Acceleration of All-to-all Communication on Multi-Layer Full Mesh, Low-cost Connectable Network Topology

Toshihiro Shimizu
Fujitsu Laboratories Ltd.
Kawasaki,Japan
Email: t.shimizu_2@jp.fujitsu.com

Masahiro Miwa
Fujitsu Laboratories Ltd.
Kawasaki,Japan
Email: masahiro.miwa@jp.fujitsu.com

Kohta Nakashima
Fujitsu Laboratories Ltd.
Kawasaki,Japan
Email: nakashima.kouta@jp.fujitsu.com

***Abstract*—Recently, due to the increasing scale of computation, more and more servers are used to make calculations simultaneously and communicate with each other. These servers are connected by links and switches.**

Since the cost of the switch is relatively high, reduction of the number of switches is desirable to realize a larger cluster systems cost-efficiently. We have already proposed the multi-layer full mesh (MLFM) topology [1] in this purpose. In the article [1], we showed that MLFM topology can connect more servers than conventional fat tree topology and can achieve congestion-free all-to-all communication using all servers.

In this paper we propose a method of congestion-free all-to-all communication using part of the servers on MLFM. This is necessary because users of supercomputer systems typically use part of the servers rather than all of them. Our experimental results show 2.2 times higher throughput compared to the conventional communication pattern.

1. Introduction

In conventional cluster supercomputers, fat tree topology (Fig.1a) is typically used to connect servers. It has multiple pathes between servers and provides a high performance in high-load communication, such as all-to-all communication.

All-to-all communication is a type of high-load communication that sends data from each servers to all of the servers in a certain order. If the number of servers to submit the job is N, all-to-all communication consists of N phases. After the communication, each server obtains data from all of N servers. In other words, if all-to-all communication is expressed as a table shown in Fig.1b, where i-th row of the j-th column corresponds to the destination of the server sent from server j in phase i, for each row and column the numbers from 0 to $N-1$ appear exactly once.

If multiple communications pass through same link, link congestion occurs and the performance degrades. Therefore, avoiding link congestion is required. The fat tree can avoid link-congestion adapting the route as shown in Fig.1a.

Congestion-free all-to-all communication on the fat tree is possible by adapting the shift communication pattern. The rule of transfering data is that the destination of j-th server in the i-th phase is $(i+j)\%N$-th server. Fig.1b shows the destination of the shift communication pattern.

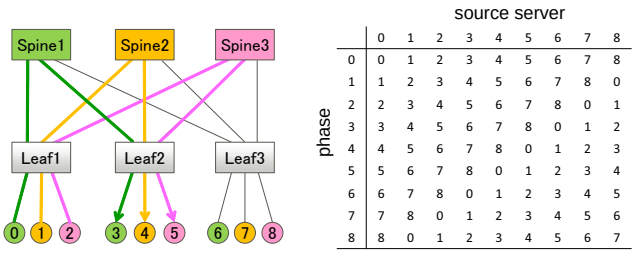


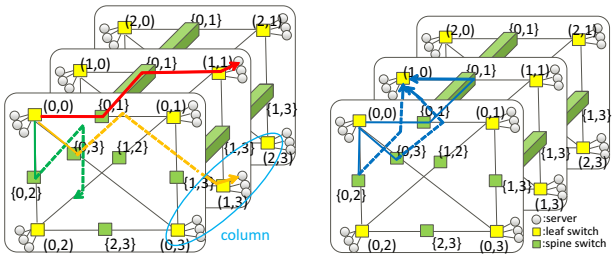
Figure 1. Fat tree and all-to-all communication

However, the fat tree requires many switches to connect servers. As a new topology that uses fewer switches, we proposed Multi-layer full mesh(MLFM) and the method of sending data to achieve congestion-free all-to-all communication using all servers [1]. MLFM is a network topology extending the full mesh topology [2]. MLFM topology enables the connection of more servers with fewer switches compared to the fat tree while maintaining the performance of all-to-all communication.

MLFM using $2d$-port switches has d layers and $d+1$ leaf switches in each layer. For each leaf switch, d servers are connected. Spine switches are inserted between each pair of leaf switcheon the same layer and these spine switches cross the layers. Fig.2 shows an example of the MLFM when $d=3$. The MLFM has $d^2(d+1)$ servers and $d(d+1)$ leaf switches, and $d(d+1)/2$ spine switches. Therefore, the MLFM using 36-port switches can connect 6156 servers while the fat tree can connect 648 servers.

However, we have not proposed a method for submiting jobs to partial servers nor the implementation and evaluation of this topology and method.

In this paper we propose a method of congestion-free all-to-all communication among partial servers. We also experiment using our method, and our experimental results show 2.2 times higher throughput compared to the conventional communication pattern.



(a) to the different column (b) to the same column

Figure 2. Multi-layer full mesh topology and communication pattern

2. Our Proposal

First, we assign id to each layer, switch and server of MLFM, where each enumeration is 0-based. We call the set of leaf switches of the same position "column". (See Fig.2) We assign the id (i, j) to j-th leaf switch of i-th layer and the id (i, j, k) to k-th server of the leaf switch. We assign the id $\{j_0, j_1\}$ to the spine switch that connects j_0-th and j_1-th leaf switch in each layer. (See Fig.2)

We can select $n\ell m$ servers of server-(i, j, k) with $0 \leq i < n$, $0 \leq j < \ell$, $0 \leq k < m$, where n, ℓ, m are any integers satisfying $1 \leq \ell - 1 \leq m \leq d$ and $1 \leq n \leq d$. Changing the value of n, ℓ, m, we can change the scale of all-to-all communication.

We propose a method of sending data to conduct congestion-free all-to-all communication among these selected servers. There are $n\ell m$(the number of selected servers) phases. We name each phase (s, t, u) where $0 \leq s < n$, $0 \leq t < \ell$ and $0 \leq u < m$. We set two type of communication in accordance with the value of t.

The first communication is to the "different column". This pattern is conducted when $t \neq 0$. The destination of server (i, j, k) is server $((i+s)\%n, (j+t+k+1)\%\ell, (k+u)\%m)$ if $t + k + 1 < \ell$ and $((i + s)\%n, (j + t + k + 2)\%\ell, (k + u)\%m)$ if $t + k + 1 \geq \ell$. The example of the communication from leaf switch $(0, 0)$ with $(s, t) = (1, 3)$ is shown in Fig.2a.

The second communication is to the "same column". This pattern is conducted when $t = 0$. The destination of server (i, j, k) is server $((i+s)\%n, j, (k+u)\%m)$ via spine switch $\{j, (j+k+1)\%\ell\}$. The example of the communication from leaf switch $(0, 0)$ with $(s, t) = (1, 0)$ is shown in Fig.2b.

Conducting these communications for all possible pattern of (s, t, u) following the above rules we complete All-to-all communication and these communication patterns avoid link congestion.

3. Evaluation

We evaluate the performance of our method on MLFM by experiment using actual switches and servers.

We construct the topology of Fig.2 using 32 servers(VMs) with physical HCA ports, where, for the last layer, we connected two servers to each of the leaf switches. We used FDR and QDR Infiniband switch for spine and leaf, respectively. We evaluate the following scales of all-to-all communication of partial servers, 12 servers ($n = 2$, $\ell = 3$, $m = 2$), 18 servers ($n = 3$, $\ell = 3$, $m = 2$), 24 servers pattern A ($n = 3$, $\ell = 4$, $m = 2$), 24 servers pattern B ($n = 2$, $\ell = 4$, $m = 3$). We use OpenMPI and Intel MPI Benchmark(IMB) for the evaluation. OpenMPI has embedded message transfer algorithm called *pairwise*, which conducts shift communication pattern. We implemented the new transfer algorithm *mlfm* on OpenMPI. The Message sizes that each server sends are 16, 64, 256MiB per node. We measure and compare the throughputs of these two algorithm. We obtained the bandwidth 3770MiB/s (QDR link-up) from ping-pong communication. We set this value as optimal throughput of bandwidth.

Fig.3 shows that our method attains throughtput close to the optimal throughput while shift communication is not. This result also shows that we achieved 2.2 times higher throughput improvement (24 servers pattern B).

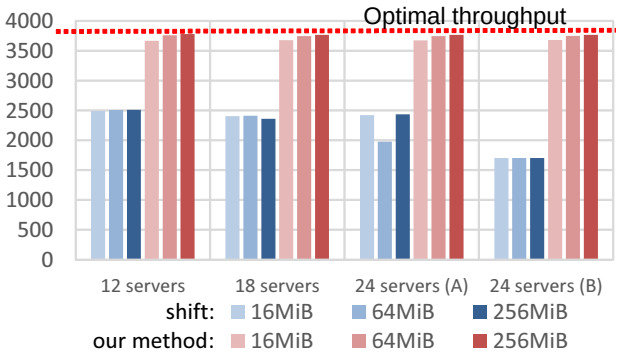


Figure 3. Result of experiment

4. Conclusion & Future work

We have proposed a method of congestion-free all-to-all communication among partial servers on MLFM. Our experimental result shows that we achieved 2.2 times higher throughput. The result is close to the optimal throughput,

Theoretically, we can achieve congestion-free all-to-all communication regardless of the number of switch ports. Therefore, our method is useful even in the large scale cluster.

We are planning to cope with the problem of fault tolerance since MLFM has few paths between servers.

References

[1] Fujitsu Laboratories Develops Technology to Reduce Network Switches in Cluster Supercomputers by 40%, Maintains network performance, lowers energy consumption, http://www.fujitsu.com/global/about/resources/news/press-releases/2014/0715-02.html, (2012).

[2] E.Totoni et al, Optimizing All-to-all Algorithm for PERCS Network Using Simulation, SC'11 Companion, 2011).