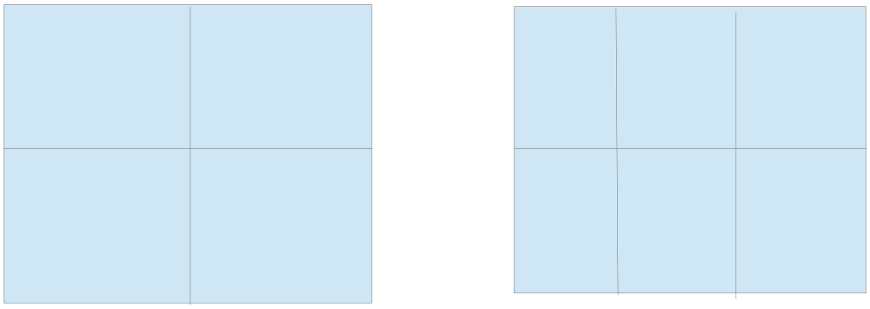
Summary Report

A distributed memory architecture code is solved to obtain velocity of random particles in a rectangular domain. The entire domain can be partitioned in both x and y direction. In each partition random particles are initiated and with the help of gradient function, the velocities in each particles are calculated. The the code is formulated in C and mpi is used for shared memory parallelization.

Equation to be solved: $v(x,y)=\blacktriangledown\psi\,(x,y)$
Length of 2D domain= -1.00 to 1.00 in both X and Y direction.
Number of random particles in each domain= 100.
Gradient of function is calculated analytically.

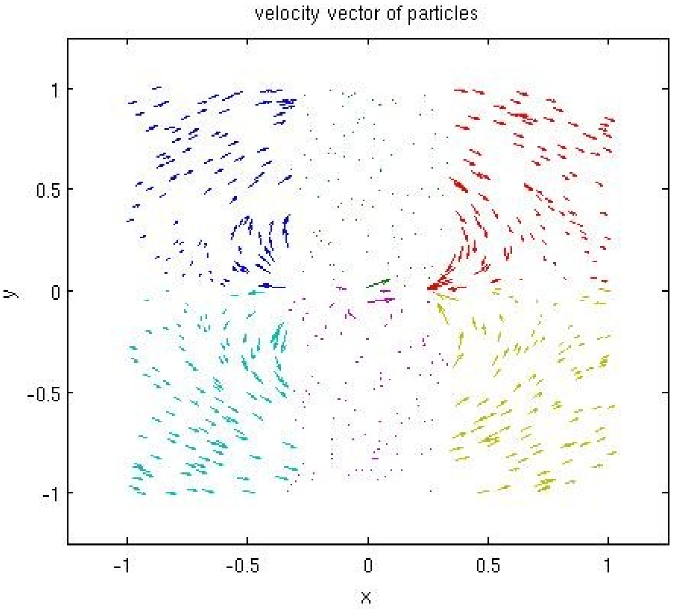
Partition Approach:
The following figure shows the partition approach of the domain for 4 and 6 number of processes.



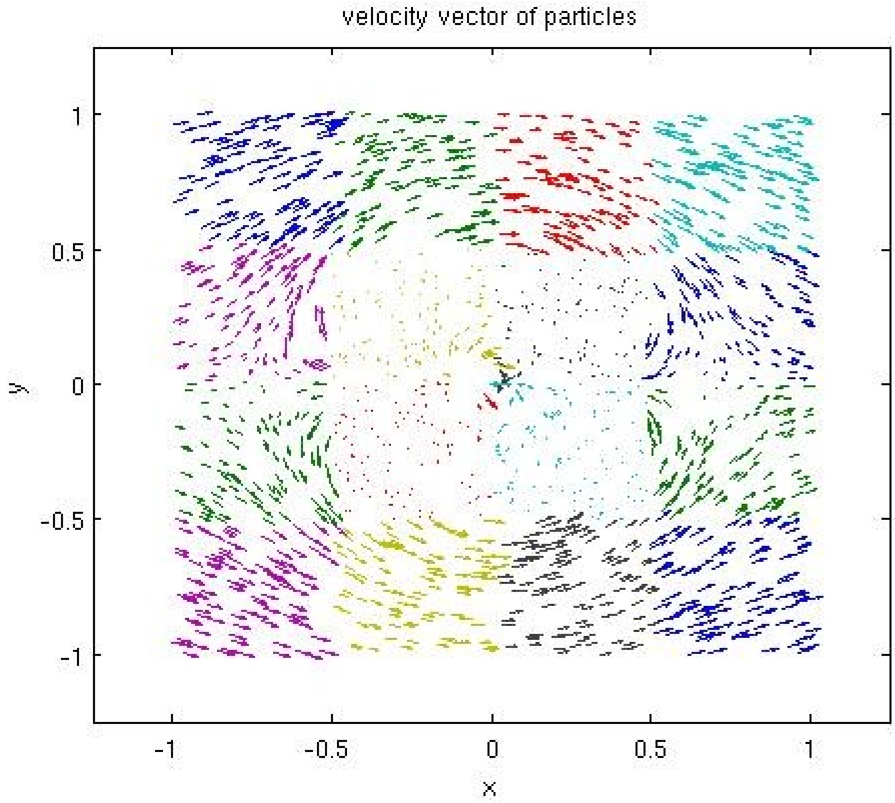
(a) 4 process (b) 6 process

Figure: partition of the domain in different processes

Result:



(a) number of process 6



(b) number of process 16

Figure 2. Particles with velocity vectors for different processes.

Minimum T:
For each case the minimum t is calculated. When the number of process was 4, the minimum t is obtained as 0.003610 sec. When number of process was 20 the minimum is obtained as 0.000179 sec.

The code was run in the gmice cluster however is was giving an error related to math.h. So it refused to compile over there. Afterwards, the code was ran in my personal laptop under debian ubuntu with gcc,icc compiler. The results presented are obtained from runs performed in my laptop.