

CORSO HPC: ESERCITAZIONE GPU + MPI

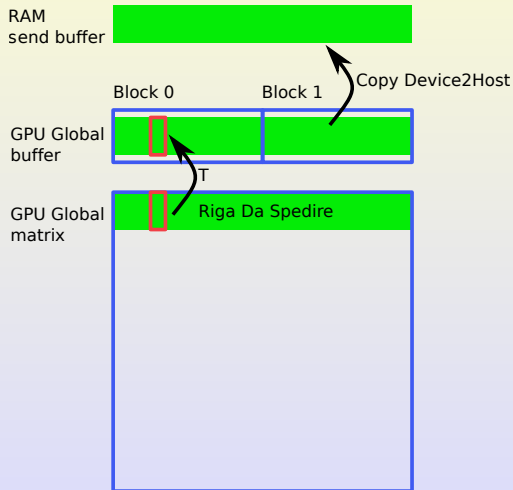
Alessandro Dal Palù¹

1. Dipartimento di Scienze Matematiche, Fisiche e Informatiche,
Università di Parma

- Analisi programma GPU + MPI
- Estensione del programma

- Heat transfer, matrice $NX \times 2NY$
- Progettato per 2 GPU
- Comunicazione MPI per scambio dati di bordo
- Trasferimento GPU -> RAM (rank 0) -> MPI -> RAM (rank 1) -> GPU
- Output: ogni rank scrive un file

- Dataflow spostamento dati e kernel copia su buffer



SETUP

PREPARARE L'AMBIENTE

- Per poter generare gli eseguibili bisogna preparare l'ambiente (CUDA + MPI)
- Lanciare `> module load gnu openmpi cuda`
- I files della lezione sono in
`/hpc/home/alessandro.dalpalu/gpu/mpi`
- Lanciare

```
cd
mkdir gpumpiheat
cd gpumpiheat
cp -R /hpc/home/alessandro.dalpalu/gpu/mpi/* .
```

SETUP

PREPARARE L'AMBIENTE

- Usiamo direttamente la riga di comando (specifica del nostro cluster!)
- Usiamo `nvcc` e lo informiamo delle librerie MPI

```
nvcc main.cu -I/opt/ohpc/pub/mpi/openmpi-gnu/1.10.7/include \  
-L/opt/ohpc/pub/mpi/openmpi-gnu/1.10.7/lib/ -lmpi
```

PROVARE L'ESEMPIO

Ci sono due script per slurm: 2 nodi x 1 GPU o 1 nodo x 2 GPU

+ > `cat slurm_gpumpi_1x2`

```
#!/bin/sh
#SBATCH --partition=gpu
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=2    # processes per node
#SBATCH --gres=gpu:p100:2      # GPU per node
#SBATCH --qos=gpu
#SBATCH --cpus-per-task=1      # CPU cores per MPI process
#SBATCH --time=0-00:01:00
#SBATCH --mem=3G
module load gnu openmpi cuda
echo "#SLURM_JOB_NODELIST      : $SLURM_JOB_NODELIST"
echo "#CUDA_VISIBLE_DEVICES    : $CUDA_VISIBLE_DEVICES"
mpirun -np 2 ./a.out
```

LANCIARE L'ESEMPIO

```
> sbatch slurm_gpumpi_1x2  
> cat test-0.txt test-1.txt > my_file.csv  
> python visualize
```

Viene generato un file png contenente la matrice disegnata (da copiare con scp)