Designing and Implementation of Parallel Numerical Computing Library for Multi-node Environment of the Earth Simulator

Group Representative
Ken'ichi Itakura The Earth Simulator Center, Research Scientist

Author
Ken'ichi Itakura The Earth Simulator Center, Research Scientist

Our project makes the parallel numerical computing library for multi-node environment of the Earth Simulator. We evaluated the performance of the data transfer between nodes. The data transfer library is not tuned for the Earth Simulator hardware. Therefore we develop new data transfer library for high performance. In FY2003, some members of Japan Atomic Energy Research Institute will join to our project to develop the numerical computing library. The new message passing library and the numerical computing library based on MPI from JAERI will make much higher performance. We will evaluate this library and more tune the routines with selecting algorithms for the Earth Simulator architecture.

Keywords: numerical computing library, message passing library, non-blocking data transfer

1. A Goal of our project

The numerical computing library "ASL/ES" which makes easy to achieve high performance on many kinds of application program is installed in the Earth Simulator. This library is selected the most effective algorithm and tuned for the hardware architecture of the Earth Simulator. Therefore, it's much useful for tuning in many user application programs.

The ASL/ES library is not only corresponded to single Application Processor (AP) but also to multi-processors with the microtask processing in a node. The user can executes a large-scale problem in a little time by this parallel processing.

However, the application programs running on the Earth Simulator are needed massively processing power, and they cannot be suitable to the ordinary super computer nor a node on the Earth Simulator. To use for these large-scale application program, the multi-nodes numerical computing library is need.

We examined and evaluated some the performance of the data transfer between nodes. From the results, the data transfer library is not tuned for the Earth Simulator hardware nor the standard application programs. The user can use MPI library. MPI is the standard for several parallel systems including PC clusters, Grid Computing and the high performance parallel computers. There are some protocol overhead in the implemented of the data transfer library on the Earth Simulator and the performance to transfer of middle size of data are degraded. Furthermore, because of the different of implementation policy, some nonblocking API's are inefficient. We develop a special data transfer library for the Earth Simulator in use our numerical computing library. The data transfer hardware between nodes of the Earth Simulator has a "stride data" transfer function. We'll take in new data transfer library.

Our project's points are below:

- We develop multi-nodes numerical computing library which use ASL/ES on each node.
- We develop the special data transfer library for our numerical computing library.

2. Time Schedule

Our project's time schedule is below:

- 1-3/2003 Developing the data transfer library
- 4-6/2003 Evaluating the data transfer library
- 5-12/2003 Developing multi-nodes numerical computing library. Evaluating by some sample codes.
- 1-3/2004 Evaluating our all systems on real application programs.

Now, we finished developing the data transfer library. The detail of our data transfer library is described in section 3 and 4.

In FY2003, we will evaluate the data transfer library to apply some application program directly. We will also develop the multi-nodes numerical computing library including the data transfer library, and evaluating on real application programs on the Earth Simulator. The detail is described in section 5.

3. The new data transfer library

In this section, we describe our developed new data transfer library. Our library issues some assembly instructions to directory control multi-nodes communication hardware (RNA Unit: Remote node Access Unit).

3.1. Functions of RNA

The RNA directly access to main memory unit for communication between nodes. This memory through put is the same of APs. In the most low software level, one AP issues a request instruction to transfer a message. After this instruction is reached to RNA, the AP can go on next instructions without finishing the data transfer. The status of data transfer will be stored in a memory area which is directed by the message transfer instruction. Therefore, the calculation on APs and the data transfer on RNA are running simultaneously.

3.2. User API

The MPI_PUT and MPI_GET are called one side communication functions in MPI2 standard and they accessed remote processes memory area which is called "Window". We develop some user API's that have the same arguments in MPI2 standard.

The new user API shows in below:

- rnampi_alloc_mem
- rnampi_win_create
- rnampi_win_fence
- rnampi_put
- and rnampi_get.

The rnampi_win_create function define the data transfer memory area. To use RNA functions, the ordinary address and the global address of the memory area are needed. The global address is used when RNA accessed to main memory and it is an alias of the ordinary address for APs. The mpi_win_create functions is specified only the ordinary address. However, we cannot get the global address from the ordinary address. Then, the data transfer memory area is limited to allocated in rnampi_alloc_mem functions, this allocate functions keep the relationship between the ordinary and global addresses.

The rnampi_win_fence functions synchronize to finish the data transfer functions (rnampi_get and rnampi_put) on all nodes. In the RNA primitive, the acknowledgment is received after finishing memory access in the remote node and the status is written in the local node. Therefore, to taking a guarantee in rnampi_win_fence functions is collecting all statuses in each node and taking a global barrier. The Earth Simulator has 128 global barrier counters and a few ones are allocated for a MPI processes group. The original MPI2 specify that a new communicator is created on mpi_win_create. Our new function rnampi_win_create do not create and the original communicator is used in the window operations. From this implementation, the operations of the window with MPI_COMM_WORLD communicator are used global barrier counter and take much shorter time than using software barrier operation.

Table 1 Performance Evaluation of overlapping cpu load and data tranfer

PROG. UNIT	mpi_put ELAPSE [sec]	rnampi_put ELAPSE [sec]
work 1	0.201	0.201
work 2	0.170	0.170
work 3	0.371	0.202

3.3. Stride Data Transfer and List Data Transfer

Rnampi_put and rnampi_get are only accessed an continues memory area. There are two more RNA primitive functions that access more complex memory access pattern. One is "stride data" transfer and the other is "list data" transfer.

These functions does not access continues memory patterns and they are specified by data derivation type or special API's which have more arguments. We select to create special API for the present because the implementation is easy and do not harm the performance.

These functions have not been completely evaluated, yet. One of the important is the memory access performance in stride and list data pattern. We will show some guidelines to use these functions from performance evaluations.

4. Results on FY2002

We checked our new data transfer library for many cases and checked the results.

Then, for performance evaluation, we tested that the calculation on APs and the data transfer on RNA are running simultaneously.

There are three workload subroutines in performance evaluation program. PRONG.UNIT work1 is cpu load only, PROG.UNIT work2 is data transfer only, and PRONG.UNIT work3 is overlapping cpu load and data transfer. The case of using mpi_put, the elapse time of work3 is summed work1 and work2. This means no overlap processing. The case of using rnampi_put, the elapse time of work3 is maximums work1 and work2. This means that the cpu load and data transfer are overlapping.

5. Future Plan

In FY2003, some members of Japan Atomic Energy Research Institute will join to our project to develop the numerical computing library. They have developed a numerical computing library for vector processing, shared memory in a node and distributed memory between nodes architecture. This library includes Simultaneous Linear Equations

(Iterative Method), Eigenvalues of a Hermitian Matrix and Fast Fourier Transforms Routines. This Simultaneous Linear Equations (Iterative Method) routines were applied for the other project on the Earth Simulator titled "Fluid simulations for for spallation type mercury target". Eigenvalues of a Hermitian Matrix routines are planed to apply for the area of Quantitative Quantum Calculations project on FY2003. The high performance of Fast Fourier Transforms Routines is one of the most demand functions and it is very important to provide for all users of the Earth Simulator.

To cooperate new message passing library based on MPI and the numerical computing library from JAERI will makes much more higher performance numerical computing library. We evaluated this library and more tuning the routines with selecting algorithms for the Earth Simulator Architecture.

地球シミュレータのマルチノード用並列計算ライブラリの構築

利用責任者
板倉　憲一　　地球シミュレータセンター　研究員
著者
板倉　憲一　　地球シミュレータセンター　研究員

地球シミュレータには、様々なアプリケーションプログラムにおいて計算機の性能を容易に利用可能とするために数学ライブラリがインストールされている。この数学ライブラリ ASL/ES は高速なハードウェアに適合し、最適なアルゴリズムを採用しており、地球シミュレータでチューニングに役立っている。しかし、地球シミュレータで行われている処理は、従来のスパコンでは処理できない規模のプロセッサパワーやデータ量の問題であり、マルチノードにおける数学ライブラリの整備が必要である。ライブラリの整備をするにあたり、分散環境下でのノード間通信について様々な調査結果を鑑みると、ハードウェアの性能を十分に活かしきれていないことが分った。汎用の通信ライブラリとしては、非常に広範囲な使い方を規範した MPI に準拠するために、単純な通信に対してプロトコルオーバヘッドが発生し、中規模程度の通信において性能改善の余地がある。今年度は、地球シミュレータのノード間通信ハードウェアの機能に関して、検討および予備的な実験を行った。来年度以降、通信ライブラリの性能評価と並列計算ライブラリの開発を行う。並列計算ライブラリの開発は、日本原子力研究所との共同研究とし、地球シミュレータのユーザーに有用であるものにする。

キーワード：数値計算ライブラリ、メッセージパッシングライブラリ、ノンブロッキング通信