

Unidad IV: Procesamiento paralelo

4.1 Aspectos básicos de la computación paralela

La computación paralela es una forma de cómputo en la que muchas instrucciones se ejecutan simultáneamente,¹ operando sobre el principio de que problemas grandes, a menudo se pueden dividir en unos más pequeños, que luego son resueltos simultáneamente (en paralelo). Hay varias formas diferentes de computación paralela: paralelismo a nivel de bit, paralelismo a nivel de instrucción, paralelismo de datos y paralelismo de tareas. El paralelismo se ha empleado durante muchos años, sobre todo en la computación de altas prestaciones, pero el interés en ella ha crecido últimamente debido a las limitaciones físicas que impiden el aumento de la frecuencia.ⁿ
^{1 2} Como el consumo de energía —y por consiguiente la generación de calor— de las computadoras constituye una preocupación en los últimos años,ⁿ²³ la computación en paralelo se ha convertido en el paradigma dominante en la arquitectura de computadores, principalmente en forma de procesadores multinúcleo.ⁿ^{3 4}

Las computadoras paralelas pueden clasificarse según el nivel de paralelismo que admite su hardware: equipos con procesadores multinúcleo y multi-procesador que tienen múltiples elementos de procesamiento dentro de una sola máquina y los clústeres, MPPS y *grids* que utilizan varios equipos para trabajar en la misma tarea. Muchas veces, para acelerar la tareas específicas, se utilizan arquitecturas especializadas de computación en paralelo junto a procesadores tradicionales.

Los programas informáticos paralelos son más difíciles de escribir que los secuenciales,⁵ porque la concurrencia introduce nuevos tipos de errores de software, siendo las condiciones de carrera los más comunes. La comunicación y sincronización entre diferentes subtareas son algunos de los mayores obstáculos para obtener un buen rendimiento del programa paralelo.

La máxima aceleración posible de un programa como resultado de la paralelización se conoce como la ley de Amdahl.

4.2 Tipos de computación paralela

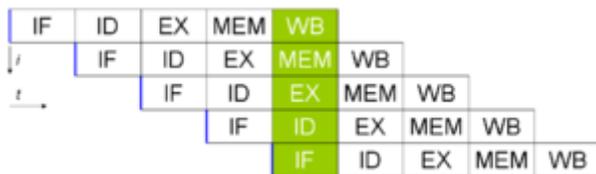
Paralelismo a nivel de bit

Desde el advenimiento de la integración a gran escala (VLSI) como tecnología de fabricación de chips de computadora en la década de 1970 hasta alrededor de

1986, la aceleración en la arquitectura de computadores se lograba en gran medida duplicando el tamaño de la palabra en la computadora, la cantidad de información que el procesador puede manejar por ciclo.¹⁸ El aumento del tamaño de la palabra reduce el número de instrucciones que el procesador debe ejecutar para realizar una operación en variables cuyos tamaños son mayores que la longitud de la palabra. Por ejemplo, cuando un procesador de 8 bits debe sumar dos enteros de 16 bits, el procesador primero debe adicionar los 8 bits de orden inferior de cada número entero con la instrucción de adición, a continuación, añadir los 8 bits de orden superior utilizando la instrucción de adición con acarreo que tiene en cuenta el bit de acarreo de la adición de orden inferior, en este caso un procesador de 8 bits requiere dos instrucciones para completar una sola operación, en donde un procesador de 16 bits necesita una sola instrucción para poder completarla.

Históricamente, los microprocesadores de 4 bits fueron sustituidos por unos de 8 bits, luego de 16 bits y 32 bits, esta tendencia general llegó a su fin con la introducción de procesadores de 64 bits, lo que ha sido un estándar en la computación de propósito general durante la última década.

Paralelismo a nivel de instrucción

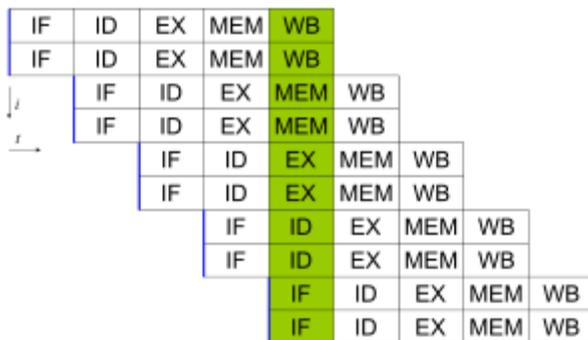


Un pipeline canónico de cinco etapas en una máquina RISC (IF = Pedido de Instrucción, ID = Decodificación de instrucción, EX = Ejecutar, MEM = Acceso a la memoria, WB = Escritura)

Un programa de ordenador es, en esencia, una secuencia de instrucciones ejecutadas por un procesador. Estas instrucciones pueden reordenarse y combinarse en grupos que luego son ejecutadas en paralelo sin cambiar el

resultado del programa. Esto se conoce como paralelismo a nivel de instrucción. Los avances en el paralelismo a nivel de instrucción dominaron la arquitectura de computadores desde mediados de 1980 hasta mediados de la década de 1990.¹⁹

Los procesadores modernos tienen "pipeline" de instrucciones de varias etapas. Cada etapa en el pipeline corresponde a una acción diferente que el procesador realiza en la instrucción correspondiente a la etapa; un procesador con un pipeline de N etapas puede tener hasta n instrucciones diferentes en diferentes etapas de finalización. El ejemplo canónico de un procesador segmentado es un procesador RISC, con cinco etapas: pedir instrucción, decodificar, ejecutar, acceso a la memoria y escritura. El procesador Pentium 4 tenía un pipeline de 35 etapas.²⁰



Un procesador superescalar con pipeline de cinco etapas, capaz de ejecutar dos instrucciones por ciclo. Puede tener dos instrucciones en cada etapa del pipeline, para un total de hasta 10 instrucciones (se muestra en verde) ejecutadas simultáneamente.

Además del paralelismo a nivel de instrucción del pipelining, algunos procesadores pueden ejecutar más de una instrucción a la vez. Estos son conocidos como procesadores superescalares. Las instrucciones pueden agruparse juntas sólo si no hay dependencia de datos entre ellas. El scoreboarding y el algoritmo de Tomasulo —que es similar a scoreboarding pero hace uso del renombre de

registros— son dos de las técnicas más comunes para implementar la ejecución fuera de orden y la paralelización a nivel de instrucción.

4.2.1 Taxonomía de las arquitecturas paralelas

El primer paso hacia la paralelización de las arquitecturas de los computadores, se da con la aparición de los procesadores o sistemas vectoriales. Los procesadores vectoriales extienden el concepto de paralelismo por segmentación al tratamiento de grandes cadenas de datos.

Problemas. Se han propuesto diversas definiciones para arquitecturas paralelas. La dificultad en definir con precisión el término está entrelazada con el problema de especificar una taxonomía de arquitecturas paralelas. El problema central para poder especificar una definición y consiguientemente la taxonomía para las modernas arquitecturas paralelas es lograr satisfacer el siguiente conjunto de imperativos:

Excluir las arquitecturas que incorporan solamente mecanismos de paralelismo de bajo nivel y que se han popularizado tanto como característica típica de las modernas computadoras.

Mantener los elementos útiles de la clasificación de Flynn tales como los flujos de datos e instrucciones.

Incluir los procesadores vectoriales pipelinizados y otras arquitecturas que intuitivamente ameritan incluirse como arquitecturas paralelas, pero que no se ajustan fácilmente al esquema de Flynn.

4.2.2 Arquitectura de los computadores secuenciales

A diferencia de los sistemas combinacionales, en los sistemas secuenciales, los valores de las salidas, en un momento dado, no dependen exclusivamente de los

valores de las entradas en dicho momento, sino también de los valores anteriores. El sistema secuencial más simple es el biestable.

La mayoría de los sistemas secuenciales están gobernados por señales de reloj. A éstos se los denomina "síncronos" o "sincrónicos", a diferencia de los "asíncronos" o "asincrónicos" que son aquellos que no son controlados por señales de reloj.

A continuación se indican los principales sistemas secuenciales que pueden encontrarse en forma de circuito integrado o como estructuras en sistemas programados:

* Contador

* Registros

4.2.2.1 Taxonomía de Flynn

Probablemente la clasificación más popular de computadores sea la clasificación de Flynn. Esta taxonomía de las arquitecturas está basada en la clasificación atendiendo al flujo de datos e instrucciones en un sistema. Un flujo de instrucciones es el conjunto de instrucciones secuenciales que son ejecutadas por un único procesador, y un flujo de datos es el flujo secuencial de datos requeridos por el flujo de instrucciones.

4.2.2.2 Organización del espacio de direcciones de memoria

Organización lógica

Los programas a menudo están organizados en módulos, algunos de los cuales pueden ser compartidos por diferentes programas, algunos son de sólo-lectura y otros contienen datos que se pueden modificar. La gestión de memoria es responsable de manejar esta organización lógica, que se contrapone al espacio de

direcciones físicas lineales. Una forma de lograrlo es mediante la segmentación de memoria.

Organización física

La memoria suele dividirse en un almacenamiento primario de alta velocidad y uno secundario de menor velocidad. La gestión de memoria del sistema operativo se ocupa de trasladar la información entre estos dos niveles de memoria

4.3 Sistemas de memoria compartida: Multiprocesadores

Cada procesador posee su propia unidad de control ejecuta su propio código sobre sus propios datos, puede ejecutar cualquier aplicación (no solo programas vectoriales).

Memoria Compartida Centralizada:

La memoria compartida por todos los procesadores y accesible desde cualquiera. Descompuesta en varios módulos para permitir el acceso concurrente de varios procesadores

Cada procesador debe tener un espacio de direccionamiento suficientemente amplio como para poder direccionarla completamente.

Multiprocesador con un sistema de memoria compartida en el cual el tiempo de acceso varía dependiendo de la ubicación de la palabra de memoria.

La memoria compartida se distribuye físicamente por todos los procesadores (memorias locales). El conjunto de memorias locales forma el espacio de direccionamiento global accesible por todos los procesadores. En los multiprocesadores cada procesador suele tener asociada una cache local y ello introduce el problema de la coherencia en cache: cualquier modificación local de una determinada posición de la memoria compartida se realizara primeramente sobre una chache local y ello puede dar lugar a una visión global incoherente de la memoria. los elementos que integran un multiprocesador puede estar conectados entre sí a través de una estructura

Jerárquica de buses. los buses digitales son los sistemas de interconexión fundamentales adoptados en sistemas comerciales desde estaciones de trabajo a minicomputadores, mainframes y multiprocesadores.

4.3.1 Redes de interconexión dinámicas o indirectas

Características:

Antes de definir las características de las redes de interconexión diremos que se llama nodo a cualquiera de los dispositivos que se quiera conectar a la red, tales como elementos de proceso, módulos de memoria, procesadores de entrada/salida, etc.

- ✓ Grado de los nodos
- ✓ Diámetro de una red
- ✓ Ancho de bisección
- ✓ Latencia de una red
- ✓ Productividad
- ✓ Escalabilidad
- ✓ Simetría
- ✓ Conectividad

Clasificación de Redes de interconexión:

El criterio más importante para la clasificación de las redes de interconexión se basa en la rigidez de los enlaces entre los nodos: a este respecto a las redes pueden clasificarse en estáticas o dinámicas. Una red estática se caracteriza porque su topología queda establecida de forma definitiva y estable cuando se instala un sistema; su única posibilidad de modificación es crecer. Por el contrario, una red dinámica puede variar de topología bien durante el curso de la ejecución o de los procesos o bien entre la ejecución de los mismos.

Por otra parte, las redes pueden ser jerárquicas o no, los son si están formadas por una serie de niveles, con diferente número de nodos, dentro de cada uno de

los cuales existe simetría. La mayoría de las redes jerárquicas suelen ser estáticas, sin embargo, hay algún tipo de topología dinámica que también puede serlo.

Redes de interconexión dinámicas

Las redes de interconexión dinámicas son convenientes en los casos en que se desee una red de propósito general ya que son fácilmente reconfigurables. También por eso, este tipo de Redes facilitan mucho la escalabilidad. En general, las redes dinámicas necesitan de elementos de conexión específicos como pueden ser árbitros de bus, conmutadores, etc. Las principales topologías de redes dinámicas son las siguientes:

- Buses
- Redes de líneas cruzadas o matriz de conmutación (crossbar)
- Redes multietapa o MIN (Multistage Interconnection Network)
 - Redes Omega
 - Redes de línea base
 - Redes Mariposa
 - Redes Delta
 - Redes de Closs
 - Redes de Benes

4.3.1.1 Redes de medio compartido

Entorno de medios compartidos

Ocurre cuando varios host tiene acceso al mismo medio. Por ejemplo, si varios PC se encuentran conectados al mismo cable físico, a la misma fibra óptica entonces se dice que comparten el mismo entorno de medios.

Entorno extendido de medios compartidos

Es un tipo especial de entorno de medios compartidos en el que los dispositivos de networking pueden ampliar el entorno de modo que pueda incluir accesos múltiples o distancias mayores de cableado.

El vehículo básico que empleamos para acceder a nuestra red es la conexión de nuestro ordenador a la misma. Se realiza generalmente mediante cables. Dependiendo del cable y de sus características físicas, podremos realizar diferentes conexiones.

La conexión física entre el ordenador y la red se establece siempre a través de un puerto. Un conector permite enlazar el medio de transmisión con la circuitería de acceso a la red. Para cada sistema de cableado se emplea un puerto distinto y algunas veces un dispositivo accesorio

El cable, que más proyección tiene hoy en día es el de fibra óptica, pero hoy por hoy es caro y difícil de instalar. Sin embargo, es recomendable su utilización para enlazar redes distantes y para crear enlaces muy rápidos entre servidores o interconexión de redes.

Los tres principales medios de transmisión utilizados en las redes locales son:

[par trenzado](#)

[cable coaxial](#)

[fibra óptica.](#)

4.3.1.2 Redes conmutadas

Consiste en un conjunto de nodos interconectados entre si, a través de medios de transmisión, formando la mayoría de las veces una topología mallada, donde la información se transfiere encaminándola del nodo de origen al nodo destino mediante conmutación entre nodos intermedios. Una transmisión de este tipo tiene 3 fases:

- Establecimiento de la conexión
- Transferencia de la información
- Liberación de la conexión

La conmutación en un nodo a la conexión física o lógica de un camino de entrada al nodo con un camino de salida del nodo con el fin de transferir la información que llegue por el primer camino al segundo. las redes conmutadas son las redes de área extensa.

Las redes conmutadas se dividen en:

- Conmutación de paquetes
- Conmutación de circuitos

La conmutación de paquetes:

Es un método de envío de datos en una red de computadoras. Un paquete es un grupo de información que consta de dos partes: los datos propiamente dichos y la información de control, que indica la ruta a seguir a lo largo de la red hasta el destino del paquete. Existe un límite superior para el tamaño de los paquetes; si se excede, es necesario dividir el paquete en otros más pequeños.

Ventajas:

- Los paquetes forman una cola y se transmiten lo más rápido posible.
- Permiten la conversión en la velocidad de los datos.
- La red puede seguir aceptando datos aunque la transmisión sea lenta.
- Existe la posibilidad de manejar prioridades (si un grupo de información es más importante que los otros, será transmitido antes que dichos otros).

La conmutación de circuitos:

Es un tipo de conexión que realizan los diferentes nodos de una red para lograr un camino apropiado para conectar dos usuarios de una red de telecomunicaciones. A diferencia de lo que ocurre en la conmutación de paquetes, en este tipo de conmutación se establece un canal de comunicaciones dedicado entre dos

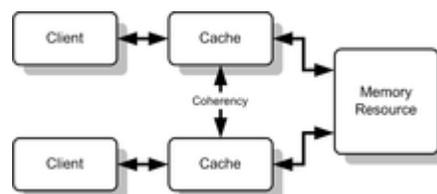
estaciones. Se reservan recursos de transmisión y de conmutación de la red para su uso exclusivo en el circuito durante la conexión. Ésta es transparente: una vez establecida parece como si los dispositivos estuvieran realmente conectados.

Ventajas

- El ancho de banda es definido y se mantiene constante durante la comunicación.
- El circuito es fijo, no se pierde tiempo en el encaminamiento de la información.
- La transmisión se realiza en tiempo real, siendo útil para la comunicación de voz y video.
- Si bien existe retardo en el establecimiento de la llamada, el retardo de la transmisión posterior es despreciable; si el tráfico se realiza generalmente entre el mismo par de estaciones puede ser más veloz.

4.3.2 Coherencia de cache

La coherencia de cache hace referencia a la integridad de los datos almacenados en las caches locales de los recursos compartidos. La coherencia de la cache es un caso especial de la coherencia de memoria.



Múltiples caches con recursos comunes.

Cuando los clientes de un sistema, en particular las CPUs en un multiprocesador, mantienen caches de una memoria compartida, los conflictos crecen. Haciendo referencia al dibujo, si el cliente de arriba tiene una copia de un bloque de memoria de una lectura previa y el cliente de

abajo cambia ese bloque, el cliente de arriba podría estar trabajando con datos erróneos, sin tener conocimiento de ello. La coherencia de la cache intenta administrar estos conflictos y mantener consistencia entre las caches y la memoria.

4.4 Sistemas de memoria distribuida. Multicomputadores: Clusters

Un cluster es una tipo de arquitectura paralela distribuida que consiste de un conjunto de computadores independientes (y bajo coste en principio) interconectados operando de forma conjunta como un único recurso computacional

Sin embargo, cada computador puede utilizarse de forma independiente o separada.

4.4.1 Redes de interconexión estáticas

4.4.2 Cluster

El término clúster (del inglés *cluster*, "grupo" o "racimo") se aplica a los conjuntos o conglomerados de computadoras construidos mediante la utilización de hardwares comunes y que se comportan como si fuesen una única computadora.

La tecnología de clústeres ha evolucionado en apoyo de actividades que van desde aplicaciones de supercómputo y software de misiones críticas, servidores web y comercio electrónico, hasta bases de datos de alto rendimiento, entre otros usos.

El cómputo con clústeres surge como resultado de la convergencia de varias tendencias actuales que incluyen la disponibilidad de microprocesadores económicos de alto rendimiento y redes de alta velocidad, el desarrollo de herramientas de software para cómputo distribuido de alto rendimiento, así como la creciente necesidad de potencia computacional para aplicaciones que la requieran.

Simplemente, un clúster es un grupo de múltiples ordenadores unidos mediante una red de alta velocidad, de tal forma que el conjunto es visto como un único ordenador, más potente que los comunes de escritorio.

Los clústeres son usualmente empleados para mejorar el rendimiento y/o la disponibilidad por encima de la que es provista por un solo computador típicamente siendo más económico que computadores individuales de rapidez y disponibilidad comparables.

De un clúster se espera que presente combinaciones de los siguientes servicios:

1. Alto rendimiento
2. Alta disponibilidad
3. Balanceo de carga
4. Escalabilidad

4.4.3 Programación de clusters

Programación paralela: estos cluster están diseñados y optimizados para correr programas paralelos. En este caso, los programas tienen que ser hechos específicamente para funcionar en forma paralela. Típicamente estos programas son modelos que requieren realizar gran cantidad de cálculos numéricos. La ventaja de programarlos de esta manera y correrlos en un cluster es que se reducen drásticamente los tiempos de proceso. En el caso de modelos meteorológicos usados para predecir el tiempo es obvia la necesidad de correrlos en tiempo mínimo.

Cuando se programa un modelo en una plataforma multiprocesadores (una máquina con más de un procesador), es necesario usar esquemas de programación paralela. Las bibliotecas (MPI) son las que permiten paralelización de tareas. En el caso de los cluster SCALI, portar programas hechos con bibliotecas MPI es directo gracias al uso de bibliotecas SCAMPI.

Es sumamente importante los sistemas de interconexión utilizados entre los nodos en este tipo de clusters. SCI es un standard internacional (IEEE1596) para interconexiones de alta velocidad. Dolphin Interconnect Solutions fabrica una variedad de productos SCI, entre los cuales se encuentran las tarjetas adaptadoras SCI-PCI que SCALI utiliza en sus clusters. El performance de estas tarjetas es sorprendente: varios cientos de megabytes por segundo y tiempos de latencia inferiores al microsegundo en la capa SCI llegando a los límites del bus PCI (64bits/66Mhz).

Usos:

- Análisis Estructural
- Simulaciones de mecánica de fluidos
- Modelos genómicos
- Predicción meteorológica
- Y el estudio de cualquier fenómeno que pueda ser modelado matemáticamente.

4.4.4 Consideraciones sobre el rendimiento de los clusters

Un cluster de alto rendimiento es un conjunto de ordenadores que está diseñado para dar altas prestaciones en cuanto a capacidad de cálculo. Los motivos para utilizar un cluster de alto rendimiento son:

- el tamaño del problema por resolver

- el precio de la máquina necesaria para resolverlo.

Por medio de un cluster se pueden conseguir capacidades de cálculo superiores a las de un ordenador más caro que el costo conjunto de los ordenadores del cluster.

Ejemplo de clusters baratísimos son los que se están realizando en algunas universidades con computadoras personales desechados por "anticuados" que consiguen competir en capacidad de cálculo con superordenadores carísimos.

Para garantizar esta capacidad de cálculo, los problemas necesitan ser paralelizables, ya que el método con el que los clusters agilizan el procesamiento es dividir el problema en problemas más pequeños y calcularlos en los nodos, por lo tanto, si el problema no cumple con esta característica, no puede utilizarse el cluster para su cálculo.

Para que un problema sea paralelizable se debe hacer uso de bibliotecas especiales como lo es PVM (parallel virtual machine) o MPI (Message passage interface), donde la primera es usada especialmente en cluster con nodos heterogéneos (arquitectura del procesador, sistemas operativo, entre otras), y pertenecientes a diferentes dominios de red, la segunda biblioteca usada para cluster homogéneos

Un tipo de software para realizar cluster high performace es OSCAR (Open Source Cluster Application Resources) distribuido bajo licencia GPL. Este software trabaja sobre el sistema operativo Linux. En Windows se podría utilizar "WCC 2003" (Windows Computer Cluster).

4.5 Casos de estudio