

- Gustavo Adolfo Cruz Bardales - 22779
 - Programacion Paralela y Distribuida
-

Descripción del problema

El objetivo es simular el comportamiento de una intersección vehicular, donde varios semáforos y vehículos interactúan en tiempo real. En cada tick se deben actualizar los estados de todos los semáforos y calcular los movimientos de los vehículos considerando las señales y su posición actual. Además se requiere plantear una solución secuencial y paralela.

Estrategia de paralelización

La estrategia se basa en dos niveles de paralelismo. Primero, se aprovecha la independencia entre semáforos y vehículos para ejecutar estas tareas en paralelo mediante `#pragma omp parallel sections`. Mientras un grupo de hilos actualiza los semáforos, otro grupo procesa el movimiento de los vehículos.

Dentro de cada sección, se paralelizan los bucles que recorren los elementos. Para los semáforos se utiliza un reparto estático, ya que su trabajo es uniforme. Para los vehículos se emplea un reparto dinámico con bloques de tamaño 16, porque su carga puede variar según la lógica de movimiento.

El bucle principal de la simulación no se paraleliza, ya que existe una dependencia temporal: cada tick necesita el estado completo del anterior para mantener la coherencia de la simulación. Por eso, el paralelismo se concentra en las operaciones internas de cada tick.

Uso de OpenMP

Se habilitó el paralelismo dinámico `omp_set_dynamic` para que el número de hilos pueda ajustarse automáticamente según la carga y el hardware disponible, evitando la sobresuscripción y reduciendo overhead cuando hay pocos elementos activos.

También se activó el paralelismo anidado `omp_set_nested`, lo que permite que las secciones paralelas creen sus propios equipos de hilos. Esto mejora la utilización de sistemas con varios núcleos, ya que cada sección puede explotar todos los recursos disponibles.

Además, las estrategias de reparto **schedule** se definieron según la carga: estático para semáforos, por ser predecible, y dinámico para vehículos, por ser más variable.
