

Tema IV

PROCESAMIENTO PARALELO

4.2 Introducción

Aunque no existe un consenso sobre la definición de un multiprocesador, las siguientes propiedades corresponden a una de las primeras caracterizaciones de sistemas multiprocesador, que fue realizada por P.H. Enslow en 1977:

- Debe estar compuesto por dos o más procesadores.
 - Los procesadores deben compartir el acceso a una memoria común.
 - Los procesadores deben compartir acceso a canales de E/S, unidades de control y dispositivos.
 - El sistema es controlado por un único sistema operativo.
-

4.3 Tipos de plataforma de computación paralela

▶ Organización lógica

- Visión que tiene el programados
 - Capacidad de expresar tareas paralelas (Estructura de control)
 - Método de comunicación entre las tareas (modelo de comunicación)

▶ Organización Física

- Estructura del hardware
- Modelos de espacio de direcciones común
 - Memoria compartida
 - Un único sistema de memoria física
 - Memoria distribuida
 - Cada procesador tiene su propia memoria

4.3.1 Organización basada en la estructura de control

► Criterios de clasificación

- propiedades del proceso (estructura, topología y ejecución).
- propiedades de interacción.
- propiedades de los datos (división y localización).

- *Descomposición iterativa*: Algunas aplicaciones están basadas en la ejecución de un lazo donde cada iteración se puede realizar de forma independiente. Esta técnica se implementa a través de una cola central de tareas ejecutables, y se corresponde con el paradigma de descomposición en grupos de tareas.
- *Paralelismo algorítmico*: el cual se centra en paralelizar el flujo de los datos de entrada.
- *Descomposición geométrica*: El dominio del problema se divide en pequeños subdominios y cada procesador ejecuta el algoritmo en la parte del subdominio que le corresponde.
- *Descomposición especulativa*: Se intentan N técnicas de solución simultáneamente, y $(N-1)$ de ellas se eliminan tan pronto como una de ellas devuelve una respuesta correcta.
- *Descomposición funcional*: La aplicación se divide en distintas fases, y cada fase ejecuta una parte diferente del algoritmo para resolver el problema. También puede denominarse *segmentación de datos*.
- *Maestro/Esclavo*: El proceso maestro es el responsable de descomponer el problema entre sus procesos esclavos y de, posteriormente, recoger los resultados que le envían los esclavos para ordenarlos y obtener el resultado final.
- *SPMD (Single Program Multiple Data)*: En este paradigma cada procesador ejecuta el mismo código pero sobre distintas partes de los datos.
- *Descomposición recursiva o divide y vencerás*: El problema se divide en subproblemas que se resuelven de forma independiente para, posteriormente, combinar sus resultados parciales y obtener el resultado final.

4.3.1.1 Paradigma Maestro Esclavo

- ▶ El maestro es el responsable de descomponer el proceso en pequeñas tareas, distribuirlas, recoger los resultados y ordenarlos
- ▶ Si hay un gran número de tareas puede ser un cuello de botella
 - Mas de un maestro
- ▶ Balances de carga
 - Estático
 - Se realiza al comienzo de la computación
 - Dinámico
 - Cuando en número de tareas es mayor que el de procesadores
 - Cuando el número de tareas es desconocido

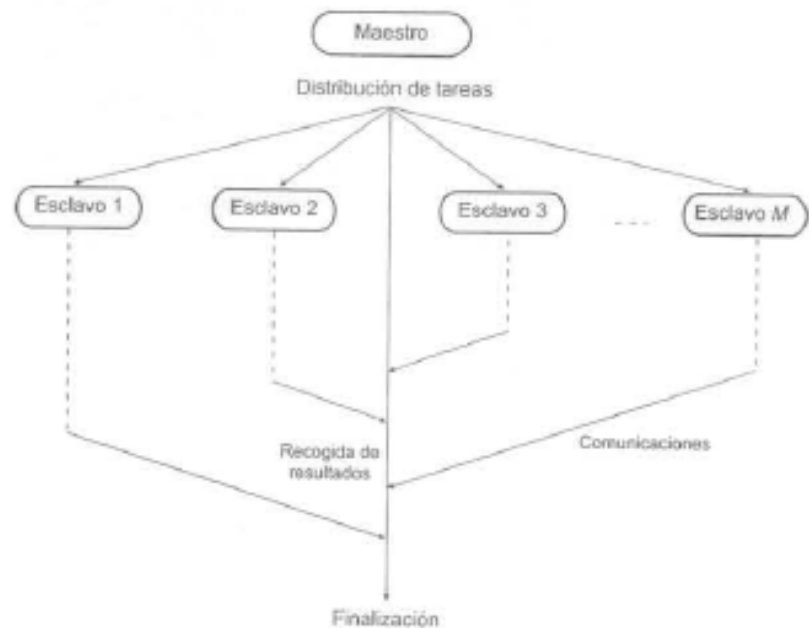


Figura 4.1: Estructura Maestro/Eslavo estática.

4.3.1.2 Paradigma SPMD (Single Program Multiple Data)

- ▶ Cada procesador ejecuta el mismo código pero sobre distintas partes de los datos
 - La comunicación es entre esclavos
 - Eficiente si los datos están bien distribuidos y el sistema es homogéneo
 - Muy sensible a la pérdida de un procesador



Figura 4.2: Estructura básica de un programa SPMD.

4.3.2.1 Espacio de direcciones compartidos

► Parámetros

- Latencia de la red
 - Tiempo que tarda en enviar un mensaje a través de la red de interconexión
- Ancho de banda
 - Número de bits que puede enviar por unidad de tiempo
- Número de procesadores pequeño
 - Compartir medios de interconexión son cuellos de botellas

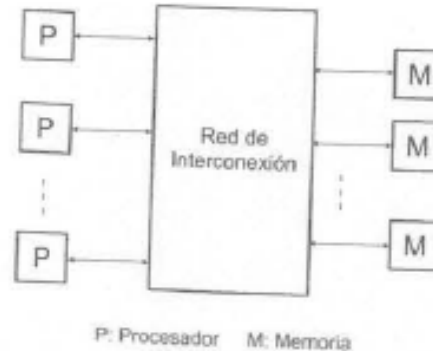


Figura 4.3: Esquema de la arquitectura de memoria compartida.

UMA – Uniform Memory Access

- ▶ El tiempo de acceso es el mismo para cualquier palabra
- ▶ Incorporan memoria cache local y global
 - Problemas de coherencia de caches
- ▶ Bus común
 - Problema de escalabilidad (máximo entre 16 y 32 procesadores)

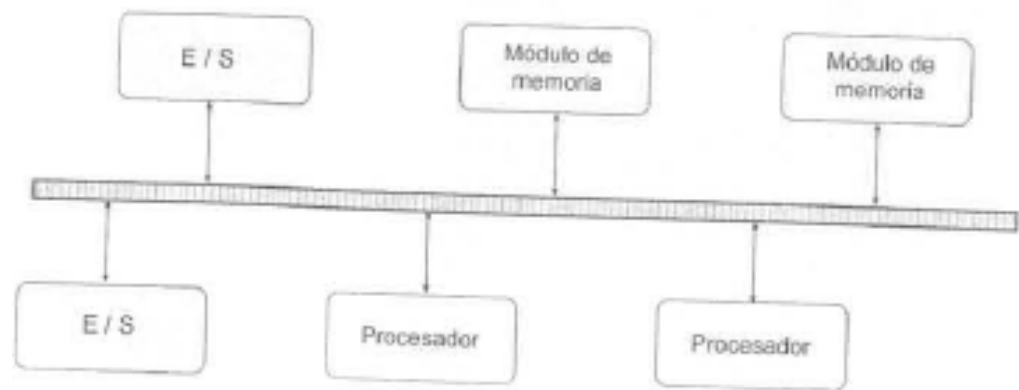


Figura 4.4: Organización multiprocesador de bus común.

NUMA– Non Uniform Memory Access

- ▶ Cada procesador con una memoria local
 - Con código y datos que no tengan que ser compartidos con otros procesadores
 - Evitan el acceso a memoria a través de la red de interconexión
- ▶ Incluye mecanismos hardware dedicado a la coherencia de caches
 - ccNUMA (caché coherent NUMA)
 - COMA (Cahé-Only Memory Access)

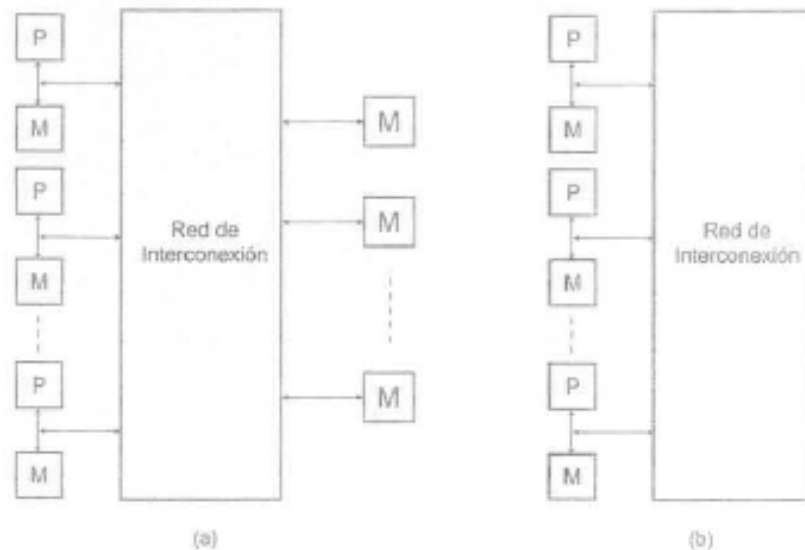


Figura 4.5: a) Arquitectura de memoria compartida con acceso no uniforme a memoria, b) memoria compartida con acceso no uniforme a memoria solo con memoria local

ccNUMA (caché coherente NUMA)

- ▶ Cada nodo tiene una porción de la memoria total del sistema
- ▶ Las variables compartidas se reparten de manera de que solo existe una copia de cada variable
- ▶ Cada nodo consta de uno o varios procesadores con sus caches y su memoria principal
- ▶ La coherencia de caché se mantiene por
 - Escaneo de operaciones (protocolo snoopy)
 - Registro de localización de variables

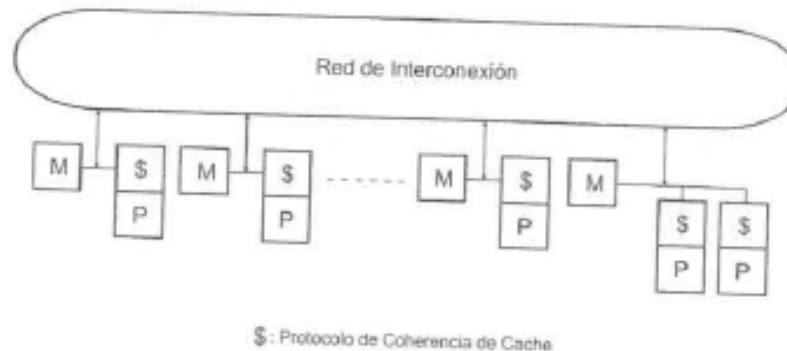


Figura 4.6: Arquitectura ccNUMA.

COMA (Cahé-Only Memory Access)

- ▶ Los procesadores que componen cada nodo no incluyen memoria local, solo caché
- ▶ Si tiene que acceder a una posición de memoria que se encuentra en un nodo remoto, lo copia
 - Complejidad en mantener la coherencia de las variables

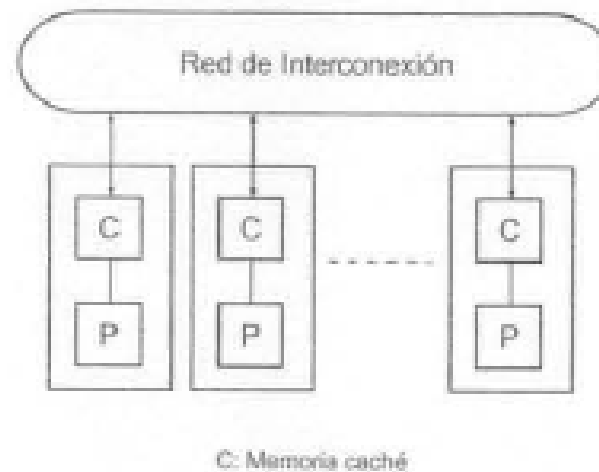


Figura 4.7: Arquitectura COMA.

4.3.2.2 Paso de Mensajes

- ▶ Intercambio de información, en forma de mensajes, entre los diferentes procesadores que forman el sistema
- ▶ Elementos necesarios
 - Emisor, receptor, canal de comunicación, y el mensaje
- ▶ Operaciones básicas necesarias
 - Envío: Usada para enviar el mensaje.
 - Recepción: Usada para recibir el mensaje.
 - Identificación: Usada para identificar cada uno de los procesadores del sistema, mediante un identificador único. Este identificador debe acompañar a las operaciones de envío y recepción para poder identificar al emisor y el receptor.

4.4 sistemas de Memoria Compartida

▶ Aspectos de diseño

- Organización de la memoria (ya comentado)
- El diseño del protocolo de coherencia de la caché
- El diseño de la red de interconexión
 - Para acceder a la memoria remota
 - Intercambiar mensajes entre procesadores
- Topologías de redes de interconexión
 - Estáticas
 - Definidas durante la construcción de la máquina
 - Dinámica
 - Que puede adaptarse a los requisitos de comunicación de los programas que se ejecuten

4.4.1.1 Redes Estáticas

- ▶ Redes unidimensionales
 - Conectar cada procesador con dos procesadores vecinos

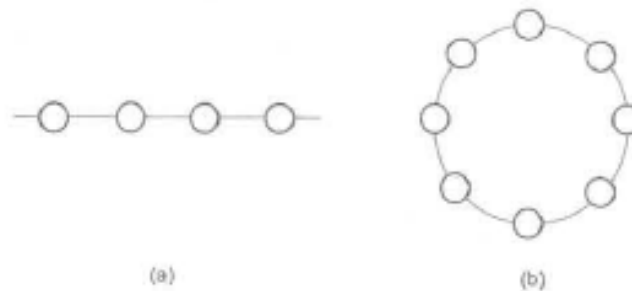


Figura 4.8: Topologías de red unidimensionales: a) lineal, b) anillo.

Redes bidimensionales

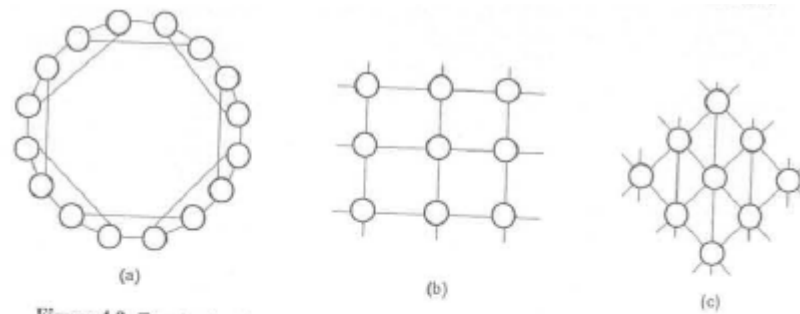


Figura 4.9: Topologías de red bidimensionales: a) anillo cordal, b) malla, c) red sistólica.

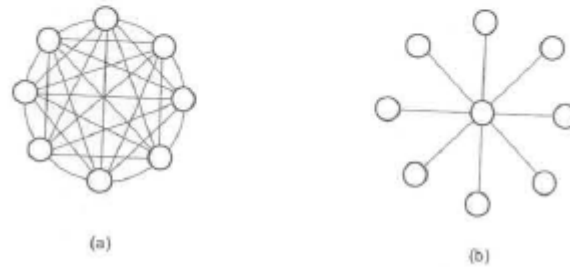


Figura 4.10: Topologías de red bidimensionales: a) red completamente conectada, b) red en estrella.