

Informe del lab_6_gpu

Antonio José Manuel Martínez

La práctica 6 ha servido para familiarizarse con el uso de GPUs en cálculos científicos dentro del clúster, utilizando distintas librerías del ecosistema Python. El trabajo ha permitido comparar directamente CPU y GPU en tareas numéricas de gran tamaño, lo cual resulta muy útil para entender en qué situaciones merece la pena recurrir a aceleración hardware y cómo hacerlo correctamente.

Un aspecto especialmente positivo ha sido comprobar que las mejoras de rendimiento dependen más de *cómo* se gestionan los datos que de la librería usada. *CuPy* ofrece grandes ventajas cuando los datos residen en la GPU y no es necesario transferirlos continuamente, mientras que *Numba* resulta práctica cuando se quiere acelerar código *Python* sin cambiar demasiado la estructura original. Esta comparación ayuda a entender que la GPU no siempre acelera por defecto, sino que requiere un planteamiento adecuado.

También ha sido útil trabajar con *sbatch* y la cola *bohr-gpu*, ya que replica un entorno real de trabajo en HPC: gestionar recursos, lanzar experimentos con distintos tamaños y analizar resultados de forma reproducible. El hecho de poder variar parámetros como *N* o *n* desde línea de comandos facilita mucho la experimentación y la automatización, y es algo aplicable a otros laboratorios o proyectos más grandes.

La actividad extra con *PyTorch* ha resultado interesante porque permite ver la versatilidad del paquete más allá del aprendizaje profundo. Su rendimiento en operaciones de álgebra lineal en GPU es muy notable, y la integración con Python hace que sea sencillo de usar. Esto aporta una visión más moderna del cálculo científico y de cómo herramientas actuales pueden servir también para tareas numéricas clásicas.

Como sugerencia, sería útil disponer de alguna guía rápida sobre versiones compatibles de librerías y GPUs para evitar errores en nodos concretos, aunque esto forma parte de las limitaciones habituales en entornos HPC. Aun así, el conjunto de la práctica ha sido claro, útil y directamente aplicable a futuros trabajos que requieran acelerar código o manejar grandes volúmenes de datos.