

**UNIVERSIDAD TECNOLÓGICA DE SANTIAGO
(UTESA)**



PRESENTADO POR:

ERIK CRUZ
1-18-0759

PRESENTADO A:

IVAN MENDOZA

ASIGNATURA:

ALGORITMOS PARALELOS

ASIGNACION:

TAREA SEMANA 2 – INVESTIGACIÓN

**Santiago de los Caballeros
República Dominicana
Febrero, 2024**

INVESTIGAR:

1. Procesamiento paralelo

El procesamiento paralelo es un enfoque en la informática donde múltiples tareas se realizan simultáneamente utilizando varios procesadores o núcleos de procesamiento. En lugar de ejecutar una sola tarea en un único procesador, el procesamiento paralelo distribuye las tareas entre múltiples unidades de procesamiento, lo que puede mejorar significativamente el rendimiento y la eficiencia de los sistemas informáticos.

2. Modelos de cómputo paralelos

Los modelos de cómputo paralelo son diferentes enfoques arquitectónicos y conceptuales utilizados para implementar el procesamiento paralelo. Estos modelos incluyen la memoria compartida, la memoria distribuida y el procesamiento en masa, cada uno con sus propias características y aplicaciones específicas. Los modelos determinan cómo se distribuyen y comparten los datos entre los procesadores y cómo se coordinan las tareas paralelas.

Características de los modelos:

a) Memoria compartida:

- En este modelo, varios procesadores comparten un espacio de memoria común al que todos pueden acceder.
- Los procesadores pueden comunicarse entre sí simplemente escribiendo y leyendo en la memoria compartida.
- Este modelo simplifica la programación, ya que los procesadores pueden compartir datos de manera directa y fácil.
- Ejemplos de sistemas basados en memoria compartida incluyen las computadoras multiprocesador (SMP) y las arquitecturas de multiprocesadores de núcleo único (CMP).

b) Memoria distribuida:

- En este modelo, cada procesador tiene su propia memoria local y no comparte directamente la memoria con otros procesadores.
- Los procesadores se comunican intercambiando mensajes a través de una red de comunicación.
- Este modelo es común en sistemas distribuidos y clústeres de computadoras, donde cada nodo del clúster tiene su propia memoria y recursos.
- La programación en este modelo puede ser más compleja debido a la necesidad de coordinar y sincronizar la comunicación entre los procesadores.

c) Procesamiento en masa:

- En este modelo, se utilizan grandes cantidades de procesadores simples e independientes para realizar tareas de manera paralela.
- Cada procesador opera de forma autónoma y no comparte memoria con otros procesadores.
- Este modelo es útil para problemas altamente “paralelizables” y se utiliza en sistemas como las computadoras en malla (mesh computers) y los clústeres de computadoras de alto rendimiento (HPC clusters).
- La escalabilidad es una característica clave de este modelo, ya que es posible agregar más procesadores para manejar cargas de trabajo más grandes.

3. Desempeño computacional de algoritmos paralelos

El desempeño computacional de los algoritmos paralelos se refiere a su velocidad y eficiencia en comparación con sus contrapartes secuenciales. Se evalúa utilizando métricas como el speedup, la eficiencia y la escalabilidad, que miden cuánto más rápido es un algoritmo paralelo, cómo utiliza los recursos disponibles y cómo se comporta cuando se aumenta el número de procesadores utilizados.

Speedup:

El speedup es una medida que indica cuánto más rápido es un algoritmo paralelo en comparación con su versión secuencial. Un speedup ideal sería igual al número de procesadores, lo que significa que el algoritmo se ejecuta exactamente N veces más rápido en un sistema paralelo con N procesadores que en un sistema secuencial. Sin embargo, en la práctica, factores como la comunicación entre procesadores y la concurrencia pueden afectar el speedup real.

Eficiencia:

La eficiencia es una medida de qué tan bien se están utilizando los recursos de procesamiento paralelo en relación con el número de procesadores utilizados. Una eficiencia del 100% indicaría que todos los procesadores están contribuyendo de manera óptima al rendimiento del sistema paralelo. Sin embargo, en la práctica, la eficiencia rara vez alcanza el 100% debido a la sobrecarga asociada con la coordinación y la sincronización de tareas entre procesadores.

Escalabilidad:

La escalabilidad se refiere a la capacidad de un algoritmo paralelo para mantener un buen desempeño a medida que se agregan más procesadores. Un algoritmo paralelo se considera escalable si su speedup aumenta linealmente con el número de procesadores. Sin embargo, en la práctica, la escalabilidad puede verse afectada por varios factores, como la naturaleza del algoritmo, la arquitectura del sistema y la eficiencia de la comunicación entre procesadores.

4. Complejidad de la comunicación

La complejidad de la comunicación es un aspecto crítico en el diseño y la evaluación de algoritmos paralelos. Se refiere a la cantidad de tiempo y recursos necesarios para que los procesadores en un sistema paralelo se comuniquen entre sí y compartan datos. Una comunicación eficiente es esencial para evitar cuellos de botella y maximizar el rendimiento del sistema.

5. Optimización

La optimización de algoritmos paralelos implica mejorar su eficiencia y desempeño mediante la minimización de la comunicación entre procesadores, la maximización del uso de los recursos disponibles y la reducción de la sobrecarga asociada con la coordinación y sincronización de tareas. Esto puede implicar la reestructuración de algoritmos, el uso de técnicas de programación paralela específicas y la selección adecuada de la arquitectura de hardware.

<https://github.com/kelok3rik/TAREA-SEMANA-2-PARALELOS>