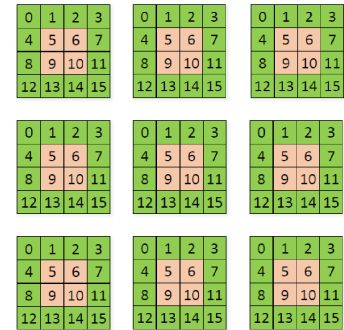


Assignment 2 (100 marks)

Modify your Assignment 1 halo exchange code to incorporate topology-aware hierarchical communication (Lecture 14).

The goal is to reduce the number of inter-node communications. For example, consider a 9-process decomposition of the 12x12 domain on the right. In the default case, there will be 0->3, 1->4, 2->5 communications from node 1 to node 2 (i.e. 3 simultaneous communications) in case of 3 nodes with ppn=3. You may select any rank as the leader rank on each node and reduce the number of inter-node communications (as discussed in class).



The assignment is to compare the performance of data exchange with and without leader ranks for 9-point stencil (you may use any datatype/mode of communication for the halo exchanges).

Perform the experiments for the following configurations. Note that you need to run 5 times on Prutor, rest of the configurations will be run automatically for the submission event corresponding to Assignment 2 on Prutor.

for execution in 1 to 5 // repeat each configuration 5 times – run on Prutor 5 times
// The below configurations will be run on Prutor when you “Execute”
for P (#processes) in 8, 12 [We’ll use Px=4] (note: we may test for higher process counts)
for N (double data points per process) in 4096^2, 8192^2 (note: we may test for larger data)
mpirun -np P -f hostfile ./halo Px N <num_time_steps> <seed>

You may use a variable ‘leader’= 1 for the case “with leader ranks” and ‘leader’ = 0 for the case “without leader ranks” within your code. Every process must randomly initialize the N data points that it owns at time step 0 using the ‘seed’ value given in the argument and the following equation.

```
srand(seed*(myrank+10));  
data[i][j] = abs(rand()+(i*rand()+j*myrank));
```

We will use 4 processes per node. Time the entire data exchange for all the time steps per configuration. Output 3 lines (1) “Time with leader =” time (2) “Time without leader =” time and (3) “Data=” data[0][0] i.e. the final value of the first element of your data array. *Any other output line in your final submission will attract a penalty of up to 5 marks.*

Plot the time (in seconds) for each data size per method per process count. Use boxplots (from the 5 executions) for every data point in the plot. Time in seconds in Y-axis and (P, N, leader/noleader) in X-axis. You should report the total time taken by the main halo exchange function, excluding initialization. Note that there are a total of 8 data points on the X-axis.

Execution and submission instructions

1. Submit your code on PRUTOR. Do not use the arguments pragma while collecting timing information and your final submission.
2. Submit GroupXY.zip to hello.iitk.ac.in (Assignment 2). It should necessarily contain the source code ('src.c'), 'readme.pdf', plot script and data file. The source code should be well documented. The 'readme.pdf' should contain an explanation of your code including mode of communications and explanation of the hierarchical communication that you may have adopted, the timing plot, your detailed observations regarding performance and compare the two cases with reasons, any optimizations you have used within the code (maximum 4 pages). You should also mention the contribution of each group member. Clearly mention your group number, member details (names, roll numbers, IITK email IDs) in your readme.pdf.

Due date: 16-04-2024 (There will be NO extensions)

Important Points

1. The above submissions must be made by only one group member. I will not consider any submissions via email, so please plan accordingly. We will consider the last submitted code on PRUTOR (before the deadline) as the final code. Multiple submissions (on MookIT/Prutor) from the same group will attract a penalty of up to 5 marks.
2. 5 bonus marks will be awarded for the fastest execution time on our (hidden) test cases.
3. It is recommended to start early to avoid high queue waiting times near the submission deadline. You may receive "Resource Busy" if there are no free nodes available to execute your job. It is your responsibility to plan early and collect the timings early enough. We will release the Assignment 2 event by April 5.
4. Incorrect submission name on MookIT will attract a penalty of up to 10 marks.
5. A bonus task (for strong scaling runs) may be released by April 6 (depending on the availability of a different resource). This will have a different submission event (possibly a different deadline) with no early credit component.