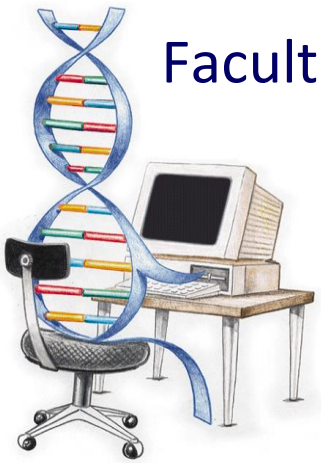
ALGORITMOS EVOLUTIVOS
Curso 2022

Tema 11: Algoritmos Evolutivos Paralelos

Centro de Cálculo, Instituto de Computación
Facultad de Ingeniería, Universidad de la República, Uruguay



cecal

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

Algoritmos Evolutivos Paralelos

Contenido

1. Computación de alta performance
   a. Arquitecturas paralelas
   b. Clusters
   c. Procesamiento paralelo y distribuido
   d. Métricas
2. Algoritmos Evolutivos Paralelos (PEA)
   a. Modelo maestro-esclavo
   b. Modelo de subpoblaciones distribuidas.
   c. Modelo celular
   d. Otros modelos

Computación de alta performance

Conceptos

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

- Motivación: el crecimiento de los requisitos de poder de cómputo
  - Por el aumento en la complejidad de los modelos y problemas abordados.
  - Por trabajar con grandes volúmenes de datos.
- Las aplicaciones modernas exigen requisitos de cómputo que superan los que proporciona un computador tradicional (con la arquitectura de Von Neumann) trabajando de modo aislado.
- En procesamiento paralelo varios procesos cooperan para resolver un problema en común.
- El procesamiento paralelo es una forma eficaz de procesamiento de la información que favorece la explotación de los sucesos concurrentes en el proceso de computación.

Conceptos

- Los avances en el procesamiento paralelo fueron posibilitados por las mejoras tecnológicas:
  - El poder de procesamiento de los microprocesadores
  - El desarrollo de las redes y la comunicación de datos
- En ese contexto se desarrolló activamente el procesamiento paralelo, basado en el estudio en Universidades y aplicado directamente en la industria y organismos científicos.
- Puede considerarse la década de 1990 como la de explosión de la computación paralela.

Procesamiento paralelo y distribuido

- Procesamiento paralelo
  - Procesos concurrentes ejecutando en un mismo computador o computadores interconectados localmente.
  - Ventajas:
    - » Mayor capacidad de proceso y menor tiempo de proceso.
    - » Permite aprovechar la escalabilidad potencial de recursos.
- Procesamiento distribuido
  - Procesos concurrentes ejecutando en forma distribuida en varios computadores no localmente interconectados.
  - Ventajas:
    - » Mejora en el desempeño, robustez y tolerancia a fallos.
    - » Aspectos a resolver: seguridad no centralizada y acceso transparente a los datos no locales.

Arquitecturas paralelas

- El modelo de computación tradicional se conoce con el nombre de arquitectura de Von Neumann y tiene las siguientes características:
  - Posee una única CPU que ejecutar un único programa y tiene acceso a la memoria.
  - Posee una única memoria con operaciones de lectura y escritura.
- Es un modelo robusto que independiza al programador de la arquitectura subyacente y que ha permitido el desarrollo de las técnicas de programación estándar.

Categorización de Flynn

		Instrucciones	
		SI	MI
Datos	SD	SISD	(MISD)
	MD	SIMD	MIMD

Categorización de Flynn (S = Single, M = Multiple)

- Arquitectura SISD (máquina de Von Neumann)
  - Un procesador capaz de realizar acciones secuencialmente, controladas por un programa el cual se encuentra almacenado en una memoria conectada al procesador.
  - Este hardware está diseñado para dar soporte al procesamiento secuencial clásico, basado en el intercambio de datos entre los registros del procesador y la memoria, y la realización de operaciones aritméticas en los registros.

Arquitecturas paralelas

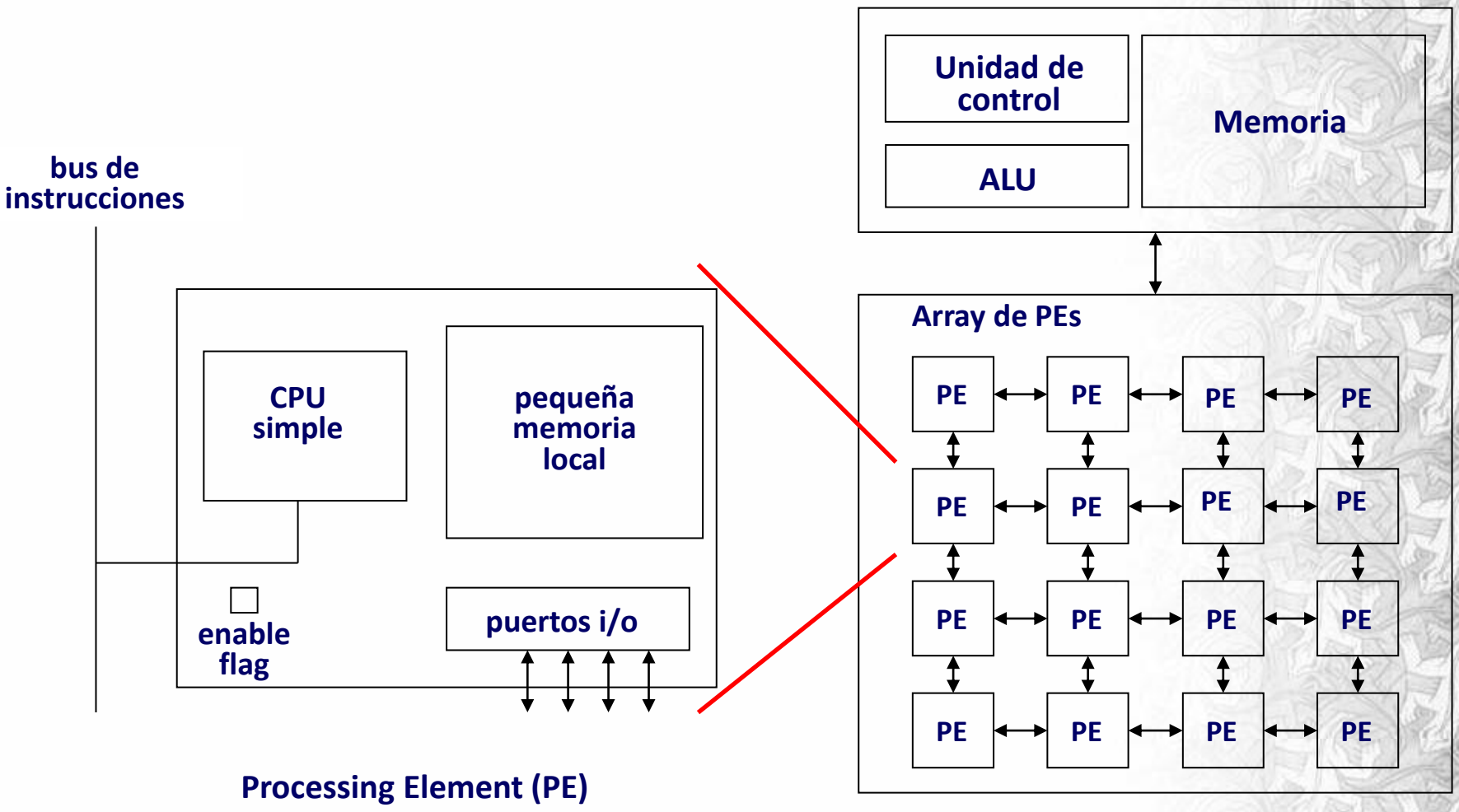
Computadores paralelos

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

- Las máquinas SISD mejoraron su performance, sin embargo los problemas también crecieron o surgió la necesidad de resolver nuevos problemas de grandes dimensiones.
- En este contexto se desarrollaron los computadores paralelos.
- Arquitectura SIMD
  - Un único programa controla los múltiples procesadores.
  - Son útiles en el caso de aplicaciones uniformes como por ejemplo: procesamiento de imágenes, diferencias finitas, etc.
  - Su aplicabilidad está limitada por las comunicaciones fijas entre los procesadores.
  - Se suelen utilizar procesadores sencillos.

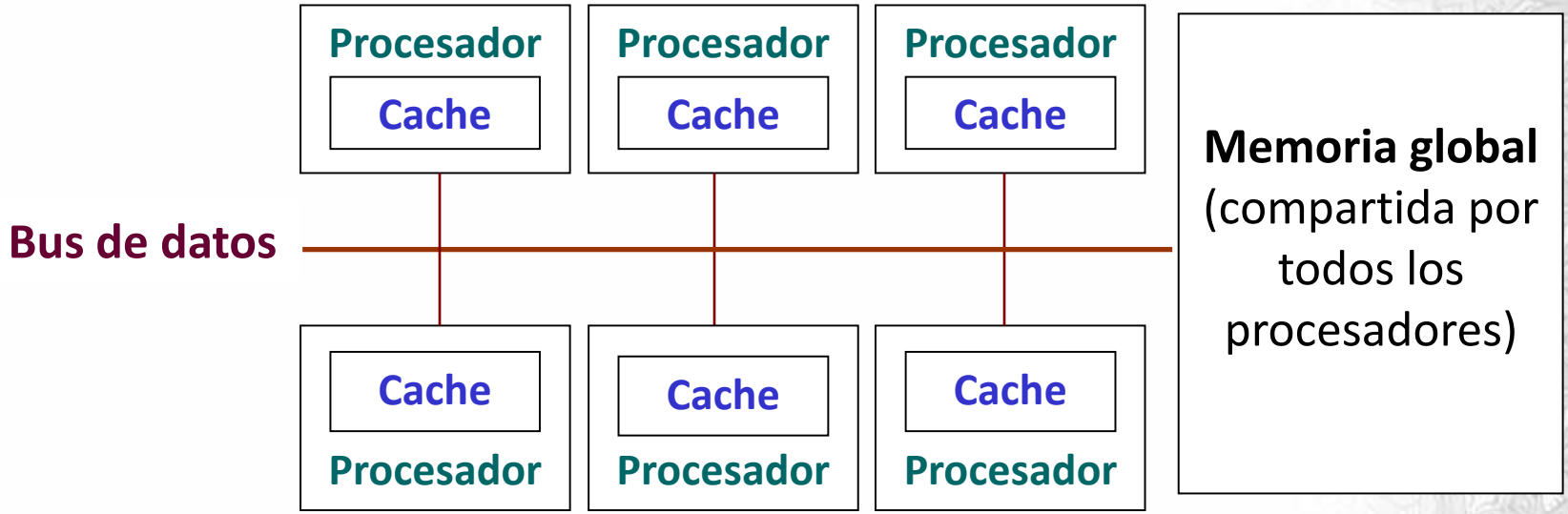
Arquitecturas paralelas

Arquitectura SIMD



Arquitectura MIMD con memoria compartida

- Los procesadores actúan en forma asincrónica y cualquier sincronización es realizada en forma explícita.
- La comunicación entre procesadores se realiza a través de la memoria compartida.
- El bus limita la escalabilidad a un máximo de pocas decenas de procesadores.



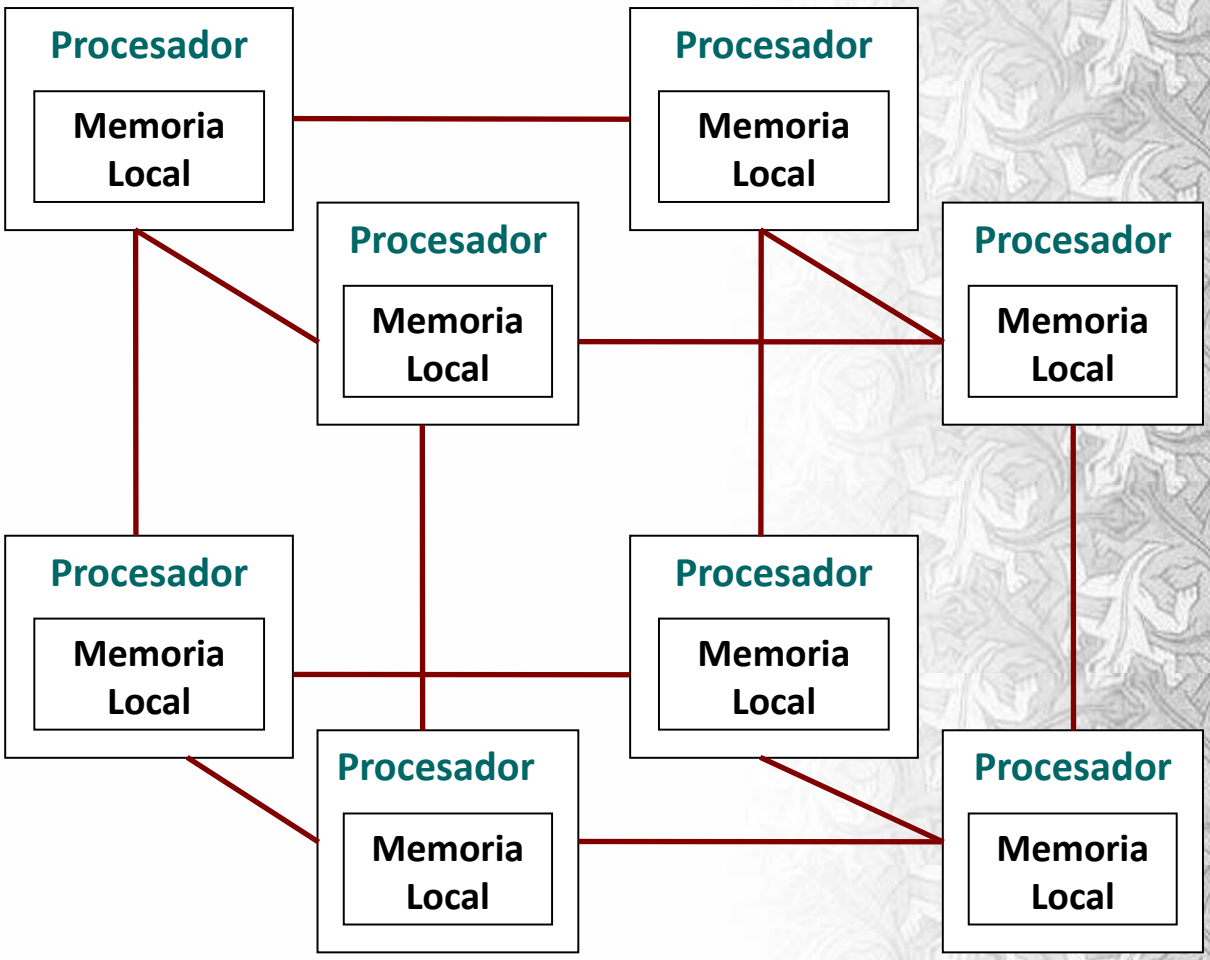
Arquitectura MIMD con memoria compartida

Arquitecturas paralelas

Arquitectura MIMD con memoria distribuida

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

- No existe el concepto de memoria global.
- La comunicación y sincronización se realiza a través del pasaje de mensajes explícitos, con mayor costo que en memoria compartida.
- La arquitectura es escalable para aplicaciones apropiadas a decenas de miles de procesadores.



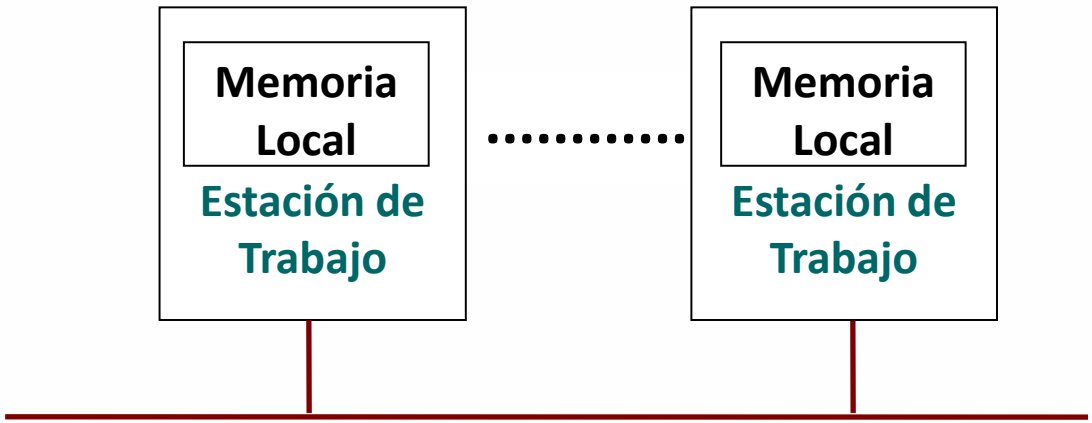
Arquitectura MIMD con memoria distribuida

Arquitecturas paralelas

Arquitectura MIMD con memoria distribuida

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

- Un caso particular: red de computadores



MÁQUINA PARALELA VIRTUAL

Red de datos (LAN, WAN, Ethernet, GBEthernet, FDDI, etc).

Máquina paralela virtual

VENTAJAS

- Usa infraestructura existente.
- Sistema escalable.
- Fácilmente programable.

DESVENTAJAS

- Grandes latencias en las comunicaciones.
- Disponibilidad, seguridad.

Arquitecturas paralelas

Clusters

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

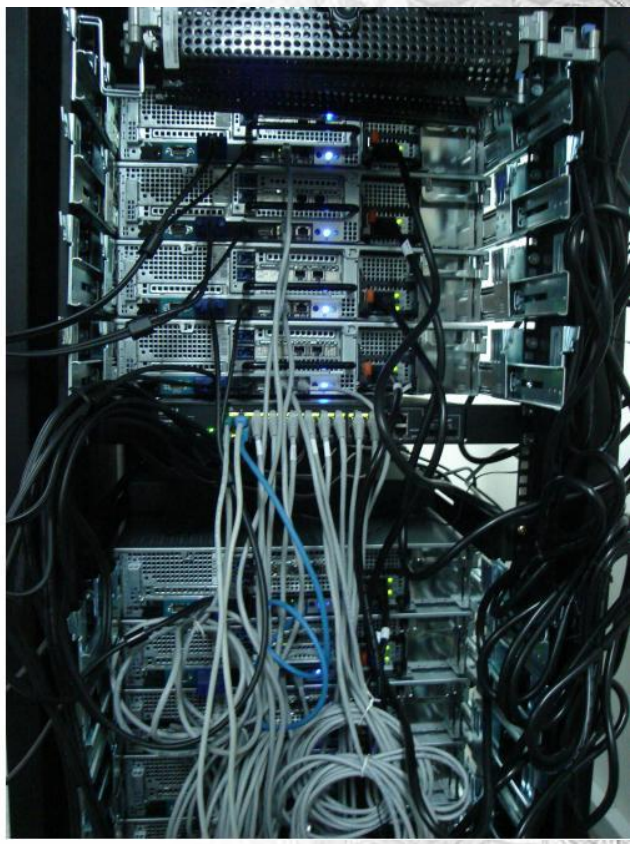
- Cluster: arquitectura de procesamiento paralelo distribuido, compuesta por un conjunto de computadores que trabajan cooperativamente como un único recurso de cómputo integrado.
- Varios factores han influido en el auge de esta arquitectura:
  - La performance de las workstations se duplica cada 18 meses.
  - Las redes de comunicaciones son cada vez más veloces.
  - Existen opciones de almacenamiento redundante de bajo costo (RAID).
- Presentan algunas ventajas interesantes para su utilización:
  - Tienen escalabilidad incremental.
  - La performance de los nodos individuales puede mejorarse sumando recursos adicionales.
  - Es actualizable (nuevos nodos pueden reemplazar a los viejos).

El Cluster FING

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

- Cluster FING: Infraestructura de cómputo de alto desempeño de la Universidad de la República.





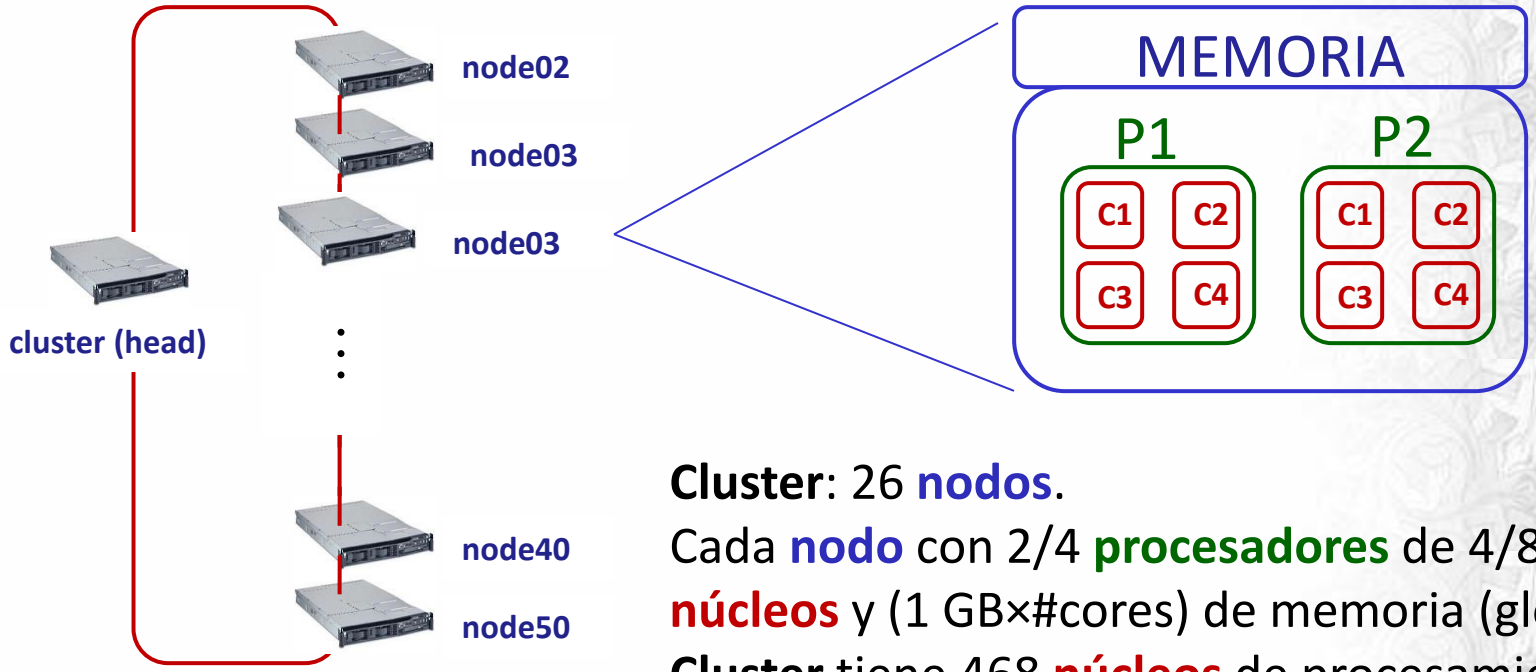
http://www.fing.edu.uy/cluster

Arquitecturas paralelas

Cluster FING: estructura

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

- Combina arquitectura de cluster (memoria distribuida) y multi-core (memoria compartida).
- Permite aprovechar características de ambos modelos de paralelismo: paralelismo de dos niveles.



Cluster: 26 **nodos**.
Cada **nodo** con 2/4 **procesadores** de 4/8/16 **núcleos** y (1 GB×#cores) de memoria (global).
Cluster tiene 468 **núcleos** de procesamiento

Arquitecturas paralelas

Cluster FING: estructura

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

- 26 servidores de cómputo
  - (Dell Power Edge 2950 y HP Proliant DL180) con procesadores Intel Xeon (8, 16 y 64 cores), AMD Magny Cours (24 cores)
- 1 Tesla GPU server (procesadores Xeon quad core y 4 NVIDIA C1060 [960 núcleos de 1.33 GHz.])
- TOTAL: 1444 núcleos de procesamiento
  - 484 núcleos de CPU y 960 núcleos de GPU
  - 992 GB de memoria RAM
  - 32 TB de almacenamiento RAID, 30 kVA de respaldo de batería
  - Swirches GbEthernet y 10GbEthernet

Pico teórico de desempeño aproximado de 5000 GFLOPS
(5×10^{12} operaciones de punto flotante por segundo),
el mayor poder de cómputo disponible en el país

Puesto 8 en LARTOP50 (2013)

Cluster FING: acceso

- Uso libre y gratuito para estudiantes e investigadores de UdelaR, PEDECIBA y universidades asociadas
- Punto de acceso: cluster.fing.edu.uy
- Habilitación de usuarios: sección “Como conectarse” de la web
- Autenticación: mediante par de claves pública/privada
  - Se genera con comandos de ssh (ssh-keygen) y utilitarios (putty-keygen).
  - Clave RSA de 1024 bits.
  - La clave pública se debe enviar por correo electrónico, y la clave privada se almacena en un archivo accesible al(a los) equipo(s) desde los cuales se establecerá la conexión.
  - El procedimiento de generación de claves y de acceso puede consultarse en http://www.fing.edu.uy/cluster/, sección “Como conectarse”.

Métricas para evaluar el desempeño

- Se plantean con los siguientes objetivos:
  - Estimar el desempeño de los algoritmos paralelos.
  - Comparar con el desempeño de los algoritmos seriales.
- Existen algunos factores intuitivos que pueden ser utilizados para evaluar la performance:
  - Tiempo de ejecución: medida tradicionalmente utilizada para evaluar la eficiencia computacional de un programa.
  - Utilización de los recursos disponibles.
- El *speedup* es una medida de la mejora de rendimiento de una aplicación al aumentar la cantidad de procesadores.
- Se define el *speedup absoluto* como $SN = T_0 / T_N$, siendo T_0 el tiempo del algoritmo serial más rápido que resuelve el problema y T_N el tiempo del algoritmo paralelo ejecutado en N procesadores.

Métricas para evaluar el desempeño

- Se define el speedup algorítmico como $S_N = T_1 / T_N$; siendo T_1 el tiempo de ejecución del algoritmo en forma serial y T_N el tiempo del algoritmo paralelo ejecutado sobre N procesadores.
- El speedup algorítmico es el utilizado frecuentemente en la práctica para evaluar la mejora en el desempeño de programas paralelos. El speedup absoluto es difícil de calcular ya que no es sencillo conocer el mejor algoritmo serial que resuelve un problema determinado.
- La situación ideal es lograr un speedup lineal, es decir al utilizar *p* procesadores obtener una mejora de factor *p*.
- No siempre es posible alcanzar el speedup ideal (es habitual obtener speedup sublineal).
- En algunos casos, es posible obtener valores de speedup superlineal.

Métricas para evaluar el desempeño

- La eficiencia computacional se define como $E_N = T_1 / (N \times T_N)$
  - Es decir $E_N = S_N / N$, correspondiendo a un valor normalizado de speedup respecto a la cantidad de procesadores utilizados.
  - Valores de eficiencia cercanos a uno identifican situaciones casi ideales de mejora del desempeño.
- La escalabilidad es la capacidad de agregar procesadores para obtener mejor rendimiento en la ejecución de aplicaciones paralelas.
  - Constituye una de las principales características deseables de los algoritmos paralelos y distribuidos.

Algoritmos evolutivos paralelos

Conceptos

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

- Las técnicas de procesamiento de alta performance se aplican a los AE con los siguientes objetivos:
  - Mejorar la eficiencia, permitiendo afrontar la lentitud de convergencia para problemas cuya dimensión motiva el uso de poblaciones numerosas, o múltiples evaluaciones de funciones de fitness costosas.
  - Mejorar la calidad del mecanismo de búsqueda genética.
- Los algoritmos evolutivos paralelos pueden explotar el paralelismo intrínseco del mecanismo evolutivo, trabajando simultáneamente sobre varias poblaciones para resolver el mismo problema.
- Por lo tanto permiten aprovechar características de paralelismo propias del problema, analizando concurrentemente diferentes secciones del espacio de búsqueda.

Algoritmos evolutivos paralelos

Modelos

- El principal criterio de clasificación surge de las diferentes formas de organización de la población.
  - Maestro-esclavo: Asigna a los procesadores distintas etapas del mecanismo evolutivo. Usualmente se distribuye la evaluación de la función de fitness, que involucra un tiempo de ejecución mayor que el de los sencillos operadores evolutivos.
  - Subpoblaciones distribuidas: Es un enfoque orientado a la distribución de datos en subpoblaciones semi-independientes.
  - Celular: se trabaja con una estructura espacial subyacente que ordena los individuos de la población.
- Estos modelos han evolucionado y se han diversificado, dando lugar a múltiples propuestas de PEAs.

Algoritmos evolutivos paralelos

Modelos

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

- Excepto para el modelo maestro-esclavo sincrónico, los AE paralelos tienen un comportamiento algorítmico diferente al de un AE secuencial.
- Al utilizar poblaciones múltiples, la interacción panmíctica, en la cual cada individuo tiene la posibilidad de interactuar, competir o cruzarse con cualquier otro, se reemplaza por un mecanismo de interacción local, ocasionando que el comportamiento del algoritmo distribuido sea diferente.
- Esta característica proporciona nuevos matices de interés sobre el problema paralelizado, dependiendo de los diferentes criterios utilizados al dividir el problema.

Algoritmos evolutivos paralelos

Modelos

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

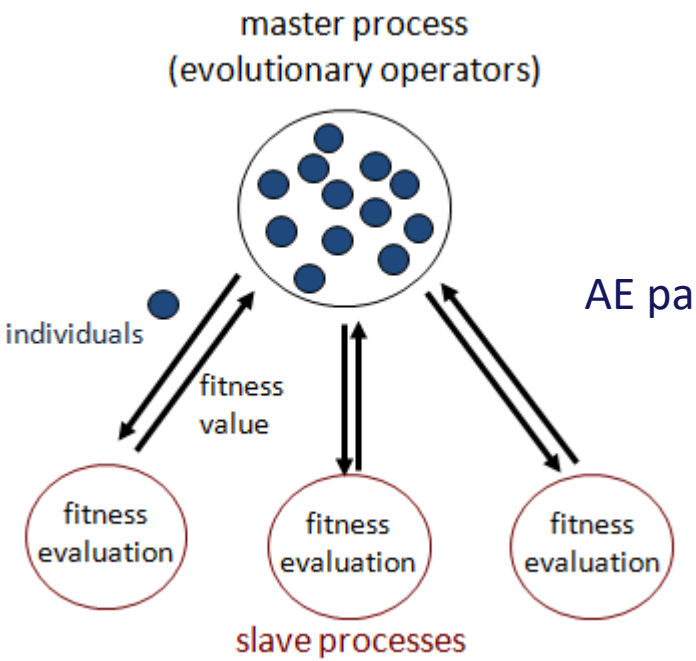
- El principal criterio para la clasificación de AE paralelos es la organización de la población
- También se han propuesto otros criterios para la clasificación de los modelos:
  - Evaluación de la función de fitness: concentrada o distribuida.
  - Número de poblaciones: panmixia o poblaciones múltiples.
  - Mecanismo de intercambio de individuos entre poblaciones múltiples.
  - Mecanismo de sincronización entre los elementos de procesamiento.
  - Cruzamiento: centralizado o distribuido.
  - Selección: aplicada sobre un conjunto local o uno global.

Algoritmos evolutivos paralelos

Modelo maestro–esclavo

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

- Estructura jerárquica:
  - maestro controla la búsqueda
  - esclavos realizan tarea subordinada
- En general, se distribuye la evaluación de la función de fitness
- Evolución panmíctica, al igual que el modelo secuencial



AE paralelo modelo maestro-esclavo

Algoritmos evolutivos paralelos

Modelo maestro–esclavo

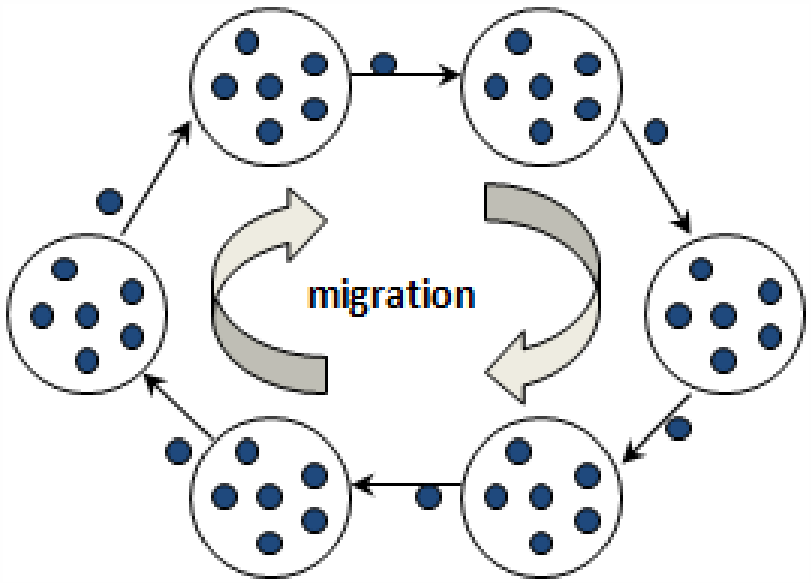
UNIVERSIDAD DE LA REPÚBLICA URUGUAY

- Usualmente el proceso maestro realiza la selección y aplica los operadores evolutivos
- Los procesos esclavos evalúan la función de fitness para subconjuntos de individuos y comunican al maestro los resultados.
- Pueden utilizarse versiones sincrónicas o asincrónicas
- Presenta un modelo centralizado de evolución, ya que la selección se realiza en forma global en el maestro
- Las comunicaciones entre maestro y esclavos son un posible cuello de botella para este modelo
- El modelo es muy efectivo para mejorar la eficiencia computacional en la resolución de problemas complejos

Algoritmos evolutivos paralelos

Modelo de subpoblaciones distribuidas

- Se distribuye la población en subpoblaciones (“islas” o “demes”), que evolucionan de modo semi-independiente



AE paralelo modelo de subpoblaciones distribuidas

- Un operador de migración comunica material genético (individuos) entre islas, incrementando la diversidad

Algoritmos evolutivos paralelos

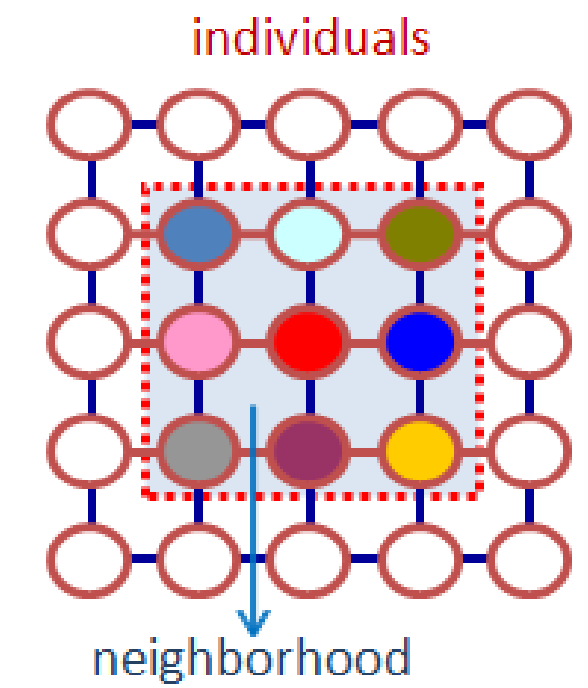
Modelo de subpoblaciones distribuidas

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

- Cada subpoblación opera con la lógica de un AE secuencial
- El AE paralelo presenta un modelo local de evolución ya que la selección y el cruzamiento se realizan en forma local en cada deme
- Las subpoblaciones se manejan en forma independiente, pudiendo inclusive tener diferentes configuraciones de parámetros o aplicar diferentes operadores evolutivos (islas heterogéneas)
- El único intercambio entre las subpoblaciones se produce a través de la migración, que puede ser sincrónica o asincrónica
- Existe una topología definida para comunicar las subpoblaciones (usualmente un anillo unidireccional)
- La diversidad provista por las múltiples islas y la migración le permiten (en general) obtener mejor calidad de resultados que un AE secuencial

Modelo celular

- Los individuos se ubican en una estructura espacial subyacente.



AE paralelo modelo celular

- Existe un mecanismo especial de transmisión de características: la *difusión*.

Modelo celular

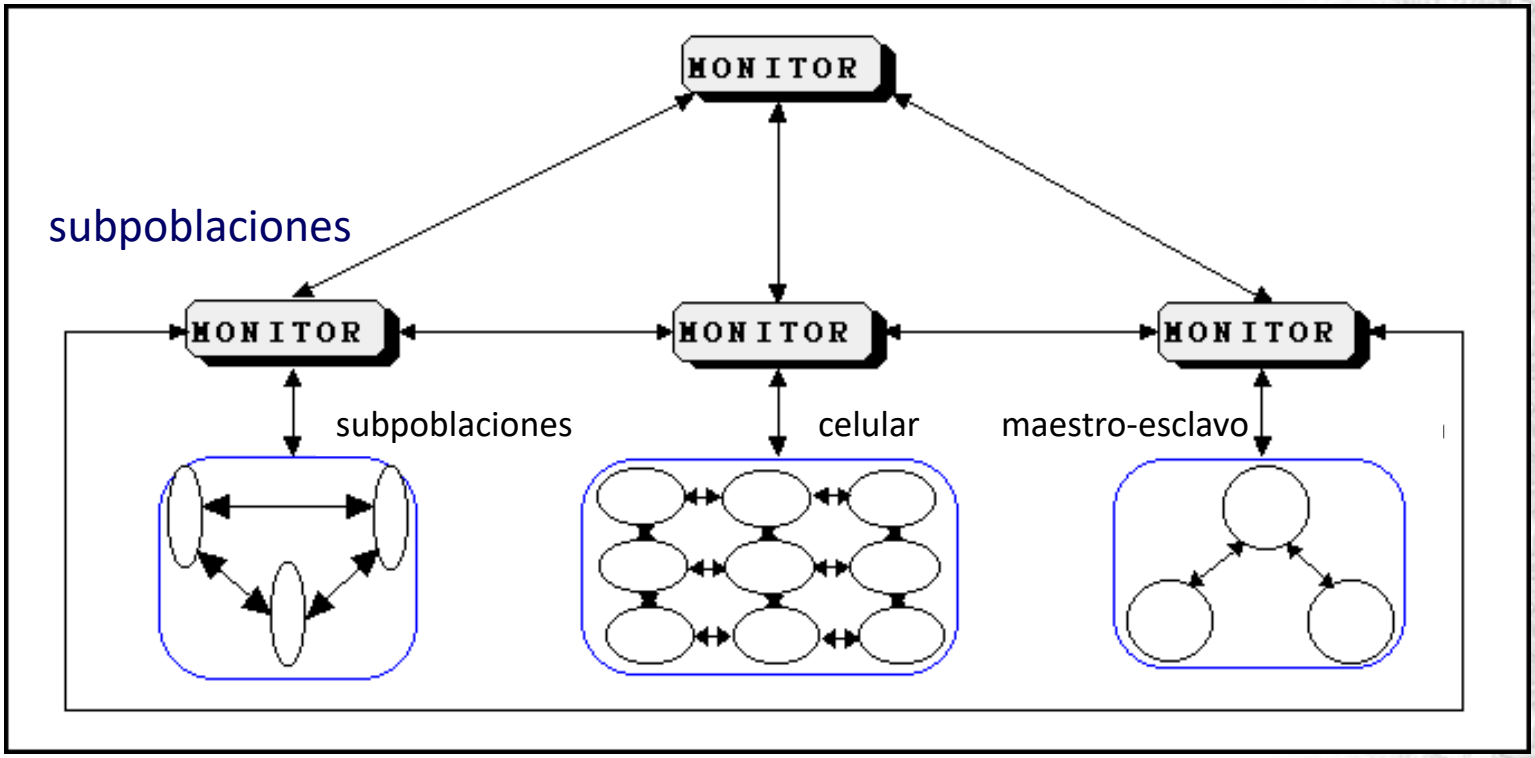
- Presenta un modelo local de evolución restringido a las vecindades ya que la selección y el cruzamiento solamente se realizan con los individuos de la vecindad.
- Las estructuras utilizadas para definir vecindades permiten configurar las características del modelo de difusión, y por tanto del propio algoritmo.
- El mecanismo de interacción restringido por la estructura espacial subyacente y la lenta propagación de características constituyen un técnica efectiva para preservar diversidad en las soluciones de la población.
- El modelo celular es adecuado para implementar sobre arquitecturas de computadores paralelos modelo SIMD.

Otros modelos

- De acuerdo a las taxonomías más comprehensivas existen otros modelos de AE paralelos:
  - Modelo distribuido sin migración
  - Modelo de subpoblaciones dinámicas con solapamiento
  - AG desordenados (messy GA)
  - Modelo de subpoblaciones independientes solapadas
  - AGP de estado estacionario
  - Modelos hibridos

Algoritmos evolutivos paralelos

Modelos híbridos



- Combinan varios modelos de paralelismo
- Tratan de explotar las ventajas de más de un enfoque
- Definen jerarquías de aplicación del paralelismo

Algoritmos evolutivos paralelos

Conclusiones

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

- Los PEA permiten mejorar la eficiencia computacional en la resolución de problemas complejos
- Inclusive son capaces de lograr speedup superlineal (Alba, 2002)
- Implementan un modelo de evolución diferente al secuencial, que con frecuencia les permite alcanzar mejores resultados
- Tienen una alta aplicabilidad para la resolución de problemas complejos del mundo real:
  - Cuando se trabaja con funciones de fitness cuya evaluación demanda altos requerimientos de cómputo
  - Cuando la complejidad del problema hace necesario utilizar grandes poblaciones

Conclusiones

- Existen múltiples bibliotecas de desarrollo disponibles públicamente que encapsulan las funcionalidades necesarias para implementar AE paralelos
- Estas bibliotecas simplifican el trabajo de implementación, haciendo que el usuario se concentre en la resolución del problema y no en los detalles de implementación del paralelismo.
  - Ejemplo en nuestro entorno de trabajo: MALLBA
    - Desarrollo transparente, solo debe incluirse un operador de serialización de individuos, a ser usado en la migración
  - Otras implementaciones de PEAs:
    - Antiguas: ASPARAGOS, DGENESIS, GALOPPS, PARAGENESIS, PGAPACK.
    - Modernas: HeuristicsLab, ECJ, ParadisEO