Wireless Network as a Multicasting Channel for MPI IO

Wenguang CHEN, Wei XUE, Jidong ZHAI and Weimin ZHENG*

Departemt of Computer Science and Technology, Tsinghua University, Beijing, China

1. COLLECTIVE I/O

MPI IO is designed to leverage the parallelism of parallel file systems to access large files quickly. Comparing to non-collective I/O, collective I/O operations can be implemented with more scalable communication routines and enable better coordination among processes.

Current implementation of collective operations usually uses recursive doubling, binomial tree or similar approaches to perform a gathering or multicasting operation, which requires at least O(logP) steps on packet switching network, where P is the number of processes. An all to all operation among P processes would require O(P) steps. When the number of processes is ever increasing, it is desired to find an approach that could significantly reduce the cost of these collective operations. Although there are many efforts on optimizing the performance of MPI collective IO, they face the inherent limitation of packet switch networks.

In this Work-In-Progress paper, we propose to explore the potential of wireless network, which offers O(1) broadcasting/multicasting naturally.

2. WIRELESS NETWORK

The common sense on wireless network is that it is much slower than wired network. However, modern wireless network is evolving radically. For example, 802.11ad, a recent WiFi standard using the 60GHz band, supports up to 6.76Gbps bandwidth.[1]. One of the local network using light, can already achieve the speed of 10Gbps now, and is planning to support 40Gbps.[2]

Comparing with the mainstream wired network for HPC systems, which is 10Gbps-56Gbps, the wireless network provides lower but comparable bandwidth. As the wireless network is much more efficient in broadcasting/multicasting, it is promising to explore the application of wireless network on HPC systems.

Recent research work uses wireless network to mitigate the communication bottlenecks between machine cabinets by offering additional point to point links, but they don't use the wireless network as a channel for fast multicasting.[3][4]

3. OPTIMIZING COLLECTIVE OPERATIONS

It should be noticed that wireless network is mainly good at broadcasting/multicasting. It is complementary to the wired packet switching network, not a replacement. Thus we need a hybrid network architecture that include both wired and wireless network.

Let's consider how to leverage the O(1) broadcasting ability to optimize current MPI collective operations. For some of them, only substitute the original broadcasting implementation with the O(1) implementation is sufficient. For example, *Allreduce* is a *reduce* followed by a *Bcast*. The implementation of reduce can not be optimized with wireless network, but the *Bcast* can benefits. For other operations, such as *Alltoall* or *Scatter*, the fast broadcasting enables new implementation algorithms. For example, *Alltoall* can be implemented as a *Gather* followed by a *Bcast*, where the *Gather* get all message to one process, and the *Bcast* to send the whole data to all processes. The new algorithm only needs *LogP* steps, while the existing *Alltoall* algorithms usually require *O(P)* steps. There's a limitation in this algorithm though, because all messages must fit in the memory of a single node. It can be improved further by leveraging multiple nodes to allow all messages fit in their memory. We omit detailed discussion due to space limitation. Table 1 compares the complexity of MPI collective operations with and without O(1) broadcasting ability.

Table 1. Complexity of MPI collective operations

	Without O(1) Bcast	**With O(1) Bcast**
Barrier	O(LogP)	O(LogP)
Bcast	O(LogP)	O(1)
Allgather	O(LogP)	O(LogP)
Gather	O(LogP)	O(LogP)
Reduce	O(LogP)	O(LogP)
Allreduce	O(LogP)	O(LogP)
Alltoall	O(P)	O(LogP)
Scatter	O(LogP)	O(1)

Since these collective operations are used in MPI IO implementations, such as ROMIO, optimizing these operations would provide better performance and scalability to MPI IO.

The benefit of O(1) multicasting is not limited to MPI IO. It is also useful to reduce the communication overhead caused by non-IO operations in MPI. What's more, the technique could find more applications in data center, such as VM deployment, building replicas in distributed file systems.

We are evaluating the scheme in MPI simulators, as well as establishing prototypes to quantitatively evaluate the idea to get more quantitatively results.

4. REFERENCES

[1] IEEE Standard 802.11ad, http://standards.ieee.org/findstds/standard/802.11ad-2012.html

[2] Li-Fi Internet Solution, *http://in.rbth.com/economics/2014/07/01/li-fi_internet_solution_from_russian_company_attracting_foreign_cli_36347.html

[3] Halperin, D., et al. Augmenting data center networks with multi-gigabit wireless links. In Proc. of SIGCOMM (2011)

[4] Zhou, X. et al. Mirror Mirror on the Ceiling: Flexible Wireless Links for Data centers, In Proc. of SIGCOMM(2012)

* Corresponding author: zwm-dcs@tsinghua.edu.cn