High Performance Computing for Science and Engineering I

Exercise 2. OpenMP: Brownian Motion and Bug Hunting

Issued: October 11, 2019

Hand-in (optional): October 25, 2019 08:00

Brownian motion

- Motion of particles $x_i(t) \in \mathbb{R} \quad i = 1 \dots N$

  with random displacements $x_i(t + \Delta t) = x_i(t) + \xi_i^{(t)} \sqrt{\Delta t}$

  from normal distribution $\xi_i^{(t)} \sim \mathcal{N}(0,\ 1)$

- Particles do not interact, good for parallel computing

- Equivalent to the diffusion equation for the density of particles $\rho(x, t)$

$$\rho_t = \frac{1}{2}\rho_{xx}$$

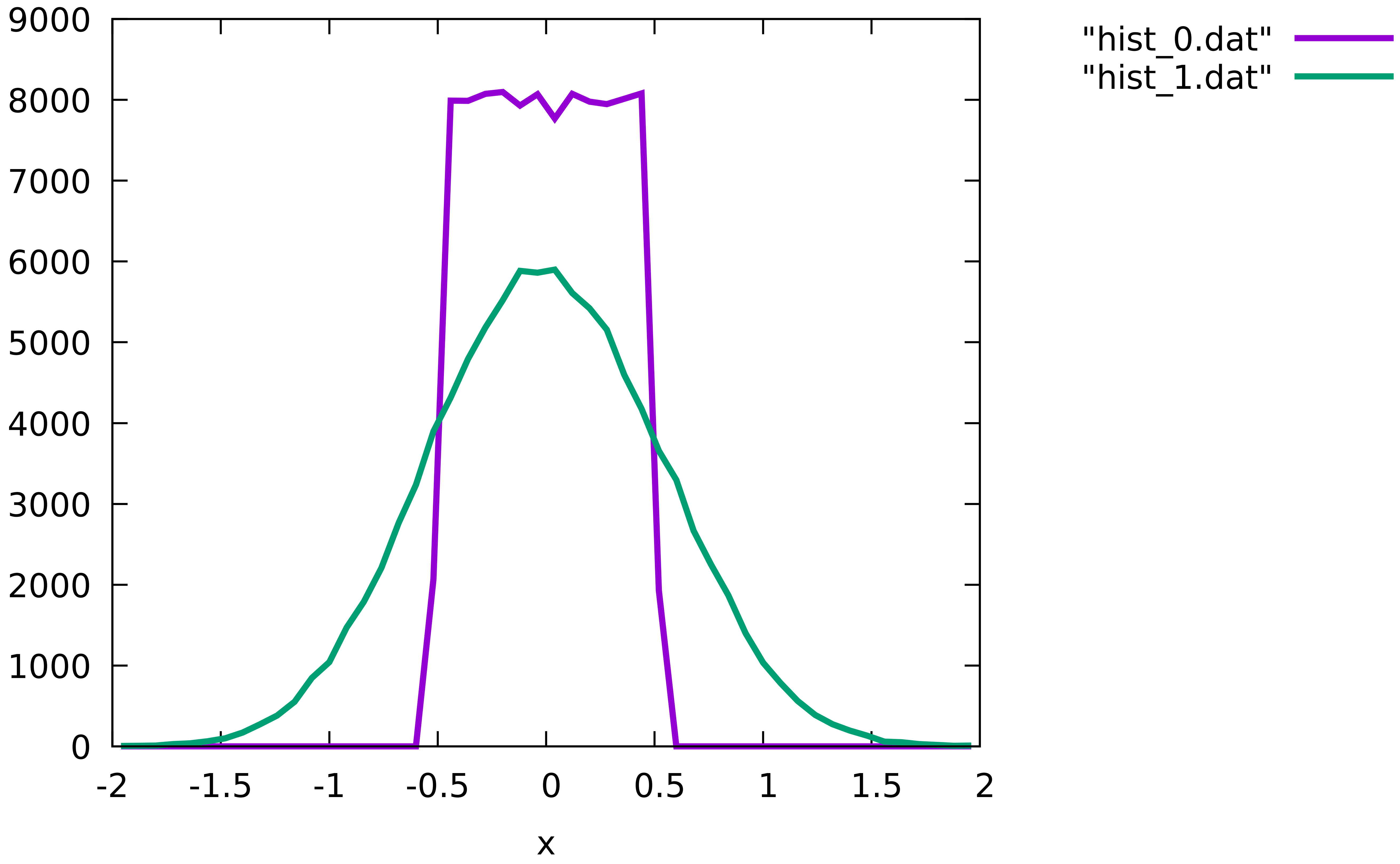
Brownian motion

- Initial positions

$$x_i(0) \sim U(-\frac{1}{2}, \frac{1}{2})$$

- Evolution until time $t_{\max} = 0.2$

- Histograms



Random numbers in C++11

```
#include <random>

// generator: provides pseudo-random sequence of integers
std::default_random_engine gen(19);  // 19 is initial seed
int a = gen();

// distribution: converts to desired type and range
std::uniform_real_distribution<double> dis(-0.5, 0.5);
double b = dis(gen);
```

- generators and distributions are not thread-safe!

OpenMP on Euler

(see also https://scicomp.ethz.ch/wiki/Getting_started_with_clusters#OpenMP)

- Job with OpenMP

```
OMP_NUM_THREADS=24 bsub -n 24 -R “span[ptile=24]” -W 0:10 ./your_executable
```

- Interactive job with OpenMP

```
OMP_NUM_THREADS=24 bsub -n 24 -R “span[ptile=24]” -W 0:10 -Is /bin/bash
```

- Copy file from Euler to your computer (run from your computer on Linux, Mac or even Windows 10)
  (<path> is absolute or relative to your home directory)

```
scp <user>@euler.ethz.ch:<path> ./

scp -r <user>@euler.ethz.ch:<path_to_directory> ./
```

(recursive)