

# 1 Modelo y justificación

## El Problema

El computo ha sido una de las herramientas más poderosas de la humanidad durante el último siglo habiéndose incrustado prácticamente todo aspecto de la vida humana; el entretenimiento, las comunicaciones, la infraestructura y el guardado de todo tipo de información así como el apoyo a disciplinas científicas y tecnológicas. El poder de los equipos de computo aumenta generación tras generación siendo el desarrollo de los CPU una parte de suma importancia en este avance, sin embargo siempre se trabaja dentro de las limitantes físicas por lo que al exigir al CPU una parte importante de la energía se convierte en calor, de ello surge la necesidad de un componente de refrigeración eficiente que mantenga al procesador dentro del espectro que permite un funcionamiento adecuado, estudiar las condiciones necesarias para que el CPU no llegue a un suicidio térmico es un problema de interés.

## Modelado

Consideremos un procesador de forma cuadrada; cuyo interior será representado por  $\Omega = (0, 1) \times (0, 1)$  por simplicidad, con  $\xi \in \mathbb{N}$  núcleos en su interior denotemos  $\tau_n$  al  $n$ -ésimo núcleo, así para  $n \in \{1, \dots, \xi\}$  definimos  $g_n : \Omega \times \mathbb{R} \rightarrow \mathbb{R}$  con  $g((x, y), t) = g(x, y, t)$  el calor generado en el punto  $(x, y) \in \Omega$  al tiempo  $t$ , denotamos  $G(x, y, t) = \sum_{n=1}^{\xi} g_n(x, y, t)$ .

Se modelará el sistema mediante la ecuación de calor

$$u_t = \alpha(u_{xx} + u_{yy}) + G \quad (\text{en } \Omega)$$

Supongamos que la refrigeración no se aplica directamente sobre todo el procesador sino solo en la frontera (que sea el aire en el que se encuentra el cual es refrigerado por un ventilador por ejemplo), así tenemos las condiciones de frontera de Robin

$$-\kappa \frac{\partial u}{\partial n} = \eta(u - T_{out}) \quad (\text{en } \partial\Omega)$$

$\kappa$  : Conductividad térmica del material

$\eta$  : Coeficiente de convección (tasa de transferencia térmica entre un material y un fluido)

Considerando lo dicho en [1] respecto al material y las condiciones comunes de una procesador haciendo el cambio de unidades correspondiente se fijará  $\alpha = 9 \times 10^{-5}$ ,  $\kappa = 152.24$ ,  $\eta = 180$ ,  $T_{out} = 16$ .

Se considera que en un inicio las el CPU se encuentra a temperatura ambiente  $u_0(x, y) = 16$ .

## 2 Ambiente de computo

La implementación computacional se llevó a cabo en lenguaje C utilizando Open MPI.

## Método de líneas

Se usó el método de líneas para la solución de la ecuación, estos es aproximar las derivadas espacial por medio de diferencias finitas centradas de forma que obtenemos una ecuación diferencial ordinaria (EDO)!para cada punto del dominio discretizado, posteriormente se utilizó Ruhen Kuta de orden 4 para resolver este conjunto de sistemas de ordinarias. La condición de Robin para ka frontera fue implementado utilizando nodos fantasma junto a los puntos frontera.

## Paralelización

Como se menciona anteriormente se usó MPI en la implementación del solver particularmente fue usado para dividir el dominio de solución en subdominios  $\Omega_1, \Omega_2, \dots, \Omega_n \in \Omega$  de forma que cada uno de estos subdominios  $\Omega_j$  sea asignado a un proceso  $\mathcal{P}_j$ . Sin embargo para determinar las EDO es necesario conocer los puntos vecinos de un punto interior, ¿que ocurre con aquellos en la frontera de los  $\Omega_i$ ? Hay dos casos.

- $x_{i,j} \in \partial\Omega_i \subset \partial\Omega$ :  
Se se usa nodo fantasma fuera de ka frontera.
- $x_{i,j} \in \partial\Omega_i \subset \text{Int}(\Omega)$ :  
Entonces hay algún vecino de  $x_{i,j}$  en algún otro subdominio  $\Omega_j$ ,  $i \neq j$  que es trabajado por otro proceso.

Para el segundo caso y por el paradigma de memoria distribuida de MPI se utiliza el comunicador cartesiano (suponemos  $\Omega_i$  también son rectángulos) para coordinar el intercambio de información y trabajo necesario.

Para cada subdominio discretizado se tiene una cubierta denominada *halo*, este servirá para guardar valores necesarios para plantear las EDO pero que no corresponden a puntos de  $\Omega_i$ , pueden ser los nodos fantasma para las condiciones de Robin o pueden ser valores correspondientes a subdominios vecinos.

## Rendimiento

Para una malla de  $220 \times 220$  es decir 48,400 puntos y de  $t_0 = 0$  a  $t_{final} = 1800$  (media hora) y  $\Delta t = 30$  se tienen las siguientes mejoras en rendimiento respecto al código secuencial:

Procesos	Speed Up
1	1
2	
4	
8	

### 3 Referencias

- [1] C.-T. Chen, C.-K. Wu, and C. Hwang, “Optimal design and control of cpu heat sink processes,” *IEEE Transactions on Components and Packaging Technologies*, vol. 31, no. 1, pp. 184–195, 2008.