Performance and Memory Usage Evaluations for Channel Interface of Advanced Communication Primitives Library

Hiroaki Honda[a,b,1)], **Takeshi Nanri** [a,b,2)], **Yoshiyuki Morie** [a,b,3)]

[a] Research Institute of Information Technology, Kyushu University
6-10-1 Hakozaki, Higashi-ku, Fukuoka 812-8581, Japan,
[b] CREST-JST

[1)] honda.hiroaki.971@m.kyushu-u.ac.jp
[2)] nanri@cc.kyushu-u.ac.jp
[3)] morie.yoshiyuki.404@m.kyushu-u.ac.jp

ABSTRACT

Currently, there are many research projects developing exascale supercomputer systems. For the exascale computing, it is indispensable to use high performance communication library which is able to be used with up to 10 million processes. Many researchers believe one of leading candidate is Message Passing Interface (MPI) library. However, memory size required by the library will be larger than the available size for each process, because current MPI libraries, such as Open MPI and MPICH, require the memory footprints proportional to the number of processes [1]. Therefore, to realize exascale communication, memory usage of communication libraries must be reduced drastically. To overcome this difficulty, our research group has been developing a new communication library, Advanced Communication Primitives (ACP). ACP library prepares allocation and de-allocation procedures for each of primitive communication interfaces. Therefore, it is expected that applications can consume just enough memory for communication. Internal structure of ACP library consists of the basic layer and the middle layer: the basic layer is an abstraction of underlying interconnect such as Ethernet, Infiniband and Tofu. Since it supports RDMA model, this layer can be implemented efficiently. On the other hand, the middle layer is prepared as the collection of primitive interfaces for programmers of applications and other middleware.

Channel interface is proposed as one of the communication interfaces of the middle layer based on the message passing method via channels. A channel is a logical path constructed between a sender process and a receiver process [3]. Currently, channel interface supports single-directional data transfer between a pair of the sender and the receiver. In addition to that, messages are transferred in-order on the channel. These features help the implementation of the interface to achieve high performance with low memory consumption.

In this paper, we report the results of preliminary evaluation of the performance and the memory usage of the channel interface. For obtaining bandwidth, we measured latencies of ping-pong type data transfer on InfiniBand. In the implementation at this time, the channels accept up to 16KB of data as one message, and the bandwidth with that size is 2.96 Gbps. The available data size and the bandwidth will be increased by the symposium. For memory usage, we confirmed that each channel consumes about 4560 byte on the sender and about 19104 byte on the receiver. These sizes change as the size and the number of the slots of each buffer are modified. In addition to that, these consumed memory could be freed completely by using the de-allocation function.

REFERENCES

[1] S. Sumimoto *et al.*, ACP: Advanced Communication Primitives for Exa-scale Systems, International Meeting on High-Performance Computing for Computational Science (VECPAR2014), Eugene, Jul.2014.
[2] ACE Project, available (on line), <http://ace-project.kyushu-u.ac.jp/index.html>.
[3] T. Nanri *et al.*, Channel Interface: a Primitive Model for Memory, 23rd Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP2015), Turku, Mar.2015.