

Balance Dinámico de Carga en Super-Cómputo

Dr. Manuel Aguilar Cornejo

Presentación elaborada por: Juan Santana Santana

Contenido

- Introducción
- Balance dinámico de carga
- Librería DLML
- Algoritmo utilizando una topología Toroide
- Algoritmo utilizando una topología de Árbol binario
- Plataforma de experimentación
- Resultados
- Conclusiones y trabajo a futuro

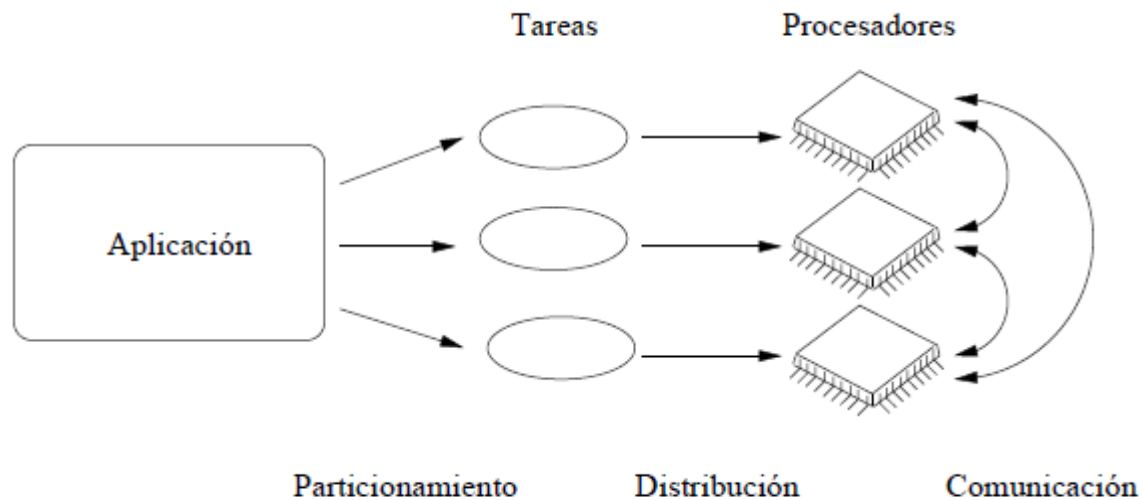
Introducción (1)

- Cómputo paralelo
 - Es una técnica que permite el uso simultáneo de un conjunto de computadoras (Cluster) para ejecutar aplicaciones que resuelven problemas que demandan gran poder de cómputo
 - Reducción de tiempos de respuesta



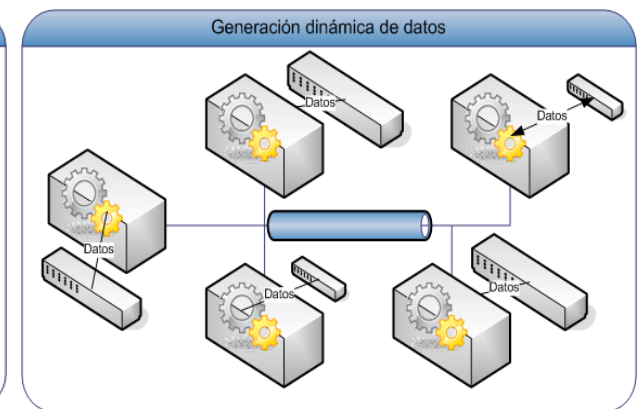
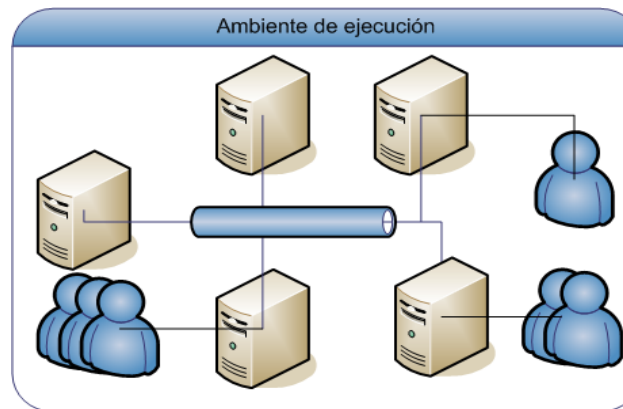
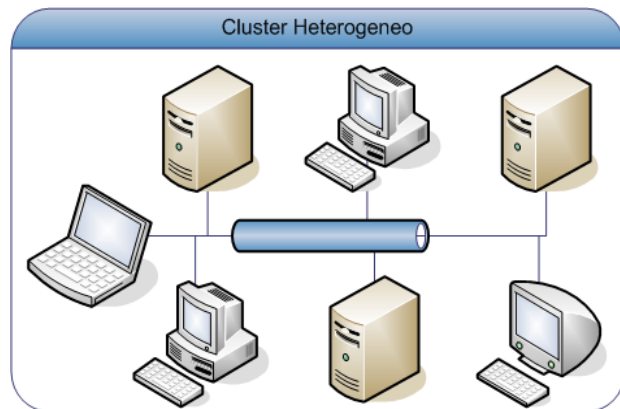
Introducción (2)

- Paralelizar una aplicación
 - Partición del problema
 - Distribución de tareas
 - Comunicación



Introducción (3)

- Inconvenientes en la ejecución
 - Clúster heterogéneo
 - Ambiente de ejecución
 - Generación dinámica de datos



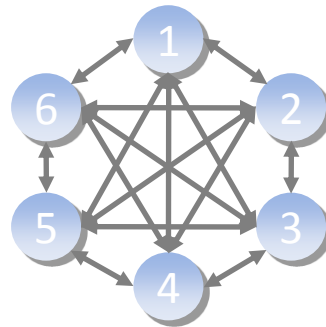
Introducción (4)

- Para reducir el desbalance de carga, una aplicación paralela necesita transferir carga entre procesadores a tiempo de ejecución, esto es conocido como Balance dinámico de carga (**BDC**)
- En general, cuando una aplicación implementa balance dinámico de carga:
 - El tiempo de respuesta de la aplicación se reduce
 - Se optimiza el uso del hardware



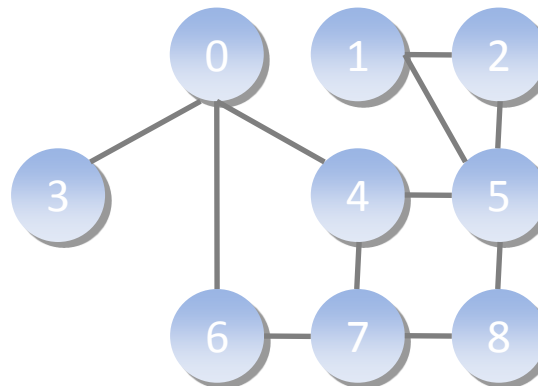
Balance Dinámico de Carga

- El Objetivo del BDC es mantener un equilibrio de carga entre los procesadores del sistema durante la ejecución de la aplicación
- En general, existen políticas globales o parciales de información en algoritmos de BDC para decidir la transferencia de carga entre procesadores
- Con una política global, el algoritmo de BDC requiere el conocimiento acerca del estado de carga de todos los procesadores del sistema



Balance Dinámico de Carga

- Cuando usamos una política de información parcial, solamente el estado de carga de un conjunto de procesadores necesita ser conocido, disminuyendo así el cuello de botella en las comunicaciones
- Sin embargo, la implantación de una estrategia de BDC no siempre es simple, especialmente para los investigadores quienes no están acostumbrados a programar en paralelo

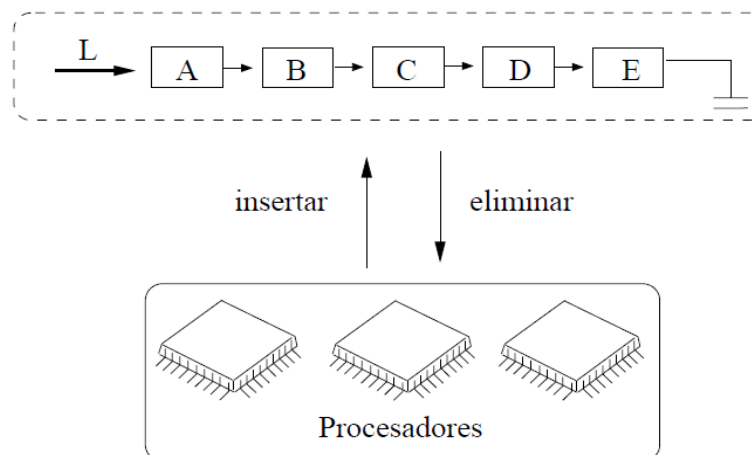


Balance Dinámico de Carga

- Para facilitar la integración y uso de una política de balance de carga varias herramientas han sido propuestas
- En particular, DLML es una librería para el desarrollo de aplicaciones paralelas basada sobre la programación de listas de datos
- DLML es implementada bajo el modelo de paso de mensajes, el cual usa MPI-C.

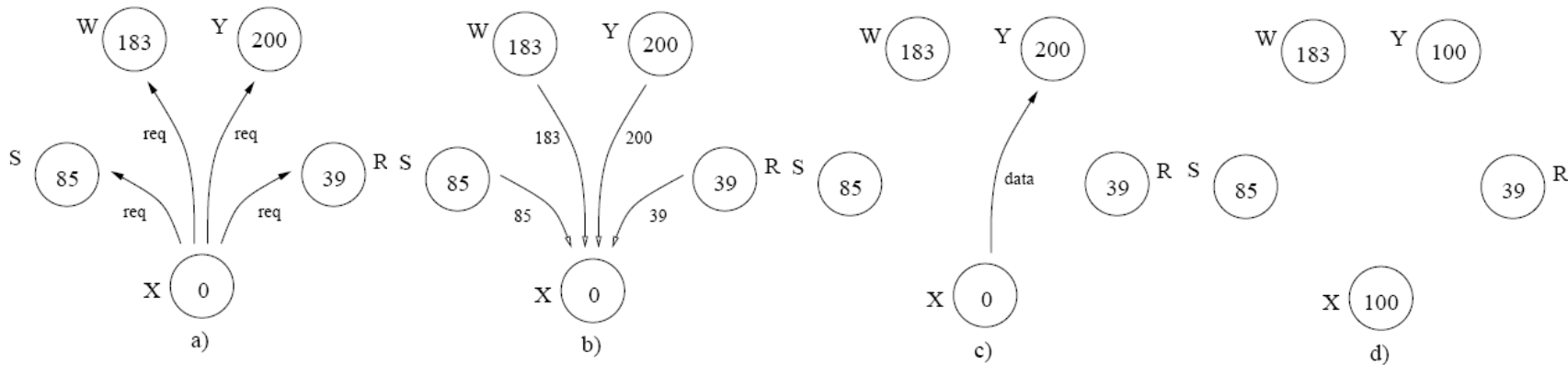
Librería DLML

- DLML (Data List Management Library)
 - Usa el modelo de programación SPMD
 - Los datos son organizados como elementos en una lista
 - Una lista es accesada usando las operaciones típicas de *get()* e *insert()*
 - Esas operaciones básicas aparentemente trabajan sobre una simple lista pero internamente DLML divide la lista entre los procesadores del cluster



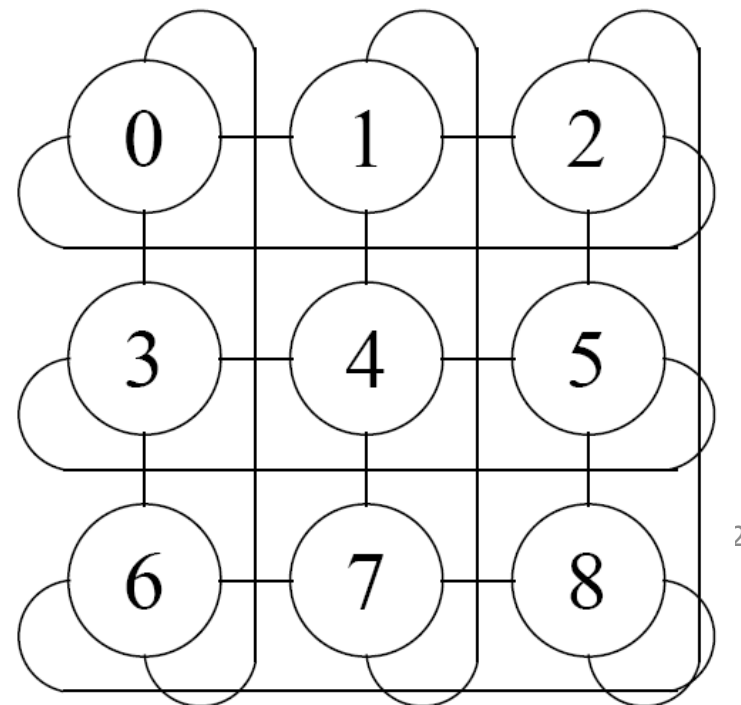
Librería DLML

- Una desventaja de esta herramienta es el algoritmo de subasta global usado para distribuir datos (carga de trabajo) generados durante la ejecución
- El algoritmo de balance de carga en DLML usa una política de subasta global



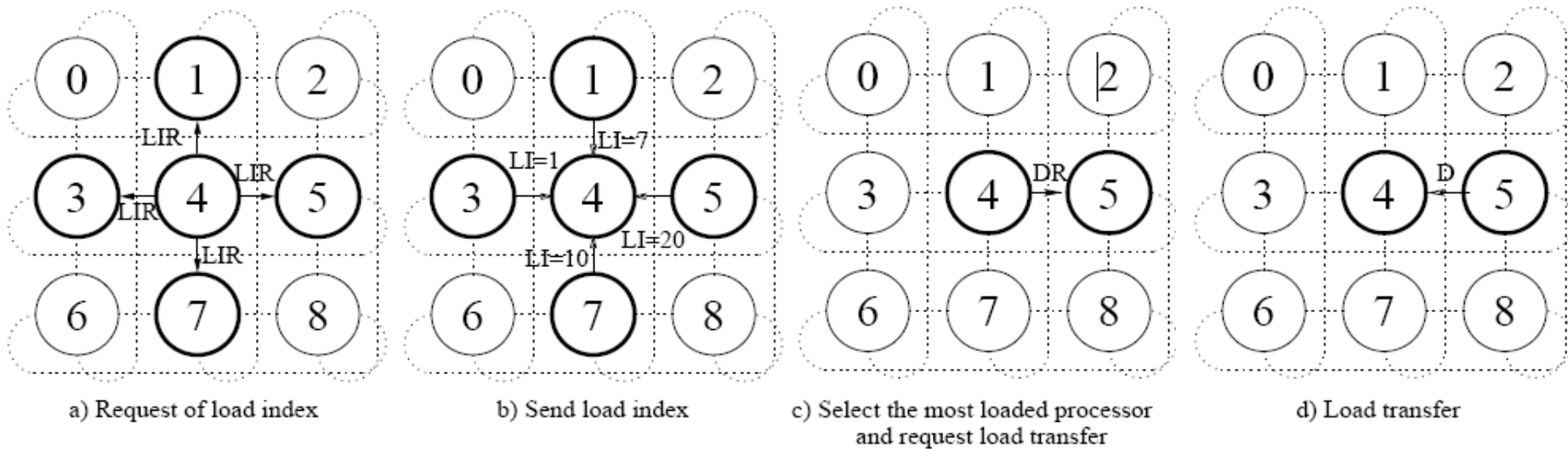
Algoritmo usando una topología Toroide

- La topología de comunicación lógica que usamos en primer instancia es conocida como toroide
- El algoritmo tiene cuatro fases:
 - **Inicialización**
 - **Fase de distribución de carga**
 - **Busqueda del estado global de carga**
 - **Fase de terminación**



Algoritmo usando una topología Toroide

- Fase de distribución de carga
 - El algoritmo considera la misma política de distribución



Algoritmo usando una topología Toroide

Fase de búsqueda global de carga

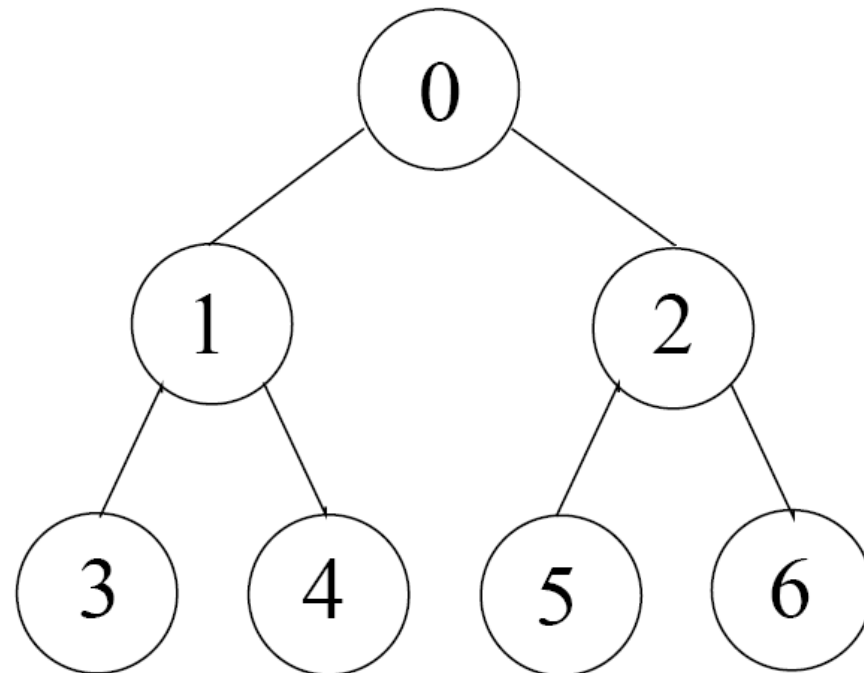
- AL final del procesamiento de datos, muchos de los procesadores resultan descargados y el principal problema es conocer cuándo un procesador debe iniciar la fase de terminación
- En esta parte un PIF (Propagation of Information with Feedback) protocolo ha sido implementado

Algoritmo usando una topología Toroide

- **Fase de terminación**
 - Un mensaje de terminación es propagado a través de un mensaje para iniciar el protocolo de terminación
 - El mensaje notifica a los procesadores que no están cargados en el sistema y así ellos tendrán que comenzar el protocolo de terminación. La propagación va de la raíz a las hojas (processor 0)

Algoritmo utilizando una topología de árbol binario

- Se hizo exactamente lo mismo que en el algoritmo anterior, pero ahora utilizando una topología lógica de comunicación de árbol binario



Plataforma experimental

- Infraestructura

Cluster Pacifico

Nodes	Processors by Node	OS	RAM by Node	Comunication
4	2 Dual core 3 GHz	Linux Centos	2 GB	Gigabit Ethernet
6	1 Quad core 2.4 GHz	Linux Centos	4 GB	Gigabit Ethernet



Cluster Aitzaloa

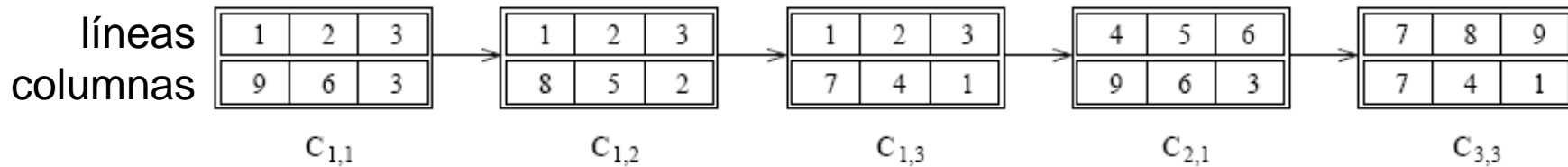
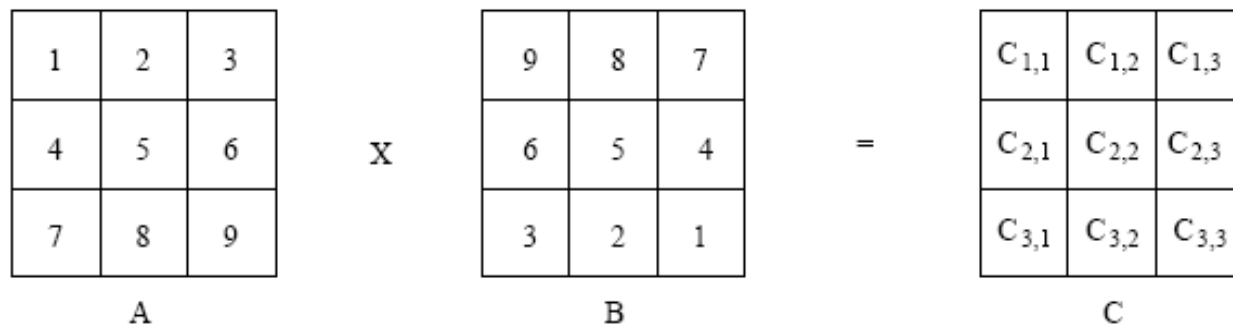
Nodes	Processors by node	OS	RAM by Node	Comunication
270	8 Intel Xeon Quad core 3 GHz	Linux Centos	16 GB	Infiniband



Plataforma experimental

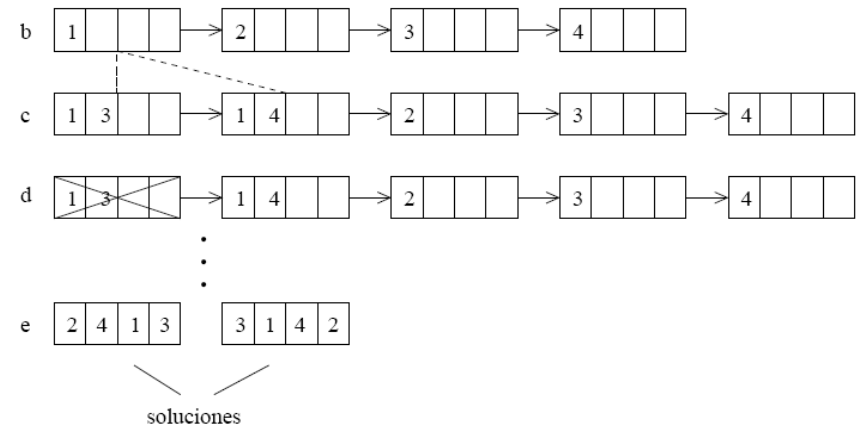
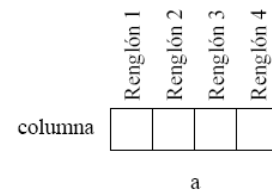
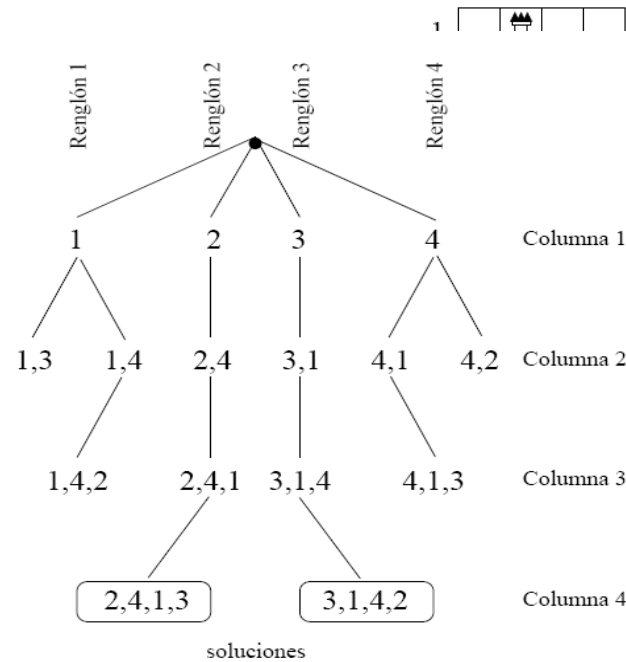
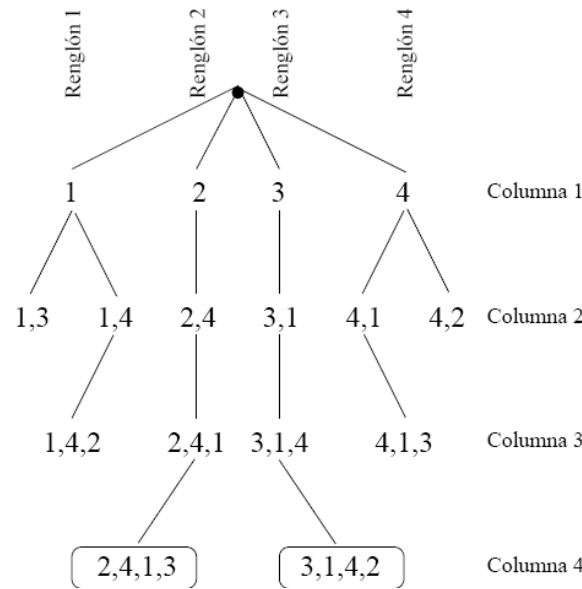
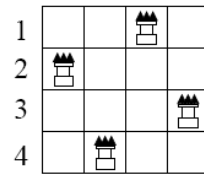
- Aplicaciones

multiplicación de matrices (aplicación sin generación dinámica de datos)



Plataforma experimental

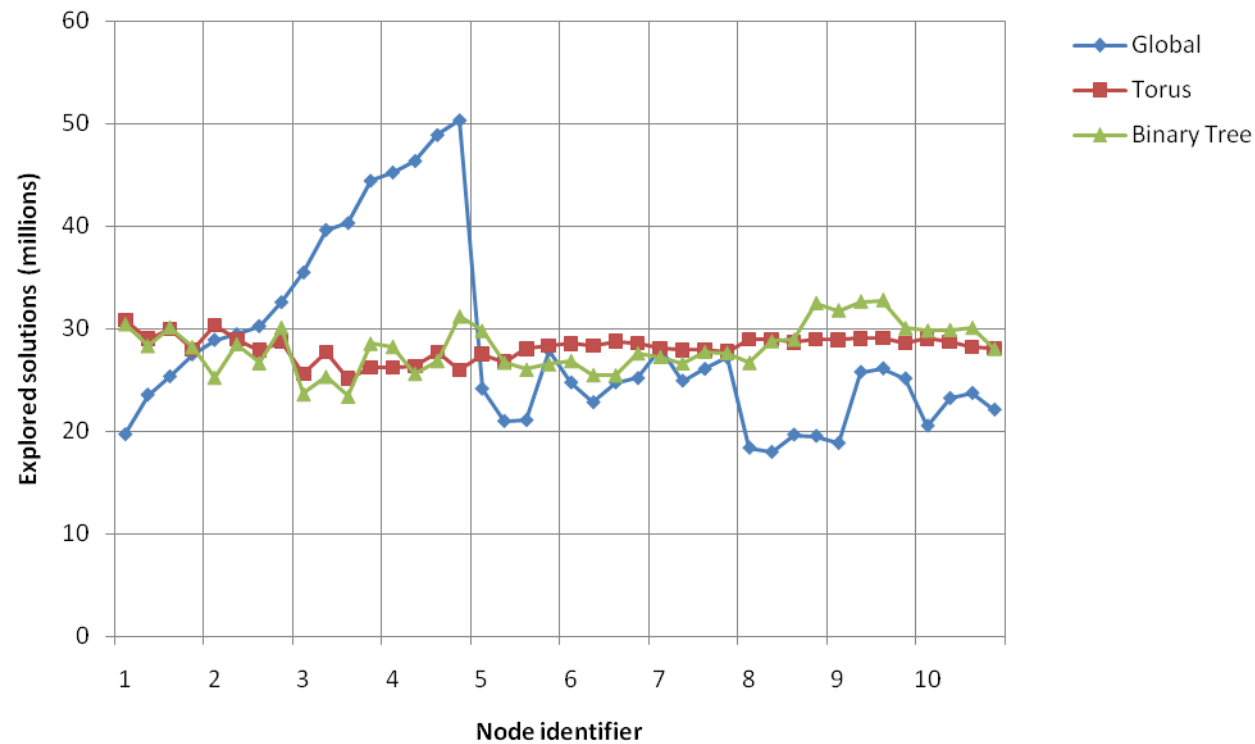
- Aplicación de las N-Reinas (con generación dinámica de datos)



Plataforma experimental

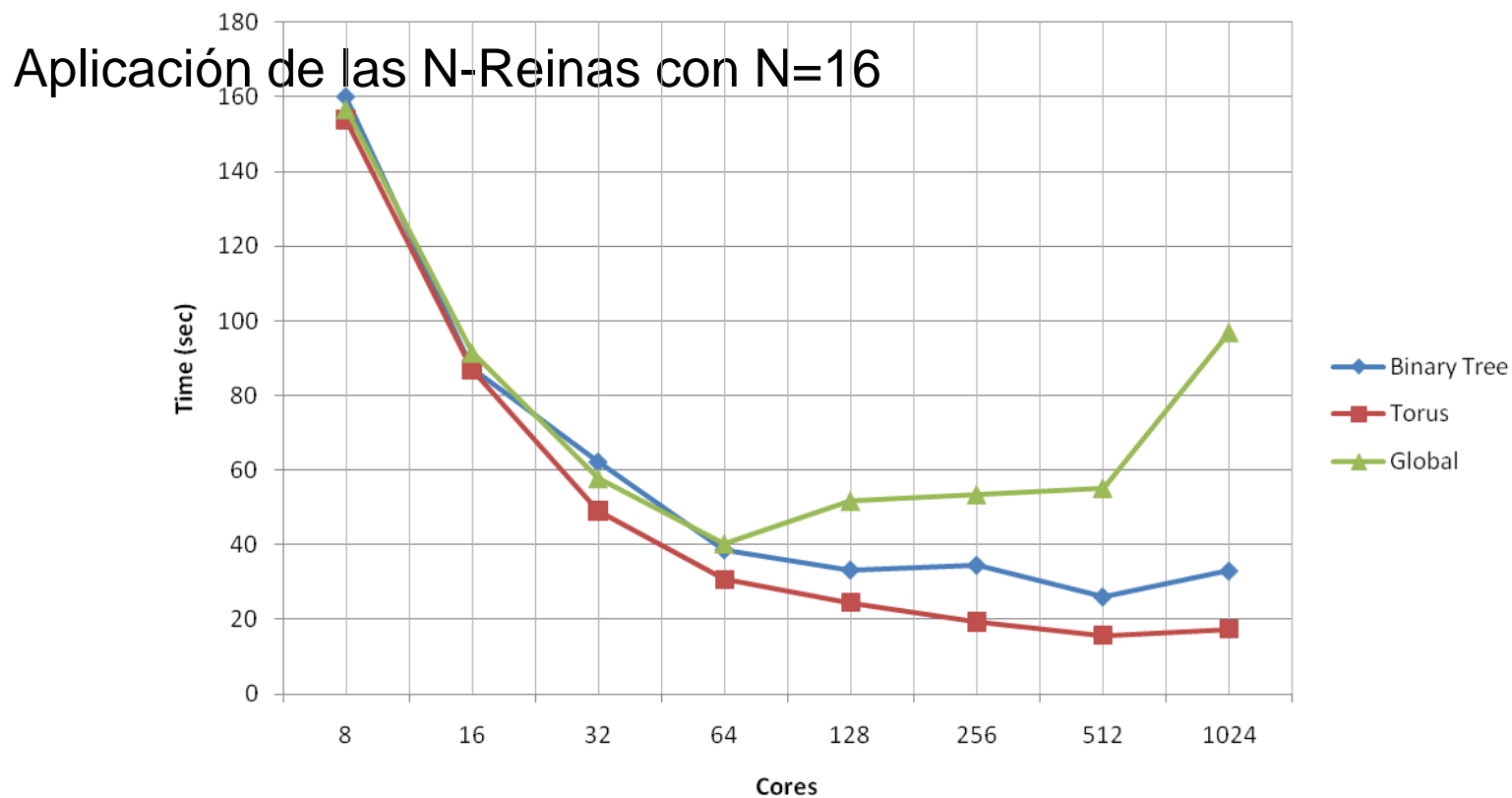
- Distribución de datos

Aplicación de las N-Reinas con $N=16$



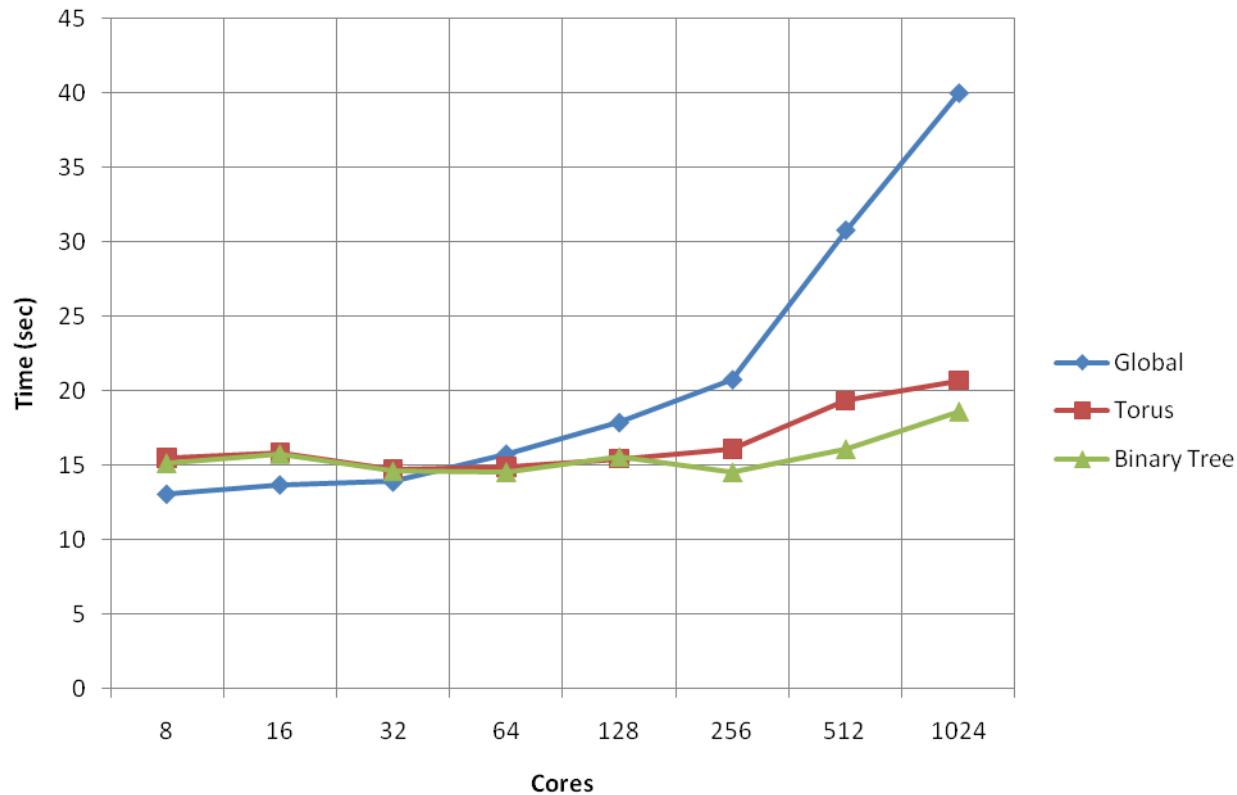
Resultados

- Tiempo de respuesta y escalabilidad



Resultados

- Tiempo de respuesta y escalabilidad
 - Aplicación de multiplicación e matrices, matriz cuadrada de tamaño 1000



Conclusiones y trabajo futuro

- Los resultados mostraron que:
 - El número de mensajes usados ha sido reducido cuando las aplicaciones usaron los algoritmos propuestos
 - Debido a la reducción en el número de mensajes, el tiempo de respuesta fue reducido también con ambos algoritmos y la escalabilidad se mejoro.
 - Se obtuvo una mejoría en la distribución de carga aún y con el cluster heterogeneo.
 - Como trabajo a futuro tenemos la extension de la herramienta DLML con sus nuevos algoritmos hacia tecnologías GRIDS.



UNIVERSIDAD AUTÓNOMA METROPOLITANA

Balance dinámico de carga en super-cómputo

Gracias!!!