debe mostrar los resultados.
- f Active el cálculo de flujo de calor.
- g <val> Tamaño de rejilla de visualización para flujo de calor. Este valor especifica cuántos píxeles cubre cada celda de la rejilla final. Por ejemplo, si se especifica 5, entonces tanto en x como en y se muestra una flecha del flujo de calor cada 5 píxeles.

El archivo de perfil térmico es un archivo de texto, que usted puede también utilizar por medio de boost `program_options`. El archivo simplemente contiene una lista de temperaturas que se van a utilizar de forma equiespaciada en cada borde.

El contenido tendrá la estructura

$$\begin{aligned} \langle \text{borde} \rangle &= \text{valor} \ [\text{valor}^*] \\ [\langle \text{borde} \rangle &= \text{valor} \ [\text{valor}^*]] \end{aligned}$$

donde $\text{borde} \in \{top, bottom, left, right\}$ y valor son números de temperatura que se distribuirán de forma equidistante en el borde. Por ejemplo,

$$\text{left} = 25$$

obliga a que el borde izquierdo tenga todo 25 grados, pero

$$\text{left} = 25 \ 100$$

hace que la temperatura aumente linealmente desde 25 grados centígrados hasta 100 grados en el borde izquierdo. Se deberá documentar bien cómo se interpretan esos números (de abajo hacia arriba o de arriba hacia abajo, si todos los bordes de igual modo, etc.). Para el caso en que se den más de 2 números, utilice trazadores cúbicos para hacer la interpolación.

Si el usuario especifica un archivo y temperaturas en los bordes, tiene prioridad lo especificado en la línea de comandos. De este modo, si el usuario tiene un archivo `perfil.txt` con el siguiente contenido:

```
top = 25 30 100
right = 100
bottom = 10 40 50 100
left = 25 1000 10
```

y el usuario llama al programa de consola

```
placa -t 50 -ib -f -p perfil.txt
```

entonces el perfil de temperatura del borde inferior en el archivo será ignorado, porque el usuario quiere ese borde aislado, y la temperatura del borde superior será homogénea en 50 grados. Los bordes derecho e izquierdo seguirán el perfil especificado en el archivo.

En caso de que ni en la línea de comandos ni en el archivo de texto se haya especificado una temperatura para alguno de los bordes, este deberá considerarse como aislado.

5 Entregables

1. Código fuente documentado, subido al tecDigital como se indica en el programa del curso.
2. Debe agregar un archivo README con las instrucciones de compilación y ejecución.
3. Agregue scripts de bash para llamar a su programa con todos los parámetros necesarios por defecto, para facilitar así la ejecución de pruebas.
4. Documentación del código en formato html, generada automáticamente del código con Doxygen.
5. Demostración del sistema funcionando.
6. Artículo corto de no más de 3 páginas, donde se evalúe el desempeño de su implementación con y sin uso de OpenMP. Siga las recomendaciones de cómo elaborar un artículo formal en el [sitio del curso](#).