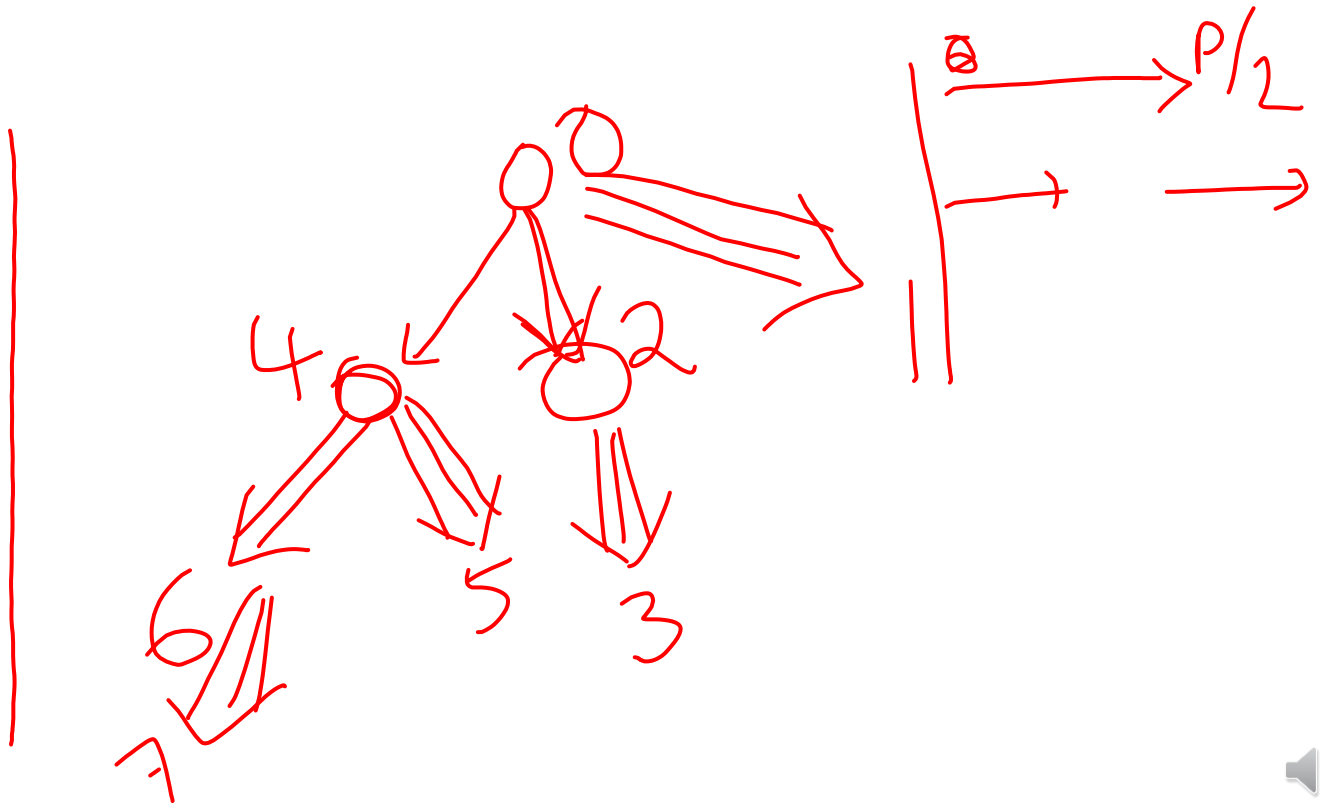
Quiz 3 Discussion

Apr 16, 2021

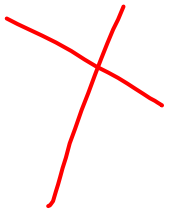
Number of steps required to broadcast 100 bytes from rank 0 to 1023 processes (total process count is 1024) using the binomial tree algorithm is

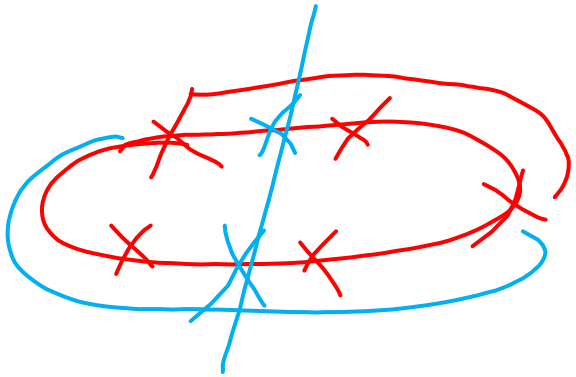
- 10
- 20
- 1023
- 32



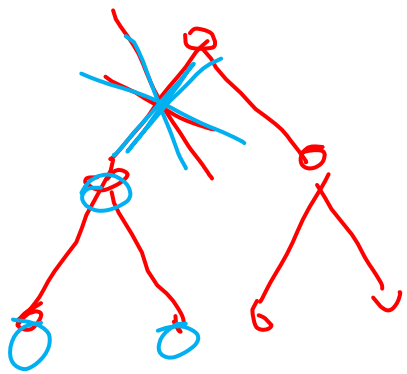
MPI_Send is a blocking call

- True
- Sometimes true?





Bisection width of a complete binary tree is



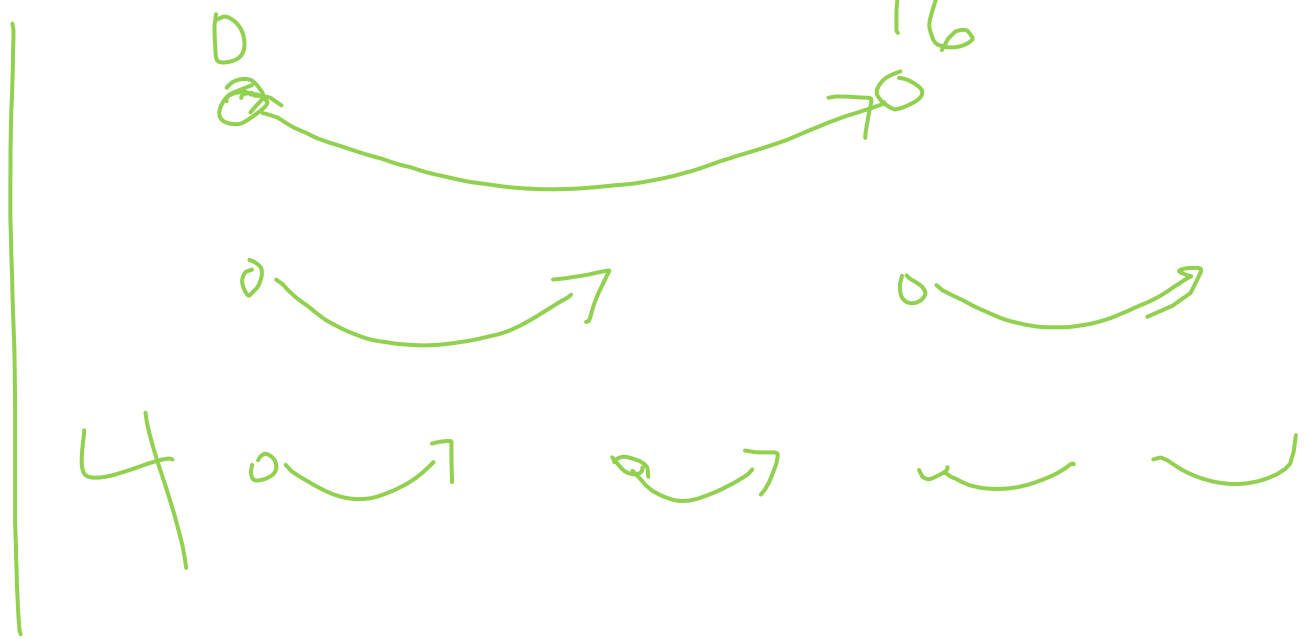
- 1
- Log p
- Number of leaf nodes
- 2 * log p
- n/2 where n is the number of processes
- Not possible. As it is not possible to divide a complete binary tree in two equal halves by cutting any number of links

Number of steps required to broadcast 100 bytes from rank 0 to 100 processes using the linear algorithm is

- P-1

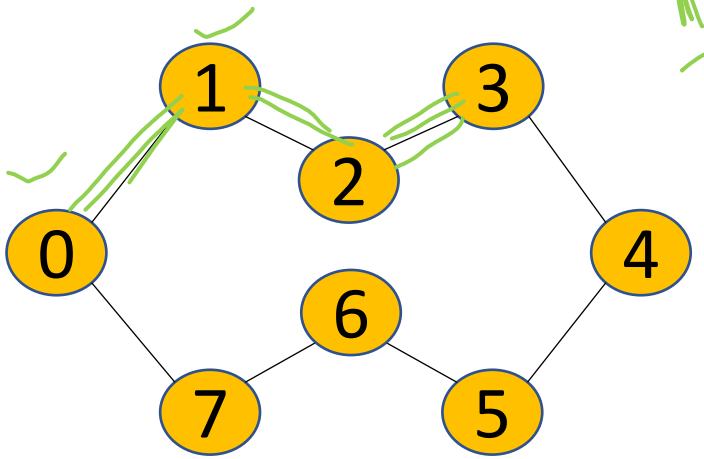
In step #3, the number of senders for MPI_Scatter of 2 MB among 32 processes, with root = 0 using recursive halving algorithm are

- 4



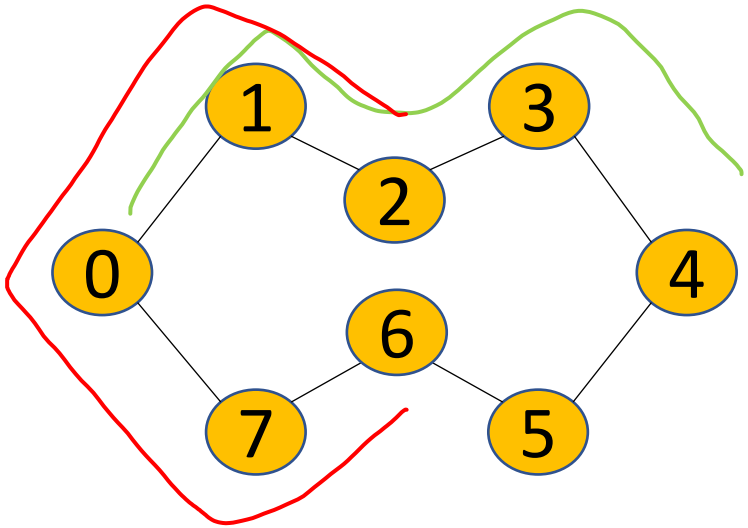
Total number of hop-bytes on the below network for the MPI_Allgather ring algorithm (root=0, #processes=8, message size per process=200 bytes) is

- 1400
- 1600
- ….



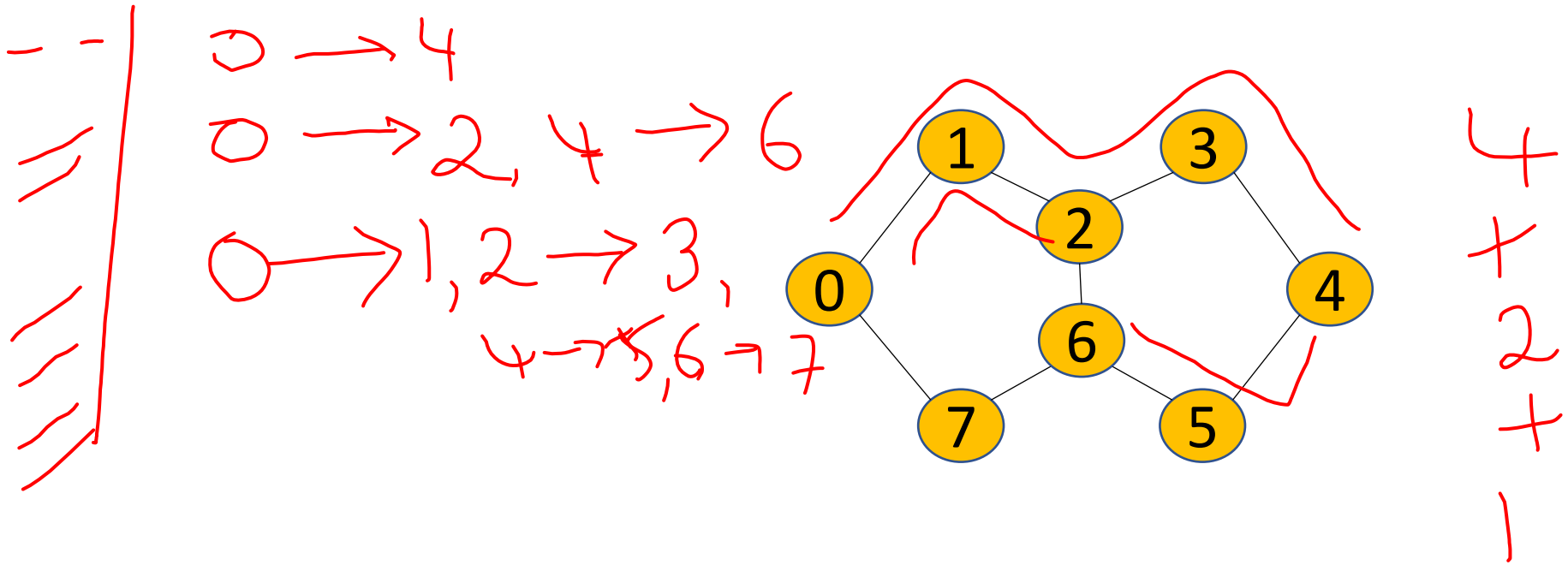
Diameter

- 4
- 2
- ....



Total number of hops on the below network for MPI_Bcast binomial algorithm (root=0) is

- 7 ✓
- 12
- ....



Number of steps required to broadcast 100 bytes from rank 0 to 99 processes (i.e. total process count is 100, root=0) using the linear algorithm is

- 99
- 9900

MPI_Isend is a blocking call

- False
- True

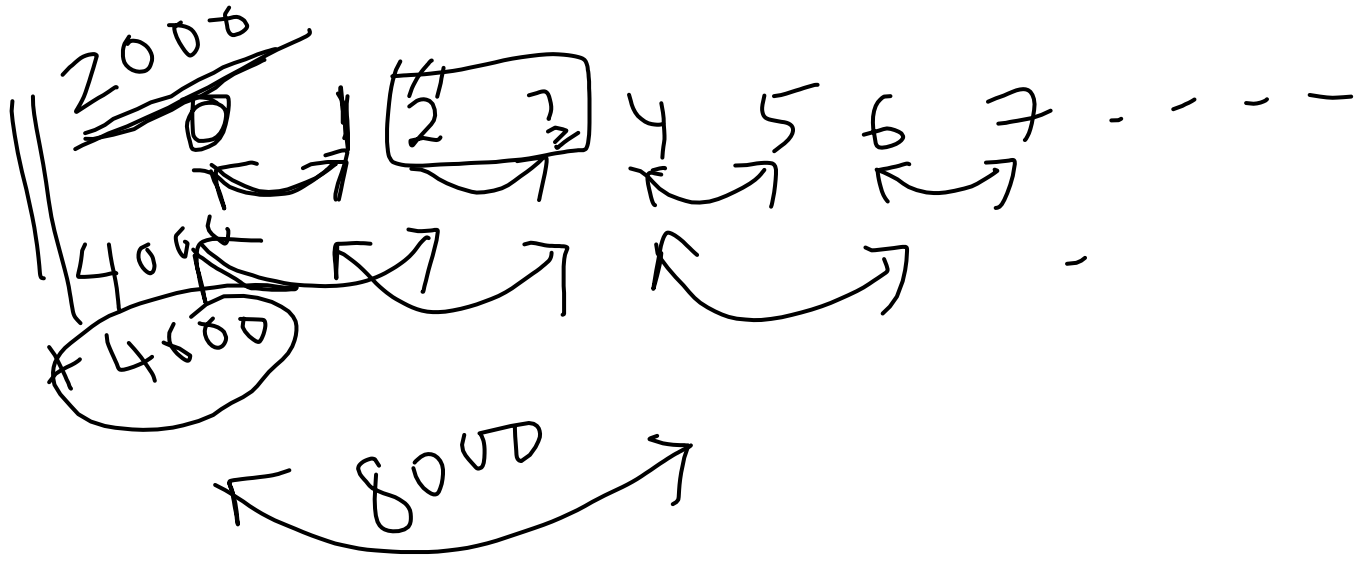
Number of steps required to broadcast 200 bytes from rank 0 to 511 processes (total process count is 512) using the binomial tree algorithm is

- 9
- 18
- …

$\log_2 512$

In the recursive doubling algorithm of MPI_Allgather (number of processes = 16, number of bytes per process = 2000), the number of bytes exchanged between communicating pair of processes in the 3rd step =

- 8000
- 2000
- …

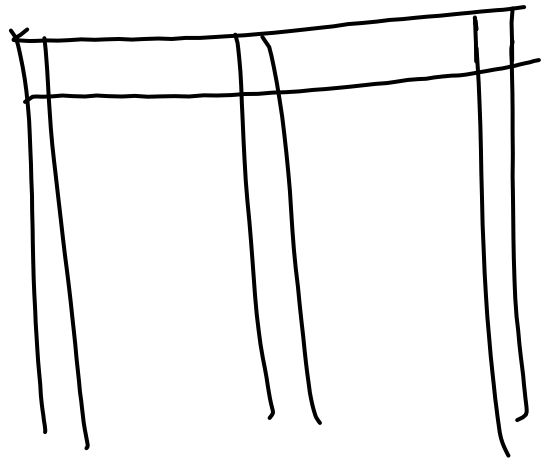


MPI_Send is usually faster than MPI_Ssend

- T
- F

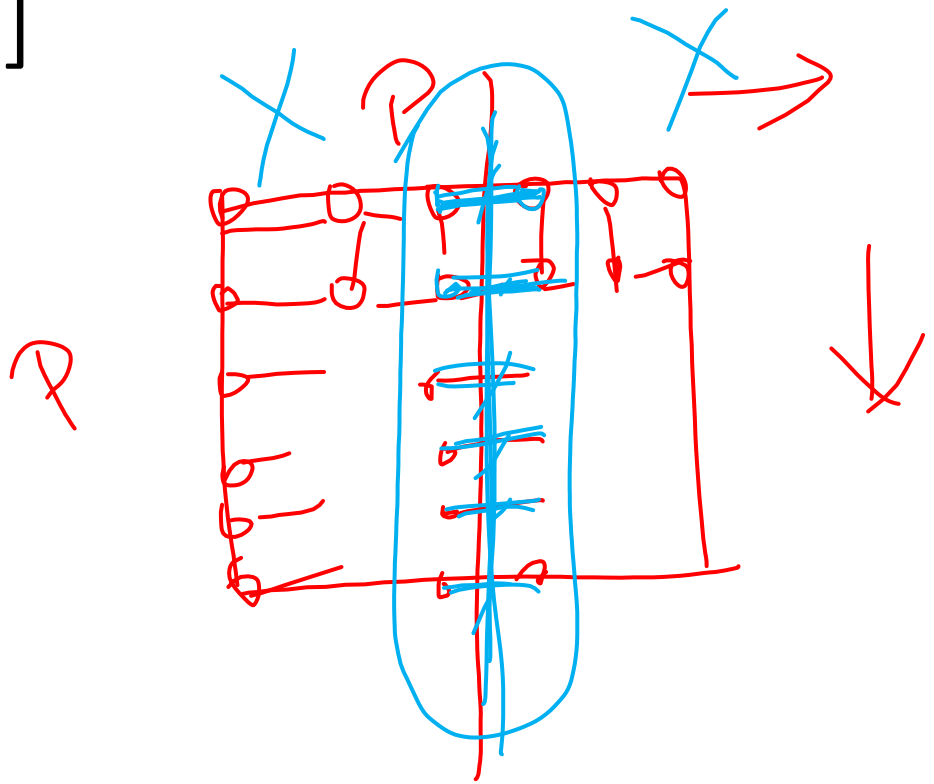
What MPI datatype is most suitable to communicate certain columns of a matrix?

- MPI_Type_vector
- MPI_Type_indexed
- Derived datatype ?

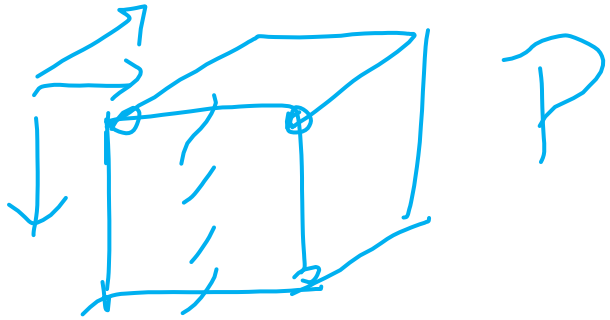


What is the bisection width of PxP 2D mesh network? [Assume P is even]

- P



What is the bisection width of PxPxP 3D mesh network? [Assume P is even]

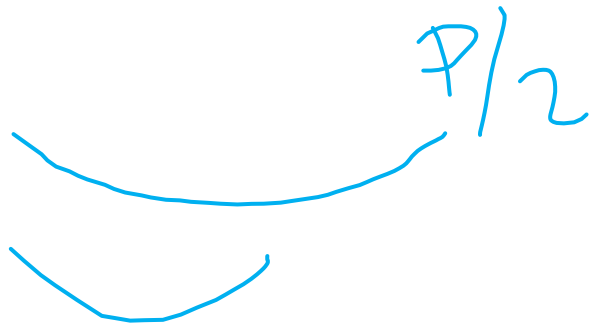


0

3(p-1) | P squared (P X P) | P/2 (P divided by 2) | P^2/4 | p * (p-1) | square root (p)

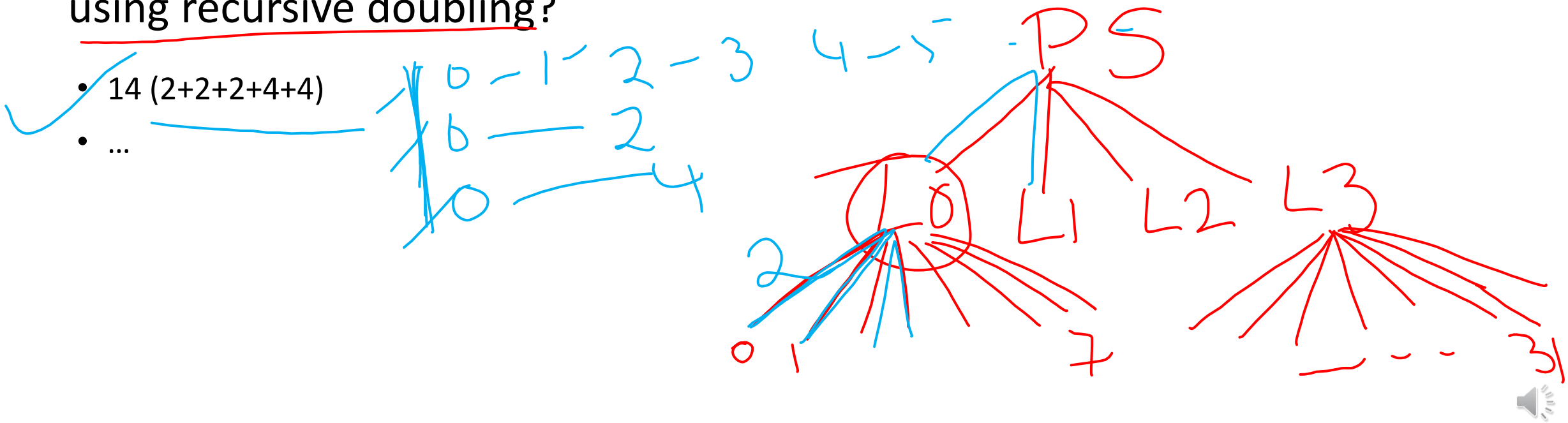
4 * square root of P | P*P*P/2 | P^(1/3) i.e. Cube r… | PxP = P^2 | p/2

In case of MPI_Scatter of 2048 KB among 16 processes, with root = 0, using recursive halving, select the correct option(s) (Assume steps are numbered 1,2,..)



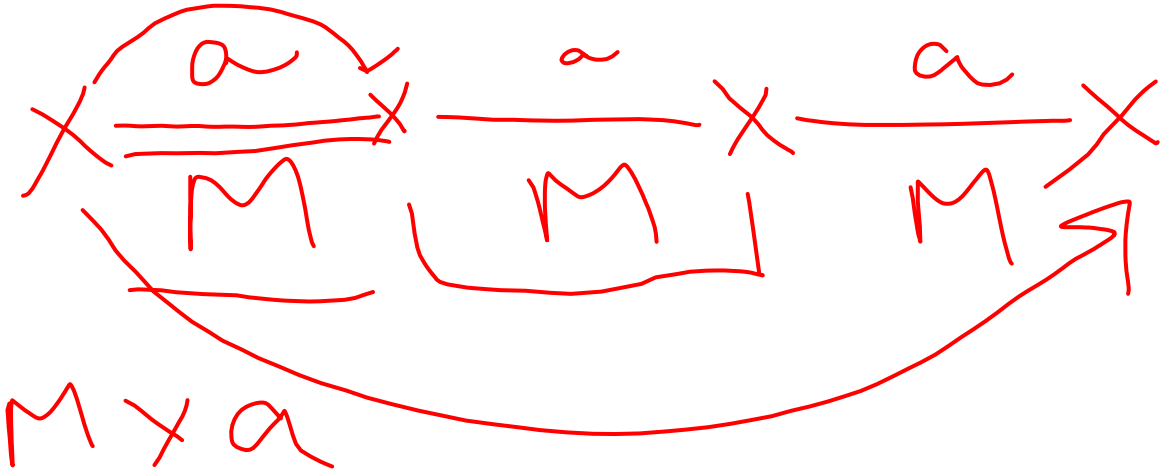
In a tree network, there are 4 leaf switches L0, L1, L2, L3. All leaf switches are connected to a parent switch. Assume that every leaf node is connected to 8 compute nodes (child nodes), i.e. a total of 32 nodes in the system. Assume that we are executing a 32-process job on this system, 1 rank on 1 node. The rank numbers are contiguously numbered on each leaf switch from left to right starting from L0 to L3. How many hops would be incurred on this network for MPI_Reduce using recursive doubling?

- 14 (2+2+2+4+4)
- …



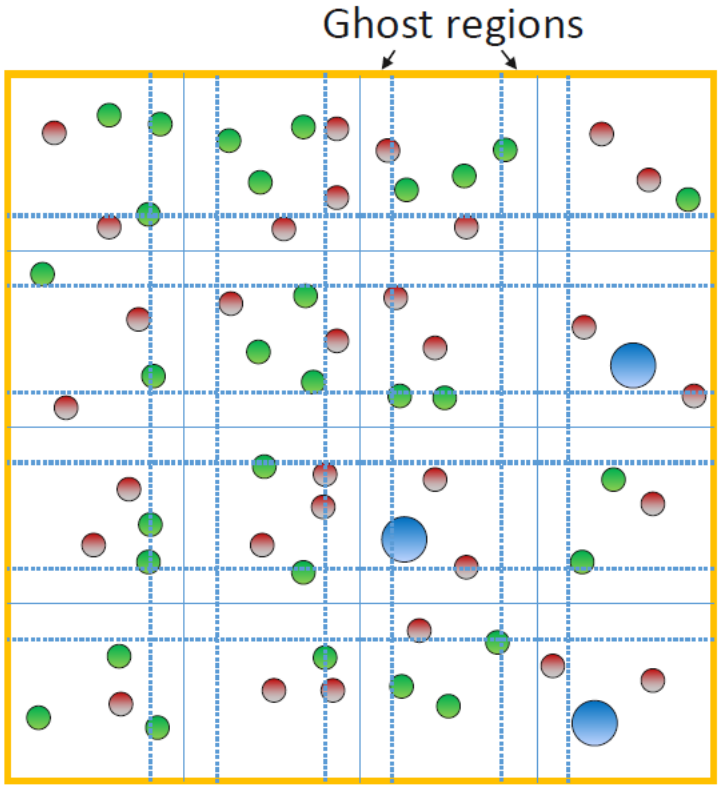
Consider a linear array interconnect LA of length M (i.e. M hops, M+1 nodes). Latency of LA is 'a' seconds. Bandwidth of every link in LA is 'b' bytes/second. What is the communication time for communicating 'n' bytes from the first node to the last node of this network?

- M (a + n/b)
- M*a + n/b
- a + (M)n/b ?
- M*a*b*n ?
- M*n/b ?



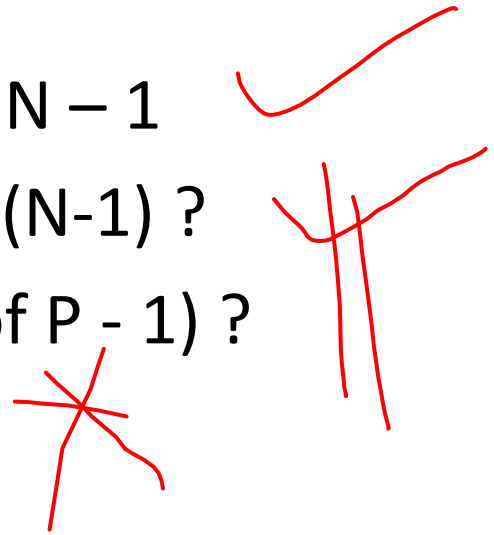
Assume that the virtual process topology is P x P as shown in the figure, and the physical process topology is a P x P mesh (assume bidirectional links). The communication pattern is a 5-point stencil, i.e. every process (including boundary processes) exchanges data with its four neighbors (with wraparound) at every time step. What is the maximum number of hops in a time step?

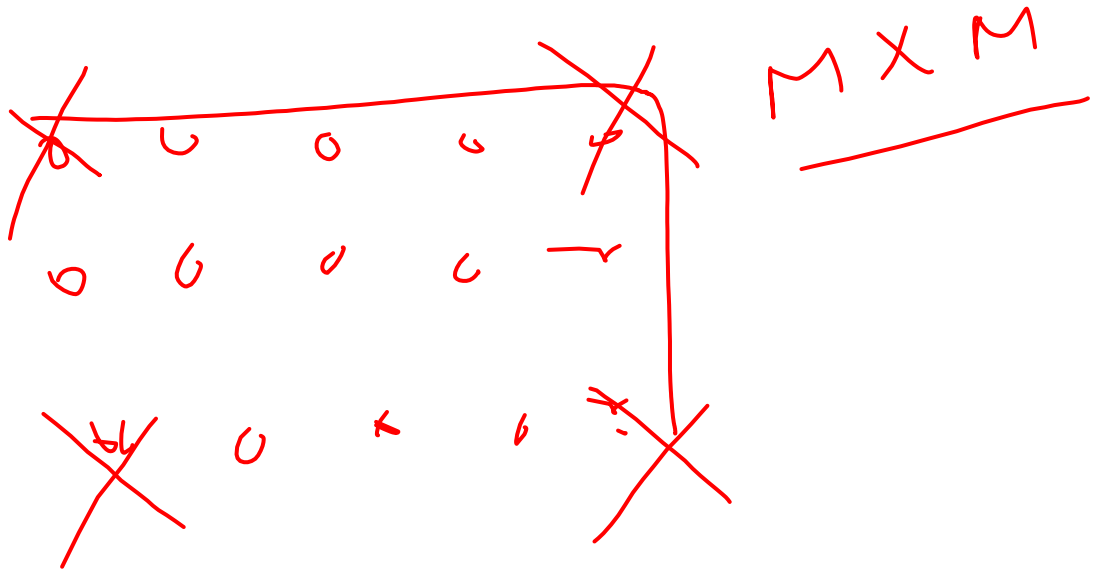
- P-1
- 4
- 16 ?
- 24 ?
- 4P^2 ?



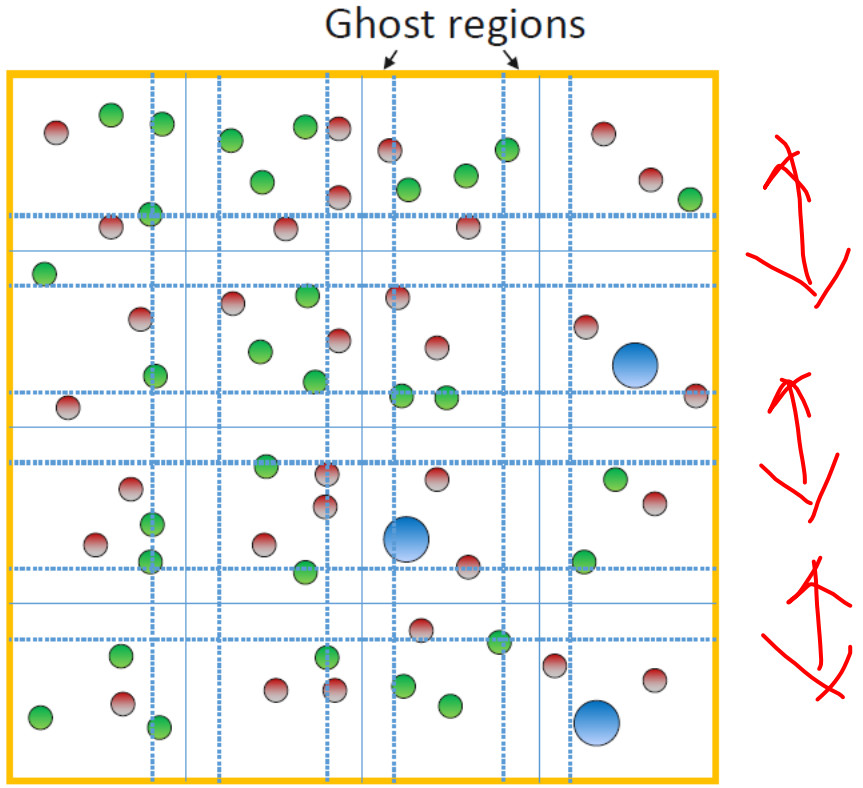
The diameter of a 2D mesh network of size M x N (where, M>N) is

- M – 1 + N – 1
- (M-1) * (N-1) ?
- 2(root of P - 1) ?



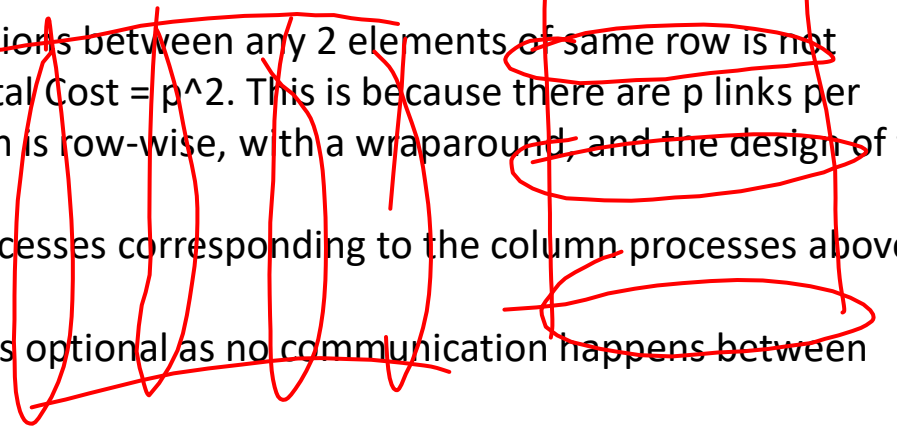


Assume that the process grid is P x P, and the average number of data points per process is M. Assume further that the average number of points in the ghost region is G. Assume that G points are exchanged with only the top and the bottom neighbors of a process, and assume that wraparound communications exist, i.e. all processes including boundary processes have top and bottom neighbors. There are only row-wise communications in this application. Suggest a cost-effective network topology (can be a new design too) that is best suited for this communication pattern, and why?



Your Answers

- Torus
- Mesh | P * P grid
- The connections between the different rows of the Torus can be cut off as the nodes present in a given row are communicating.
- The topology can be consider as multiple rings s.t each row process forms a ring in that order.
- P ring interconnects
- A ring would be good because there is wrap around and each process communicates only with its top and bottom process. The ring would be vertical connecting the processes from up to down. One process from each vertical rings can be connected with vertical rings of other process to ensure low cost.
- A 2-D torus would be efficient, but with only row-wise connections (connections between any 2 elements of same row is not needed, so connection is only present if they belong to 2 different rows). Total Cost = p^2. This is because there are p links per column, and p columns in total. This would be effective since communication is row-wise, with a wraparound, and the design of the aforementioned network precisely captures this requirement.
- 1. Nodes of each column of the above grid can be connected in ring. (All processes corresponding to the column processes above will be mapped to ring topology with P processes)
  2. One node from each such ring can together form a linear chain (this part is optional as no communication happens between columns (only between rows))
  This topology is cost-effective with only (P*(P-1) links) and if we don't even connect the rings then only P links are required. This topology is also best suited for the above case as every communication will require only 1 hop.
- 2d torus without communication link to left and right so basically multiple chain as it allows to communicate using wrap around which will be helpful here



Your Answers

- Fat binary tree?
- Star?
- Hypercube?
- Splitting MPI_COMM_WORLD ?
- A 3D torus with XZY plane can be used. XZY plane means the first P processes will be on XZ plane and then Y plane will be filled. As there are maximum row communications, every row communication will require 1 hop. There should be connection from top of the mesh to bottom node forming a torus, which will ensure 1 hop communication for boundary processes.
- 3D Torus Mapping of the form xzy. It will incurs less number of hops.
- The most cost effective network topology will be a simple 2-D grid, since the communications are between top and bottom neighbours only they will be able to communicate very easily as they are at the distance of 1 hops from each other. So a P*P 2-D grid. Further we can keep all the processes in a single column together under 1 switch, this will ensure that we have 0 inter-switch communications which are very constly and all the communications will be intra switch only. A P*P grid, where the members of one column are share the same switch.