Assignment 6, Part 1: Big Picture

Make the doubly-nested big for loops into a kernel

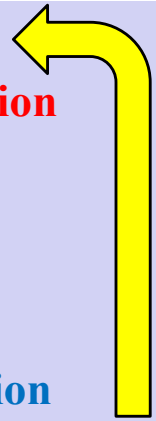
```
__global__ void gpu_histogram_kernel(float *r,float *nhis) {
  // Perform only doubly nested big pair (i, j) loops on device
}


void histogram() {   // Isolate the kernel in a wrapper function
  // Allocate device data & copy values (atom positions etc.) needed for computation
  gpu_histogram_kernel<<<numBlocks,threads_per_block>>>(dev_r,dev_nhis);
  // Copy back computed histogram from device to host
}


int main() {   // No change in the rest of the program
  // Read atomic positions from pos.d
  histogram();
}
```

Kernel definition

Kernel invocation



Map doubly-nested loops to 2D grid & block

On Assignment 6, Part 1

Q. **Why am I getting "nvcc: command not found" error?**

A. **After adding the following lines in .bashrc, you need to either log out and log in again to your account for these commands to be automatically executed, or type "source .bashrc" to explicitly execute them:**

```
module purge
module load usc
module load cuda/10.1.243
```

Q. **What's `pos.d`?**

A. **Your `pdf.cu` will open `pos.d`, and compute pair distribution function for the atomic positions in it; after compilation**

```
nvcc -o pdf pdf.cu
```

the executable `pdf` & input data `pos.d` must be placed in the same directory as the Slurm script `pdf.sl`

Q. **Should I worry about the warnings: variables `dr`, `rij`, `i`, `j` were declared but never referenced?**

A. **No. We didn't bother to remove their declaration, when moving the main computation of `histogram()` to kernel. While they won't do any harm, you could also remove their declarations at the beginning of `histogram()`.**

On Assignment 6, Part 1

Q. What to plot?

A. Pair distribution function $g(r)$ (right column) *vs.* atomic-pair distance r (left column) that will be output into `pdf.d` by your `pdf.cu` program.

Use the force! Reduced variance estimators for densities, radial distribution functions, and local mobilities in molecular simulations

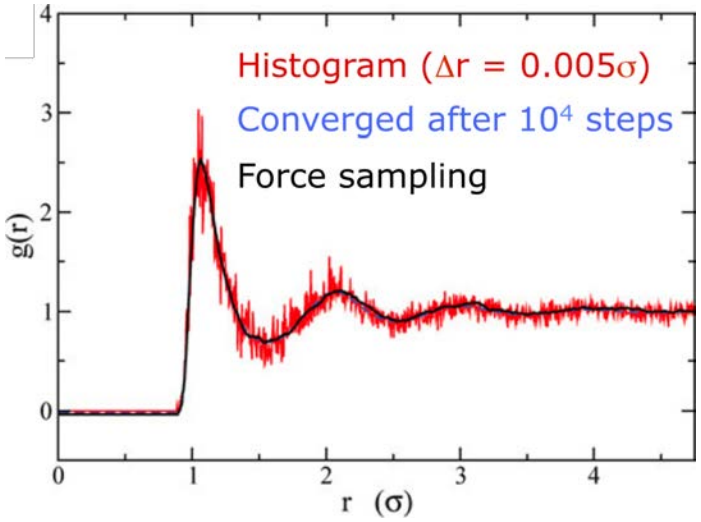
Cite as: J. Chem. Phys. 153, 150902 (2020); doi: 10.1063/5.0029113
Submitted: 9 September 2020 • Accepted: 29 September 2020 •
Published Online: 16 October 2020

View Online Export Citation CrossMark

Benjamin Rotenberg[a)]

AFFILIATIONS

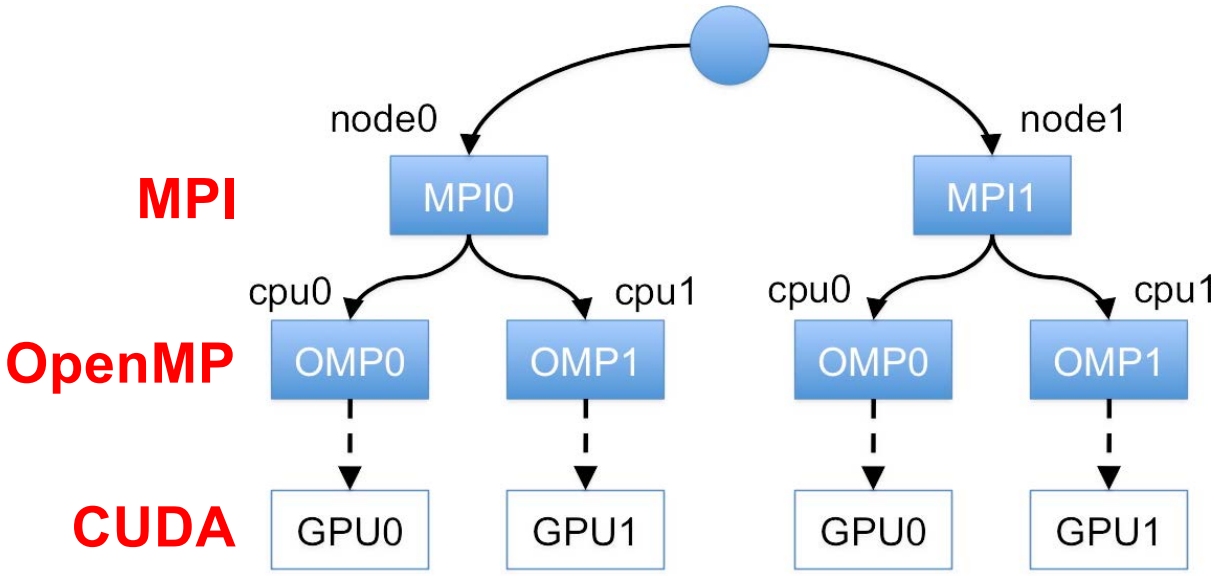
Sorbonne Université, CNRS, Physico-Chimie des électrolytes et Nanosystèmes Interfaciaux, F-75005 Paris, France

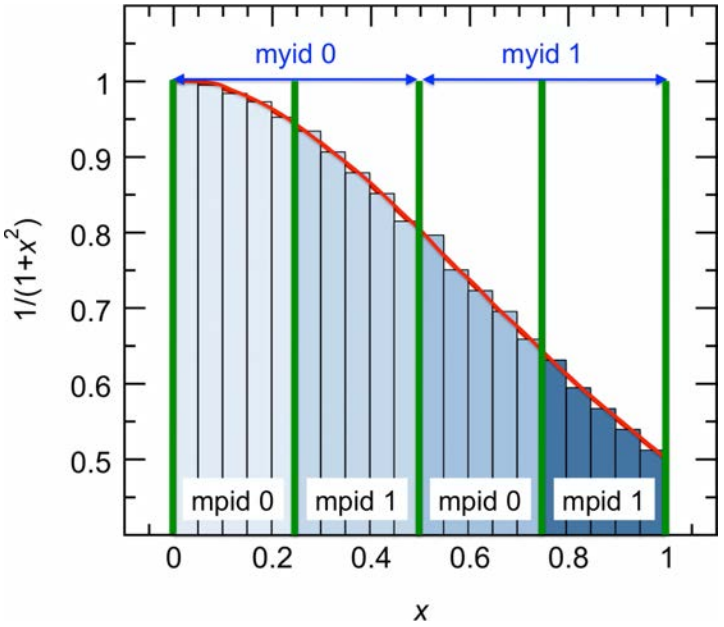


https://aiichironakano.github.io/cs596/Rotenberg-UseTheForce-JCP20.pdf

Assignment 6, Part 2: Big Picture

- **Hands-on practice of the current default programming language (MPI + OpenMP + CUDA) on a cluster of GPU-accelerated multicore computing nodes**
- **Who does what: Hierarchical spatial decomposition with MPI (across computing nodes) + OpenMP (across CPU cores), along with interleaved assignment of loop indices among CUDA threads per OpenMP thread**





Start using MPI+OpenMP+CUDA programming!