## Q4

The paper of OpenMPI on million cores analyzes the downsides of MPI scaling on millions of cores. Here we would do some comparison and analysis.

The paper points out the irregular collectives. MPI implementation must scan through the entire array of data sizes to know which processes have nonzero data, and the user must allocate and initialize this array. On large numbers of processes, the time to read the entire array itself can be large and increases linearly with system size. And this issue is also discussed in other papers like Sparse collective operations for MPI. The paper talked about parse collective operations, which could somehow optimize this kind of issue.

Then the paper talks about Graph topology. The issue is that the MPI specification requires the entire communication graph to be supplied on each process. The optimal way is to use new graph topology interface, MPI_Dist_graph_create in new version of MPI

Then the paper focuses on the functionality for forming arbitrary subsets of process groups and building communicators. Further improvement is talked about in compact representations for process groups in recent research.

Then comes about the fault tolerance. The paper illustrates that some of the interfaces for detecting and reporting errors in MPI are not scalable. Recent search has improved the fault tolerance by extending the user-defined error handler mechanism to provide an iterator interface for accessing error codes, which may reduce the memory required to represent the error codes at the cost of requiring the application to iterate over each error code.

The final issue would be collective communication. The reason it would be an issue would be that The collective communication operations in MPI have blocking semantics, and slight load imbalances or delays can lead to significant process synchronization waiting times at large scale. A current great resolution may be to design algorithms for collective operations that are more resilient to non synchronized process arrival patterns.

**REFERENCE**:
1. https://ieeexplore.ieee.org/document/5160935

2. https://dl.acm.org/doi/pdf/10.1145/1851476.1851507