

Задание на курсовой проект по дисциплине “Теория функционирования распределенных вычислительных систем”

Задание

1. В соответствии со своим вариантом реализовать в стандарте MPI алгоритмы коллективных обменов информацией между ветвями параллельных программ (MPI Collective Communication).

2. Построить графики зависимости времени выполнения алгоритмов от размера передаваемого сообщения на подсистемах следующих конфигураций: 4 узла по 8 процессорных ядер; 8 узлов по 4 ядра; 8 узлов по 8 ядер.

Требования к реализации алгоритмов

1. Алгоритмы должны быть реализованы на языке программирования C и поддерживать только MPI-интракоммуникаторы (MPI Intracommunicators).

2. При реализации алгоритмов допускается использование MPI-функций только с префиксом ‘P’: PMPI_Comm_size, PMPI_Comm_rank, PMPI_Sendrecv и т. д.

3. Необходимо корректно (в соответствии с описанием функции в стандарте MPI 3.0) обрабатывать специальные значения аргументов: MPI_IN_PLACE, MPI_PROC_NULL, MPI_COMM_SELF, MPI_COMM_NULL и др.

Этапы выполнения работы

В файл `mpicoll/coll.c` необходимо добавить реализацию заданных алгоритмов. Имя функции, реализующей алгоритм, должно начинаться с префикса `<операция>_<алгоритм>`. Например:

- `allgather_bruck`
- `bcast_btree`
- `barrier_ring`

Прототип функции (тип возвращаемого значения и список аргументов) должен соответствовать описанию функции в стандарте MPI 3.0.

В файл `coll.h` следует добавить прототипы реализованных функций – их объявления.

Описание алгоритмов можно найти в работе [1], а примеры реализации разрешается смотреть в исходных текстах библиотек MPICH2 и Open MPI.

Литература

1. Thakur R., Rabenseifner R. and William Gropp. Optimization of Collective Communication Operations in MPICH // Int. Journal of High Performance Computing Applications. – 2005. – (19)1:49-66.

Алгоритмы реализации коллективных обменов информацией

Вариант	Коммуникационная функций	Алгоритм
1	MPI_Barrier	Dissemination algorithm
2	MPI_Barrier	Double ring
3	MPI_Barrier	Recursive doubling
4	MPI_Bcast	Binomial tree
5	MPI_Bcast	Scatter (Binomial tree) + Allgather (Ring)
6	MPI_Bcast	Van De Geijn
7	MPI_Gather	Binomial tree
8	MPI_Allgather	Ring
9	MPI_Allgather	Recursive doubling
10	MPI_Allgather	Bruck
11	MPI_Reduce (только для операции MPI_SUM)	Binomial tree