


Programación Paralela y Distribuida


Presentación



Introducción

- Asignatura enfocada a la programación, por lo que la evaluación se realizará sobre las distintas prácticas que se plantearán en el curso.
- Tres partes, relacionadas pero distintas:
 - Computación de alto rendimiento (HPC): Antonio
 - E/S Paralela: María
 - Programación para Big Data: Fernando


2



Introducción

- Hay muchas aplicaciones científicas y de ingeniería que tienen tiempos de ejecución muy elevados
 - Dinámica molecular, química cuántica, estudio de nuevos materiales, **bioinformática**, defensa, finanzas, dinámica de fluidos, resistencia de materiales, electrónica, animación, **meteorología**, ...
- La paralelización (HPC) es la única forma de alcanzar tiempos de ejecución razonables y poder abordar experimentos más complejos.
- Internet ha disparado el tamaño de ciertas aplicaciones (Google, Facebook, Amazon,...) y sus necesidades


3



Introducción

- En todos estos casos se suele manejar un gran volumen de datos de entrada y de salida. Incluso hay una rama para cálculo intensivo de datos: Big Data
- La E/S pasa a ser paralela. Dos cambios:
 - Sistemas de ficheros paralelos
 - Bibliotecas para E/S paralela como MPI-IO

4



Introducción

- Al tratar con grandes volúmenes de datos, los modelos "clásicos" pierden su utilidad por falta de escalabilidad
- Se usan nuevos modelos de programación:
 - MapReduce: Dividir el problema en muchas partes y aplicar primero Map y luego recopilar resultados parciales con Reduce. Ej: Google
 - Bulk Synchronous Parallel (BSP): Una sucesión de *Supersteps* (cálculo, comunicación y sincronización) Ej: Facebook, para procesar grafos con Apache Giraph

5



Programación HPC

- Uso de herramientas de análisis (*profilers*) para conocer las aplicaciones. Saber dónde optimizar y paralelizar.
- Memoria distribuida:
 - Se programa con MPI (paso de mensajes)
- Memoria compartida:
 - Se programa con OpenMP (*threads*)
- Coprocesador Intel Xeon Phi:
 - Se programa con OpenMP y Vectorización
- Coprocesador GPU (Nvidia y AMD):
 - Se programa con Cuda (a veces tb OpenCL)

6



Introducción

- ¿Experiencia previa?
 - Profiling
 - OpenMP
 - MPI
 - MIC
 - GPUs