

# Introducción

## Ejercicio Práctico

Programación en Paralelo

ICME Summer Workshop @ Santiago: Fundamentals of Data Science

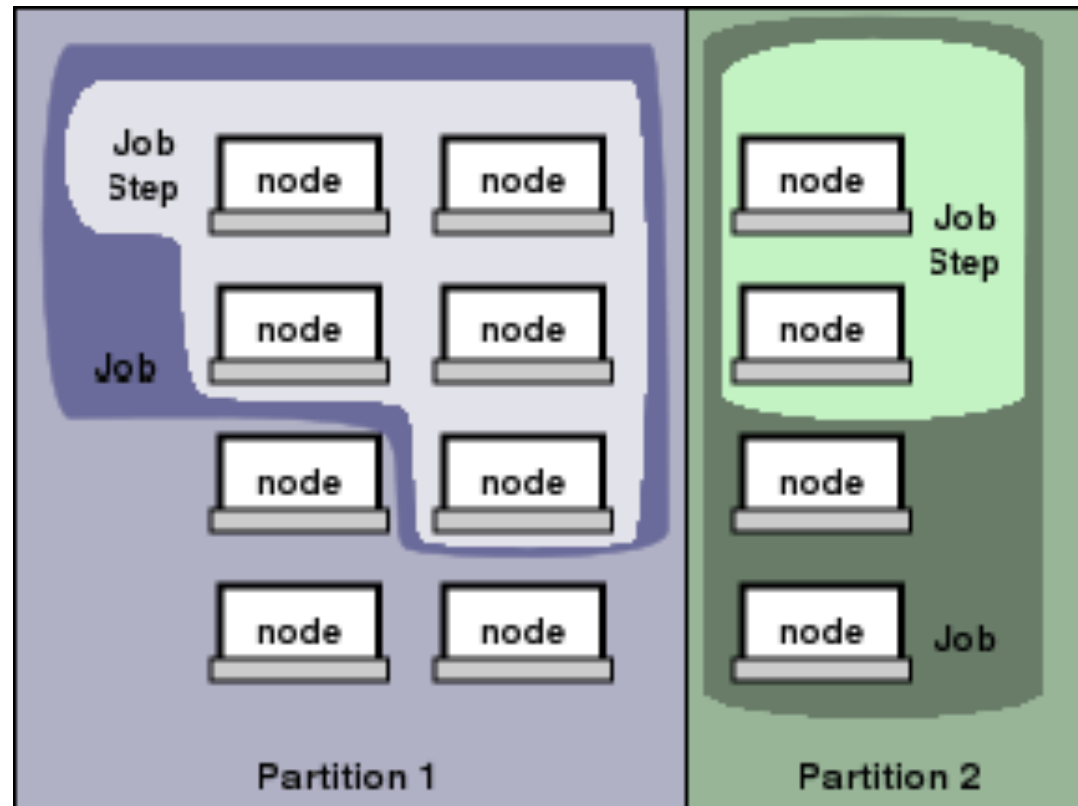
Cindy Orozco

# ¿OpenMP o MPI?

- Como hemos visto, no existe una respuesta única.
- En la realidad se combinan ambos procesos, minimizando la comunicación externa y maximizando el uso de los cores en las tarjetas madre.
- MPI por default identifica cada core como un rank. Entonces como se pueden combinar: Planificador de Tareas

# Planificador de Tarea: SLURM

- Organizador de tareas para pequeños y grandes clusters con Linux.



<https://slurm.schedmd.com/>

# SLURM

```
#!/bin/bash
#SBATCH --job-name=seq
#SBATCH --nodes=20
#SBATCH --ntasks-per-node=6
#SBATCH --cores-per-socket=12
#SBATCH --sockets-per-node=2
#SBATCH --time=24:00:00
#SBATCH --output=slurm2.out
#SBATCH --error=slurm2.err

nPartition=(1 2 3 4 5 6 7 8 9 10 11 12)

./CG_sequential

for i in ${nPartition[*]}; do
    export OMP_NUM_THREADS=$i
    ./CG_parallel
done
```

# Análisis de Rendimiento: Jumpshot

- La forma de decidir la paralelización ideal para un programa se realiza de forma constructiva:

Tomamos el tiempo de ejecución y comparamos cual presenta un mejor speed up.

Opción 1: Instrumentar el código con mediciones de tiempo y printf para imprimirlo

Opción 2: Usar un profiler

# MPE y Jumapshot

- MPI tiene una versión que documenta el tiempo de todos los envíos llamada MPE. Una vez se compila usando MPE, cada vez que se corre el programa se genera un archivo .clog2, que contiene información de todos los mensajes en el programa
- Este archivo se puede visualizar usando una aplicación en Java llamada Jumpshot.

# Para compilar con MPE

- La compilación se hace con MPI

```
mpicc -g -c hello_mpi.c
```

- El linking se hace con MPE

```
mpecc -lm -mpilog hello_mpi.o -o hello
```

- Se ejecuta con MPI

```
mpirun -n 2 hello
```

- Se produce un archivo

```
hello.clog2
```

- Se llama jumpshot

```
jumpshot
```

# ¿Cómo se diferencian los diferentes tipos de send?

- ¿Qué pasa cuando es sincrónico/buffer/standard?
- Veamos el anillo.c



# Solucionar la ecuación de Poisson

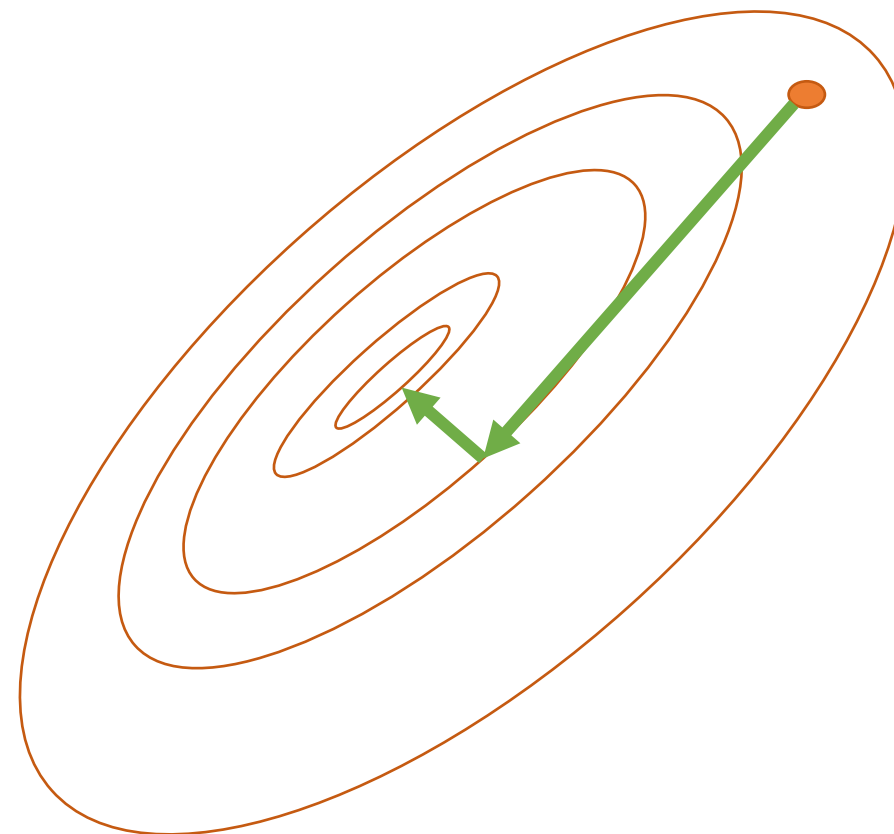
Solucionar

$$Ax = b$$

Es equivalente a solucionar

$$\min_x \|Ax - b\|_2^2$$

Escoger una dirección al azar,  
minimizar en esa dirección y encontrar  
una nueva dirección **perpendicular** de  
búsqueda. Teóricamente terminamos  
en máximo el tamaño de dimensiones  
del problema



# Solucionar la ecuación de Poisson

- La ecuación de Poisson es

$$\Delta f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = -k(x, y)$$

- Usando diferencias finitas se aproxima como

$$\frac{f(i-1, j) + f(i+1, j) - 2f(i, j)}{\Delta x^2} + \frac{f(i, j+1) + f(i, j-1) - 2f(i, j)}{\Delta y^2} = -k(i, j)$$

Supongamos  $k=1$