

Implementación de Hilbert Curve Paralelizado con OMP

Joel Challco Tito
Estudiante de Ciencia de la Computación
Univesidad Católica San Pablo
Arequipa, Perú
Email: joel.challco@ucsp.edu.pe

I. INTRODUCCIÓN

La curva de Hilbert es una curva fractal continua que recubre el plano descrita inicialmente por el matemático alemán David Hilbert en 1891, como una variante de las curvas que recubren el plano descubiertas por Giuseppe Peano en 1890.

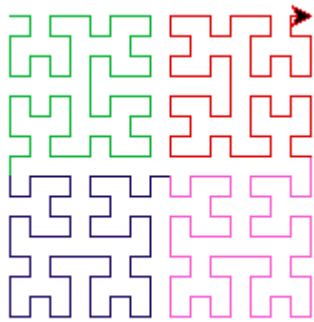


Figura 1. Hilbert curve, order 3

II. COMPARACIÓN

Para la comparación, se tomo la implementación de la curva de Hilbert recursiva y una implementación propia paralela.

```
void hilbert_curve_Parallel(int order, int angle)

// Iniciar hilos paralelos angulo de inicio a
#pragma omp parallel for
fprintf(out, "turtle.left(%d)\n", angle);

fprintf(out, "turtle.pencolor('#04B431')\n");
hilbert_curve(order, angle);
fprintf(out, "turtle.forward(%d)\n", STEP_SZ);
fprintf(out, "turtle.left(%d)\n", -angle);

fprintf(out, "turtle.pencolor('#210B61')\n");
hilbert_curve(order, -angle);

fprintf(out, "turtle.forward(%d)\n", STEP_SZ);

//mitad
fprintf(out, "turtle.pencolor('#FA58D0')\n");
hilbert_curve(order, -angle);

fprintf(out, "turtle.left(%d)\n", -angle);
fprintf(out, "turtle.forward(%d)\n", STEP_SZ);
```

```
fprintf(out, "turtle.pencolor('#DF0101')\n");
hilbert_curve(order, angle);

fprintf(out, "turtle.left(%d)\n", angle);
}
```

Esta implementación tiene las siguientes características:

- Se tomo como base la implementación recursiva.
- En la paralelización se dividió el espacio en cuatro, dándole la tarea a cada hilo una sección.
- se repite esta tarea hasta que lleguemos a nuestro caso base, orden 0.
- Se noto que mi implementación es mas lenta en comparación a la implementación recursiva.

El obtuvo el siguiente resultado.

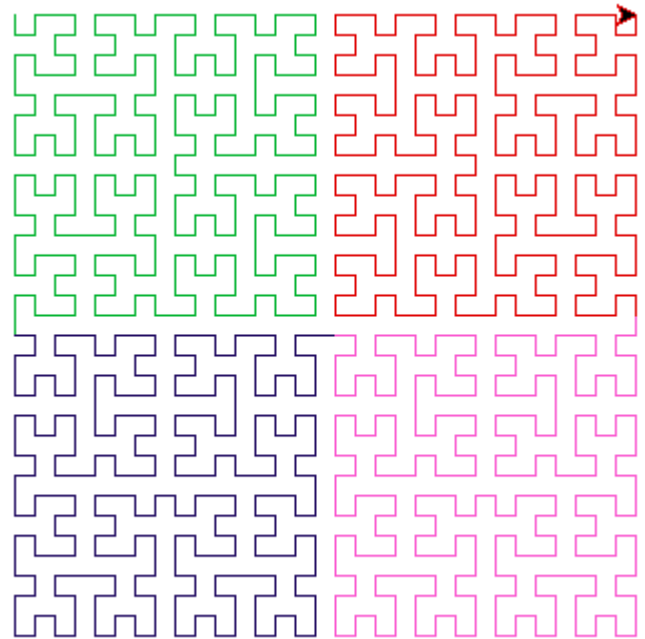


Figura 2. Hilbert curve, order 4