



The Ministry of Education and Science of the Russian Federation
Lobachevsky State University of Nizhni Novgorod
Computing Mathematics and Cybernetics faculty

The competitiveness enhancement program of the Lobachevsky State University
of Nizhni Novgorod among the world's research and education centers

Strategic initiative “Achieving leading positions in the field
of supercomputer technology and high-performance computing”

Parallel Programming
for Multiprocessor Distributed Memory Systems

02 Lecture
Point-to-Point Communications

Brief description

Nizhni Novgorod
2014

02_LECTURE. COLLECTIVE AND POINT-TO-POINT COMMUNICATIONS

OBJECTIVES

An objective of the lecture is to discuss the main types of collective communication operations in MPI as well as the types of point-to-point message transmission functions including nonblocking versions.

ABSTRACT

The mostly used collective communication operation is discussed. An example of MPI program that use data broadcast and reduction operations is considered as well as an example that use data scattering function. All supported types of point-to-point communications in MPI are considered including nonblocking communications as a way of combining the computations and the execution of the communication operations.

BRIEF OVERVIEW

The first part of the lecture is devoted to collective data transmission operations. The section contains description of mostly used collective communication functions as well as examples that demonstrates data broadcasting, reduction and scattering operations.

The section starts from the revision of data broadcast (MPI_Bcast) and reduction (MPI_Reduce) operations, their parameters and schemes of execution. Further the calculation of π number by means of some integral is discussed. The method of rectangles is used for numerical integration. Cyclic scheme of calculating distribution between processes is considered. Such a scheme is used rarely in MPI program because of data distribution incompatibility. Calculating the π number is an example where data distribution consists in broadcasting the number of intervals only. The implementation demonstrates reduction function usage also. The MPI_SUM operation in MPI_Reduce is used to calculate the estimation of π number using partial sum computed by each process. Full code of program is given.

Next part of the first section in the lecture is devoted to scattering (MPI_Scatter, MPI_Scatterv functions) and gathering (MPI_Gather, MPI_Gatherv functions) of data. The schemes of the functions are demonstrated and their parameters are described. The difference between scattering and broadcasting as well as between gathering and reduction is discussed. Scat-

tering technique is illustrated by the second example in this lecture that calculates the inner product.

At the end of the first section the all-to-all communications are discussed: gathering (MPI_Allgather), reduction (MPI_Allreduce) and data exchange (MPI_Alltoall).

Second section of the lecture starts from the discussion of possible sending operations implemented in MPI. Standard (MPI_Send), synchronous (MPI_Ssend), ready (MPI_Rsend) and buffered (MPI_Bsend) modes are described. At the end of the lecture nonblocking point-to-point operations are considered including four send-type functions, receiving function MPI_Irecv and checking/waiting functions (MPI_Test, MPI_Wait, ...). The possible scheme of combining the computations and data transmission is demonstrated. At the end of section simultaneous sending and receiving functions is considered (MPI_Sendrecv and MPI_Sendrecv_replace).

FOR STUDENTS

There are a number of sources, which provide information about MPI. First of all, this is the internet resource, which describes the standard MPI: <http://www.mpiforum.org>. One of the most widely used MPI realizations, the library MPICH, is presented on <http://www.mpich.org/>.

The following works may be recommended: Pacheco (1996), Snir, et al. (1996), Group, et al. (1999a). The description of the standard MPI-2 may be found in Group, et al. (1999b). The description of standard MPI-3 may be found at www.mpi-forum.org/docs/mpi-3.0/mpi30-report.pdf.

We may also recommend the work by Quinn (2004), which described a number of typical problems of parallel programming for the purpose of studying MPI.

REFERENCES

1. The internet resource, which describes the standard MPI: <http://www.mpiforum.org>.
2. One of the most widely used MPI realizations, the library MPICH, is presented on <http://www.mpich.org>.
3. Quinn, M.J. (2004). Parallel Programming in C with MPI and OpenMP. – New York, NY: McGraw-Hill.
4. Pacheco, P. (1996). Parallel Programming with MPI. - Morgan Kaufmann.
5. Snir, M., Otto, S., Huss-Lederman, S., Walker, D., Dongarra, J. (1996). MPI: The Complete Reference. – MIT Press, Boston, 1996.

6. Group, W., Lusk, E., Skjellum, A. (1999). Using MPI – 2nd Edition: Portable Parallel Programming with the Message Passing Interface (Scientific and Engineering Computation). – MIT Press.
7. Group, W., Lusk, E., Thakur, R. (1999). Using MPI-2: Advanced Features of the Message Passing Interface (Scientific and Engineering Computation). – MIT Press.

EXERCISES

1. Develop a sample program for each collective operation available in MPI.
2. Develop the implementations of collective operations using point-to-point communications. Carry out the computational experiments and compare the execution time of the developed programs to the functions of MPI for collective operations.
3. Develop a program, carry out the experiments and compare the results for different algorithms of data gathering, processing and broadcasting (the function **MPI_Allreduce()**).

TEST QUESTIONS

1. Select correct statements:
 - a. All application processes participate in collective operations
 - b. (+) Only processes of some communicator participate in collective operations
 - c. (+) A function corresponding to a collective operation must be called by every process, possibly with its own parameters
 - d. (+) MPI_Barrier is a collective operation
 - e. MPI_Send is a collective operation
2. Collective data transfer operations must include as participants:
 - a. (+) All processes of the used communicator
 - b. Some of the processes of the used communicator
 - c. All processes of used group of processes
3. MPI collective operations:
 - a. (+) May be implemented using point-to-point operations, but most likely such solution will be ineffective
 - b. Cannot be implemented using point-to-point operations in principle
 - c. Can be implemented using point-to-point operation, but not fully
4. Collective operations in MPI are:
 - a. (+) Data transfer operations having all processes of used communicator as participants
 - b. Operations on groups of processes

- c. Operations on communicators
5. MPI_Reduce data reduction operation can be described as:
 - a. (+) A data transfer operation in which collected values are processed during transfer and the result of processing is given to the root process only
 - b. A data transfer operation in which collected values are processed and partial values of reduction results are given to all processes of the parallel program
 - c. A data transfer operation, in which collected results are processed in some way, and the result of processing is given to all processes
 6. Data broadcasting operation is:
 - a. (+) An operation when the root process sends values to other processes and all processes receive all the data that was sent
 - b. An operation when root process sends values to other processes and all processes receive only part of the original data
 - c. An operation when the root process sends different values to other processes
 7. Data transfer operations in MPI include
 - a. (+) point-to-point and collective operations
 - b. paired and group operations
 - c. individual and collective operations
 8. In the Synchronous mode of point-to-point data transmission the function of data sending is terminated
 - a. (+) when the confirmation of the reception beginning for the transmitted message comes from the receiving process
 - b. if the message reception operation has already been initiated
 - c. after the message has been copied in the system buffer
 9. In the Buffered mode of point-to-point data transmission the function of data sending is terminated:
 - a. (+) after the message has been copied in the system buffer
 - b. when the confirmation of the reception beginning for the transmitted message comes from the receiving process
 - c. if the message reception operation has already been initiated
 10. The Ready mode of point-to-point data transmission the function of data sending may be used only:
 - a. (+) if the message reception operation has already been initiated
 - b. if the size of the message is less than the size of the system buffer
 - c. if the message reception operation will be initiated after data sending start