

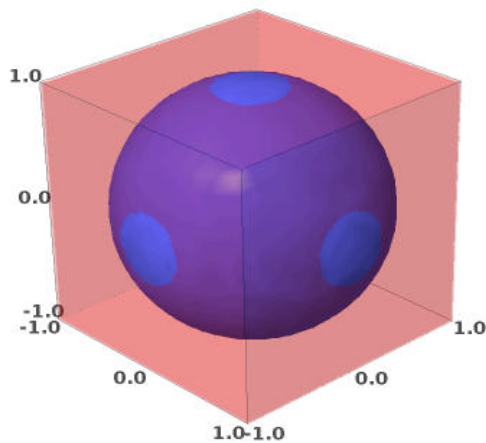
PDC2014B Home Assignment Number 1

3D Parallel Pi Monte-Carlo on a hybrid architecture (Xeon + Xeon Phi)

Guy Tel-Zur
March 2014

Goal: Find pi using parallel MC based on MPI

Governing equation: $\pi = 6N_s/N_c$



Algorithm:

rank 0 generates n random seeds and broadcasts them to all the tasks (ranks).

```
For each rank  $I$  between 0 to  $n-1$ 
  receive the seed
  set num_inside=0
  for  $j=0$  to  $N_c/n$ 
    generate a point  $(x,y,z)$  as follows:
     $x$  between  $[i/n, (i+1)/n]$ 
     $y$  between  $[0, 1]$ 
     $z$  between  $[0, 1]$ 
    distance =  $x^2+y^2+z^2$ 
    if distance < 1 num_inside++
  send num_inside to rank 0
rank 0 accumulates all the num_inside into  $N_s$ 
rank 0 computes  $\pi = 6*N_s/N_c$ 
```

What to do:

1. compile and build pimc and pimc.mic executables in a similar way that was taught in the lab.

To enable communication between host and mic set:

```
export I_MPI_MIC=enable
```

do:

```
mpirun -n <# of processes> -host <hostname1> <application> \  
      : -n <# of processes> -host <hostname2> <application>
```

run it on the host and on mic0 and mic1

you can set:

```
export I_MPI_MIC_POSTFIX=.mic
```

try running again using a machinefile:

```
xphisrv2:2
```

```
mic0:3
```

```
mic1:5
```

scp the executable to the mic and run it from there

What to submit:

(in two weeks to guycomputing@gmail.com)

1. source code
2. instructions how to compile and execute
3. a table of timing results for the different modes (host only, symmetric mode (mixed) and a single mic and two mics) and number of tasks and internal division of tasks between the devices.
4. conclusions!

version: 13:30, 24-03-2014