TRABAJO FIN DE GRADO: Implementación del método de diferencias finitas en el dominio del tiempo en GPU

PRESENTADO POR:

JUAN JOSÉ SALAZAR LÓPEZ

ÍNDICE

INTRODUCCIÓN

-¿Cuál es le método de las diferencias finitas en el dominio del tiempo?

INTRODUCCIÓN

-¿Cuál es le método de las diferencias finitas en el dominio del tiempo?

-Aplicaciones del método de diferencias finitas en el dominio del tiempo

INTRODUCCIÓN

-¿Qué es el método de las diferencias finitas en el dominio del tiempo?

-Aplicaciones del método.

-¿Por qué implementarlo en GPU?

INTRODUCCIÓN

OBJETIVOS:

-Generar dos algoritmos, uno en CPU y otro en GPU, donde se obtengan los mismos resultados.

-Comparar con ellos el tiempo de cálculo al utilizar cada componente.

MÉTODO DE LAS DIFERENCIAS FINITAS EN EL DOMINIO DEL TIEMPO

Método de la diferencia central finita:

Desarrollo en serie de Taylor de una función y(x) en los puntos (x+h), (x-h)

$$y(x+h) = y(x) + hy'(x) + \frac{h^3}{2}y''(x) + \frac{h^3}{6}y'''(x)$$

$$y(x-h) = y(x) - hy'(x) + \frac{h^3}{2}y''(x) - \frac{h^3}{6}y'''(x)$$

MÉTODO DE LAS DIFERENCIAS FINITAS EN EL DOMINIO DEL TIEMPO

Método de la diferencia central finita:

Desarrollo en serie de Taylor de una función y(x) en los puntos (x+h), (x-h)

$$y(x+h) = y(x) + hy'(x) + \frac{h^3}{2}y''(x) + \frac{h^3}{6}y'''(x)$$

$$y(x-h) = y(x) - hy'(x) + \frac{h^3}{2}y''(x) - \frac{h^3}{6}y'''(x)$$

$$y'(x) = \frac{y(x+h) - y(x-h)}{2h} + O(h^2)$$

MÉTODO DE LAS DIFERENCIAS FINITAS EN EL DOMINIO DEL TIEMPO

Ecuaciones rotacionales de Maxwell para el campo electromagnético:

$$\frac{\partial D}{\partial t} = \nabla \times H$$

$$\frac{\partial H}{\partial t} = -\frac{1}{\mu_0} \nabla \times E$$

$$D(\omega) = \varepsilon_0 \varepsilon_r^*(\omega) E(\omega)$$

MÉTODO DE LAS DIFERENCIAS FINITAS EN EL DOMINIO DEL TIEMPO

Ecuaciones rotacionales de Maxwell para el campo electromagnético:

$$\frac{\partial D}{\partial t} = \nabla \times H$$

$$\frac{\partial H}{\partial t} = -\frac{1}{\mu_0}\nabla \times E$$

$$D(\omega) = \varepsilon_0 \varepsilon_r^*(\omega) E(\omega)$$

Modo transversal magnético

$$\frac{\partial D_z}{\partial t} = \frac{1}{\sqrt{\varepsilon_0 \mu_0}}\left(\frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y}\right)$$

$$D(\omega) = \varepsilon_r^*(\omega) E(\omega)$$

$$\frac{\partial H_x}{\partial t} = -\frac{1}{\sqrt{\varepsilon_0 \mu_0}}\frac{\partial E_z}{\partial y}$$

$$\frac{\partial H_y}{\partial t} = \frac{1}{\sqrt{\varepsilon_0 \mu_0}}\frac{\partial E_z}{\partial x}$$

MÉTODO DE LAS DIFERENCIAS FINITAS EN EL DOMINIO DEL TIEMPO

Modo transversal magnético

$$\frac{\partial D_z}{\partial t} = \frac{1}{\sqrt{\varepsilon_0 \mu_0}}\left(\frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y}\right)$$

$$D(\omega) = \varepsilon_r^*(\omega) E(\omega)$$

$$\frac{\partial H_x}{\partial t} = -\frac{1}{\sqrt{\varepsilon_0 \mu_0}} \frac{\partial E_z}{\partial y}$$

$$\frac{\partial H_y}{\partial t} = \frac{1}{\sqrt{\varepsilon_0 \mu_0}} \frac{\partial E_z}{\partial x}$$

Modo transversal magnético al aplicar el método de la diferencia central finita

$$\frac{D_z^{n+1/2}(i,j) - D_z^{n-1/2}(i,j)}{\Delta t} = \frac{1}{\sqrt{\varepsilon_0 \mu_0}}\left[\frac{H_y^n(i+\frac{1}{2},j) - H_y^n(i-\frac{1}{2},j)}{\Delta x}\right] - \frac{1}{\sqrt{\varepsilon_0 \mu_0}}\left[\frac{H_x^n(i,j+\frac{1}{2}) - H_x^n(i,j-\frac{1}{2})}{\Delta x}\right]$$

$$\frac{H_x^{n+1}(i,j+\frac{1}{2}) - H_x^n(i,j+\frac{1}{2})}{\Delta t} = -\frac{1}{\sqrt{\varepsilon_0 \mu_0}}\left[\frac{E_z^{n+1/2}(i,j+1) - E_z^{n+1/2}(i,j)}{\Delta x}\right]$$

$$\frac{H_y^{n+1}(i+\frac{1}{2},j) - H_y^n(i+\frac{1}{2},j)}{\Delta t} = \frac{1}{\sqrt{\varepsilon_0 \mu_0}}\left[\frac{E_z^{n+1/2}(i+1,j) - E_z^{n+1/2}(i,j)}{\Delta x}\right]$$

MÉTODO DE LAS DIFERENCIAS FINITAS EN EL DOMINIO DEL TIEMPO

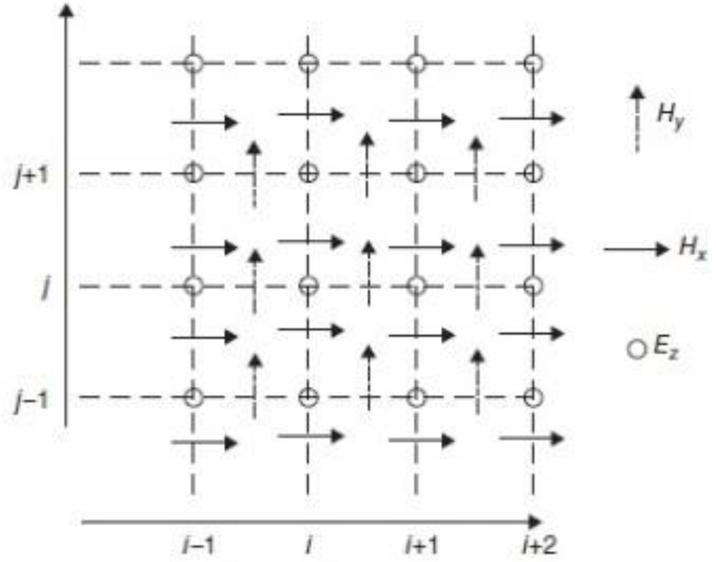
Modo transversal magnético

$$\frac{D_z^{n+1/2}(i,j)-D_z^{n-1/2}(i,j)}{\Delta t} = \frac{1}{\sqrt{\varepsilon_0\mu_0}}\left[\frac{H_y^n(i+\frac{1}{2},j)-H_y^n(i-\frac{1}{2},j)}{\Delta x}\right] - \frac{1}{\sqrt{\varepsilon_0\mu_0}}\left[\frac{H_x^n(i,j+\frac{1}{2})-H_x^n(i,j-\frac{1}{2})}{\Delta x}\right]$$

$$\frac{H_x^{n+1}(i,j+\frac{1}{2})-H_x^n(i,j+\frac{1}{2})}{\Delta t} = -\frac{1}{\sqrt{\varepsilon_0\mu_0}}\left[\frac{E_z^{n+1/2}(i,j+1)-E_z^{n+1/2}(i,j)}{\Delta x}\right]$$

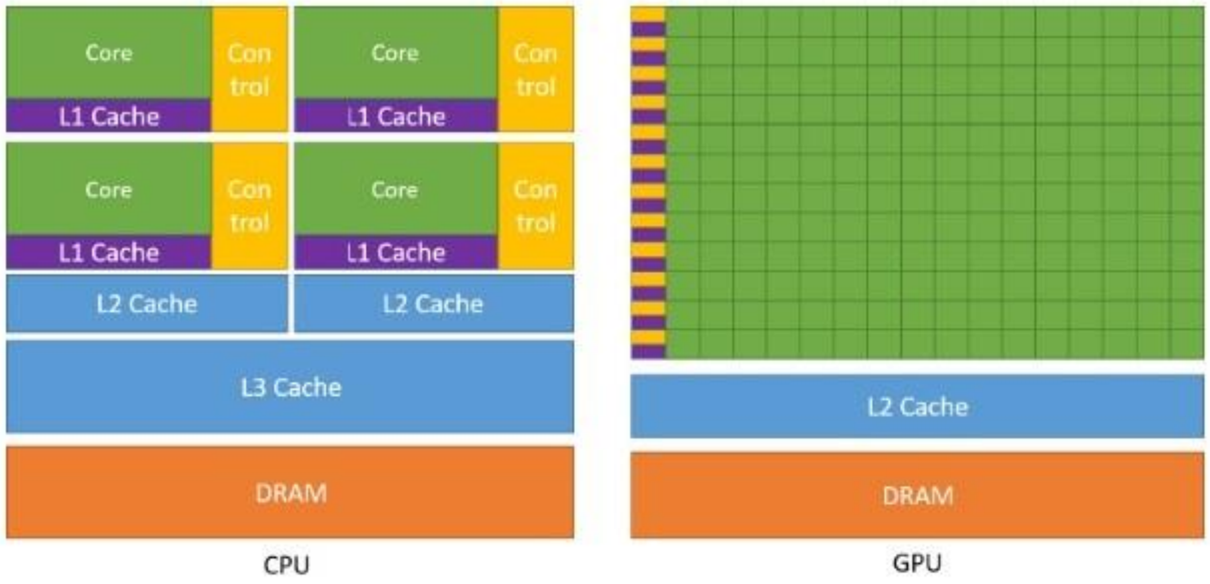
$$\frac{H_y^{n+1}(i+\frac{1}{2},j)-H_y^n(i+\frac{1}{2},j)}{\Delta t} = \frac{1}{\sqrt{\varepsilon_0\mu_0}}\left[\frac{E_z^{n+1/2}(i+1,j)-E_z^{n+1/2}(i,j)}{\Delta x}\right]$$

Campos intercalados en el espacio



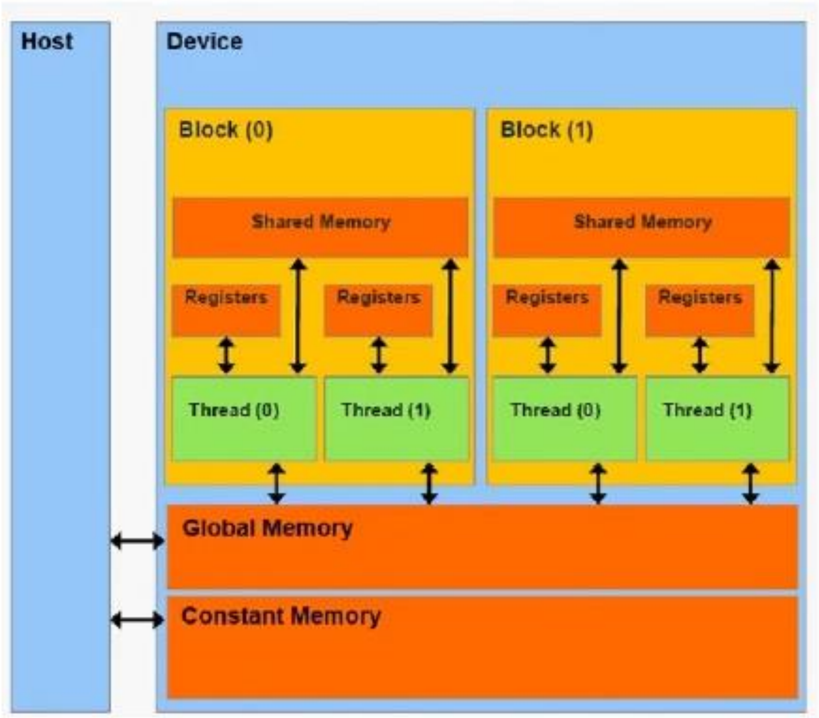
GPU

Diferencias GPU-CPU



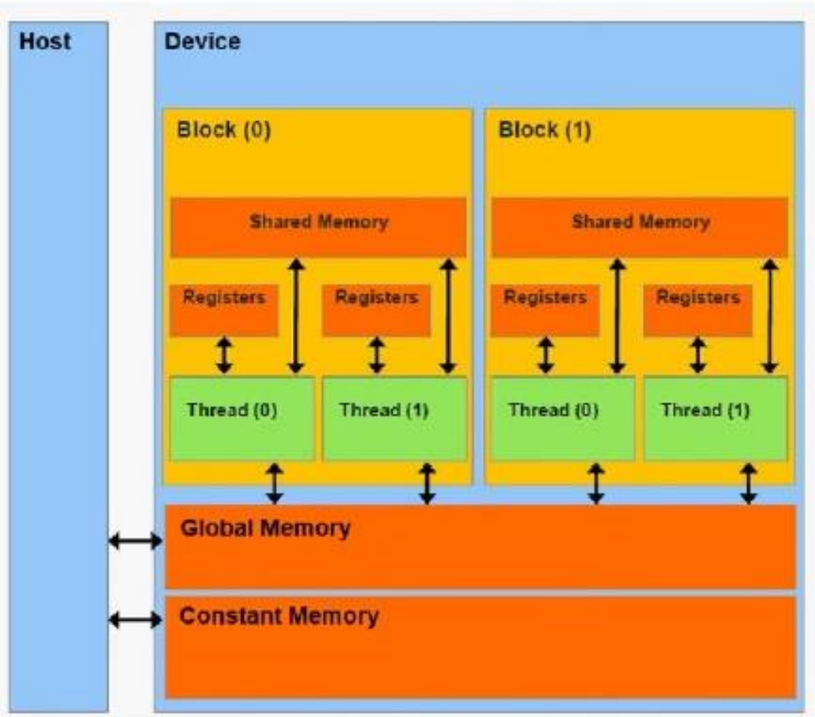
GPU

Estructura GPU

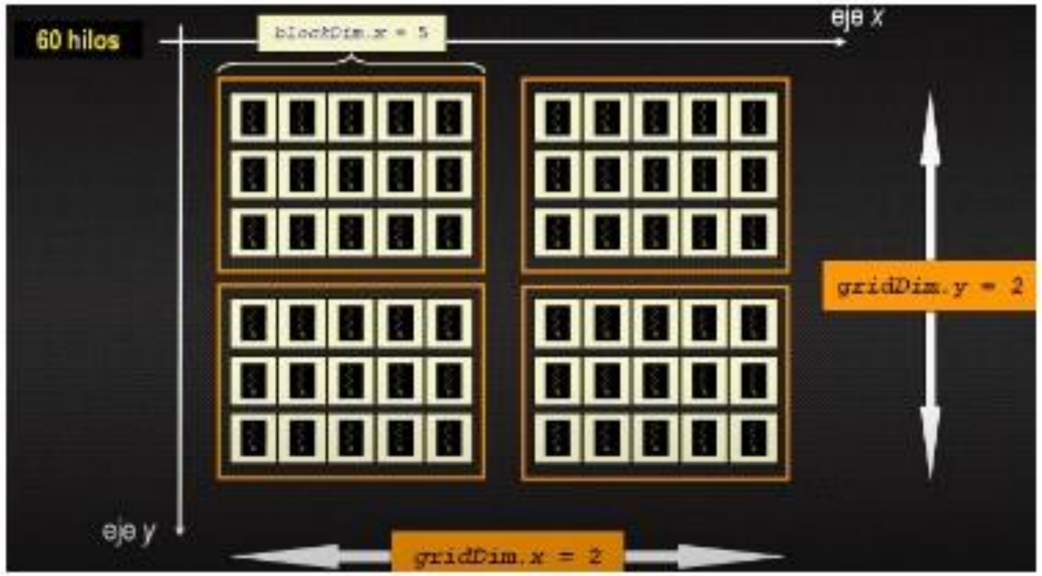


GPU

Estructura GPU

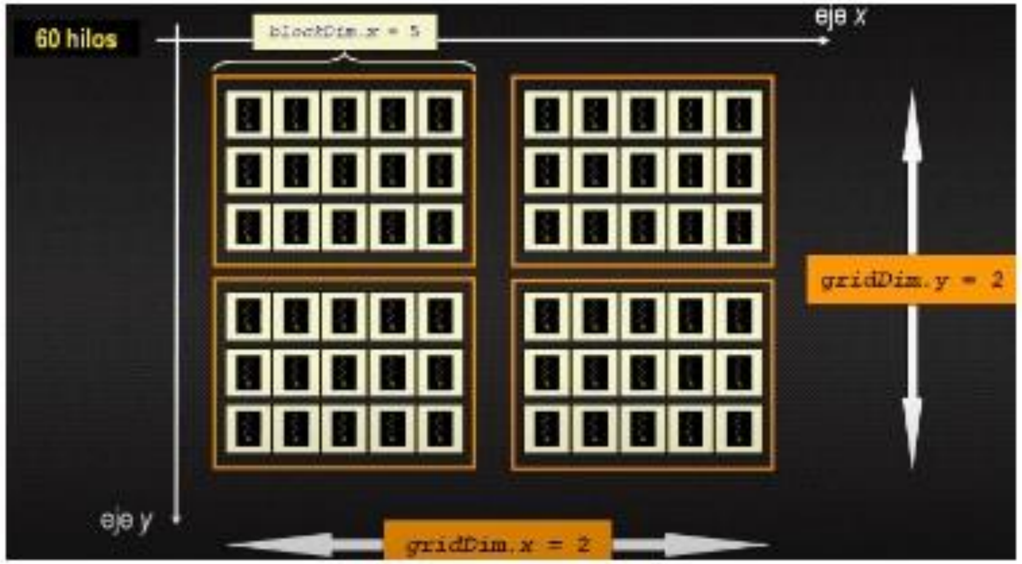


Estructura conceptual

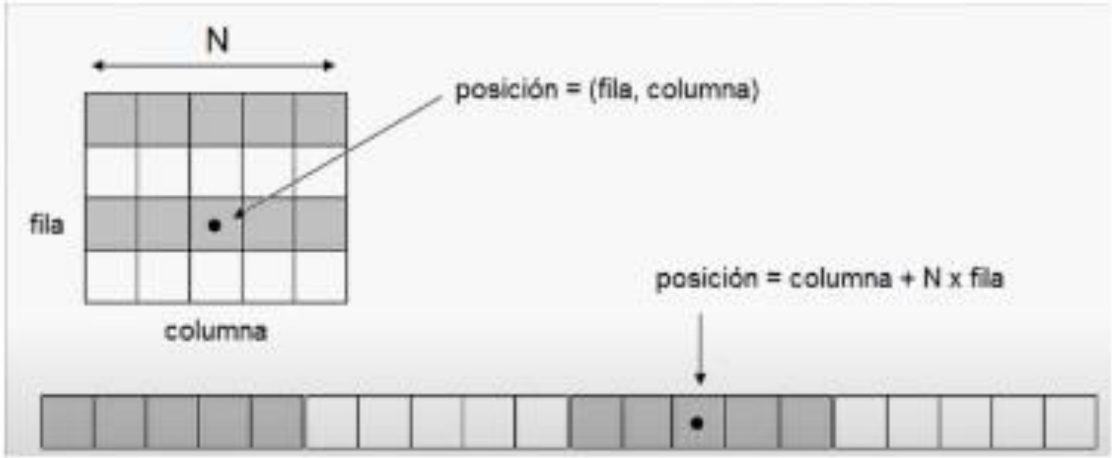


GPU

Estructura conceptual



¿Cómo localizar un hilo?



GPU

¿Cómo determinar la fila y la columna de un hilo?

```
int fil = blockIdx.y * blockDim.y + threadIdx.y;
int col = blockIdx.x * blockDim.x + threadIdx.x;
```

GPU

¿Cómo determinar la fila y la columna de un hilo?

```
int fil = blockIdx.y * blockDim.y + threadIdx.y;
int col = blockIdx.x * blockDim.x + threadIdx.x;
```

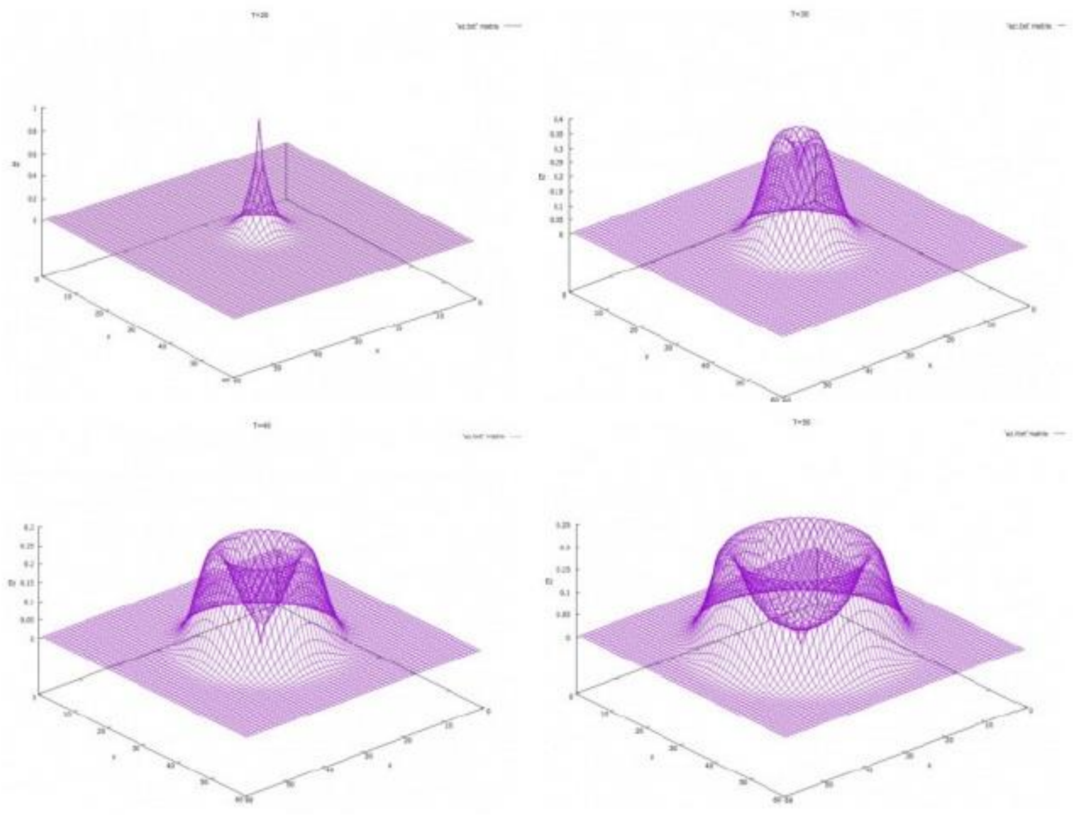
¿Cómo utilizar los hilos?

```
//Calculate Dz
if (0 < fil && fil < ydim && 0 < col && col < xdim) {
 field[fil * xdim + col].dz += 0.5 *
     (field[fil * xdim + col].hy - field[fil * xdim + col - 1].hy -
  field[fil * xdim + col].hx + field[(fil - 1) * xdim + col].hx);
}
__syncthreads();
```

Algoritmo

Tecnología utilizada:

RESULTADOS



RESULTADOS

Relación pasos temporales y velocidad de cómputo

Pasos Temporales	t_{CPU} (s)	t_{GPU} (s)	(t_{GPU}/t_{CPU})
10	1.971 ± 0.008	0.165 ± 0.001	11.91 ± 0.09
100	19.32 ± 0.05	1.633 ± 0.003	11.83 ± 0.04
1000	209.1 ± 1.4	16.397 ± 0.023	12.75 ± 0.09
10000	2150 ± 13	168.6 ± 0.9	12.75 ± 0.10

RESULTADOS

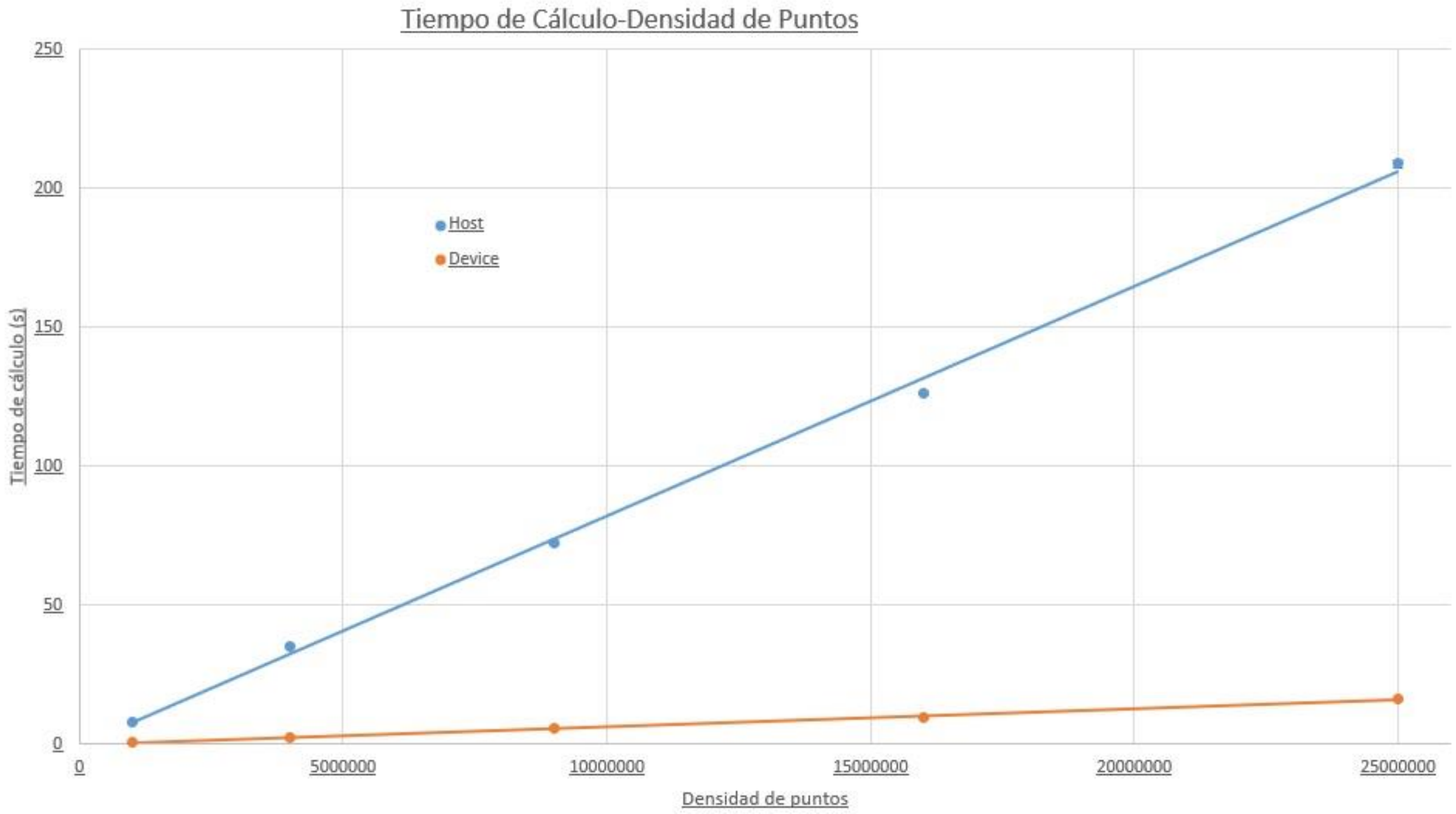
Tiempo de Cálculo-Pasos temporales
Host
Device
Tiempo de cálculo (s)
Densidad de puntos
2500
2000
1500
1000
500
0
0
1000
2000
3000
4000
5000
6000
7000
8000
9000
10000

RESULTADOS

Relación densidad de puntos y velocidad de cómputo

Densidad de puntos	t_{CPU} (s)	t_{GPU} (s)	$(t_{\mathrm{GPU}}/t_{\mathrm{CPU}})$
10^6	7.67 ± 0.04	0.640 ± 0.002	11.98 ± 0.07
$4 \cdot 10^6$	35.36 ± 0.23	2.447 ± 0.016	14.44 ± 0.13
$9 \cdot 10^6$	72.3 ± 0.3	5.582 ± 0.021	12.96 ± 0.07
$16 \cdot 10^6$	126.3 ± 0.5	9.81 ± 0.03	12.88 ± 0.06
$25 \cdot 10^6$	209.1 ± 1.4	16.39 ± 0.23	12.75 ± 0.09

RESULTADOS



Tiempo de Cálculo-Densidad de Puntos
250
200
150
100
50
0
Tiempo de cálculo (s)
Host
Device
0
5000000
10000000
15000000
20000000
25000000
Densidad de puntos

CONCLUSIONES

-Implementación de un algoritmo generado para CPU en GPU, obteniendo los mismos resultados.

-Mejora en la velocidad de cálculo de 12,6. Lo que se ejecuta en 600 s (10 horas) en CPU, en GPU se calcula en 48 minutos.

FIN

MUCHAS GRACIAS