Performance tuning of the JMA-NHM for the K supercomputer

Tsutao OIZUMI[1,2], Thoru KURODA[1,2], Kazuo SAITO[2,1], Le DUC[1,2], Junshi Ito[2],Syugo HAYASHI[2]

[1] Japan Agency for Marine-Earth Science and Technology, [2]Meteorological Research Institute

E-mail: oizumi@jamstec.go.jp

1. Introduction

The K supercomputer (owned by RIKEN, hereafter called “K computer”) is the most powerful supercomputer in Japan. Half of the computational resources is allocated to the "HPCI Strategic Program for Innovative Research (SPIRE)". One of the main research fields of the SPIRE is “Advanced Prediction for Natural Disaster Prevention and Reduction (Field 3)”. The Japan Meteorological Agency Non-Hydrostatic Model (JMA-NHM, hereafter refer as to NHM) is one of the main application programs in the Field 3. The NHM has been developed under a vector-type computing system, and the K computer is a scalar-type computing system. Therefore, performance tuning of NHM for the K computer is necessary for efficient operation. We and Fujitsu Co. Ltd, which is a vendor of the K computer, improved the time integration part of NHM's codes, MPI communication, memory allocation, and the file I/O system.

2. Tuning method

We have conducted tuning of the time integration in the source codes of NHM since 2011. In the tuning process, we first evaluated the calculation cost and status of thread parallelization in each loop, and obtained information on SIMD from compile process. Then we listed a top of 160 heavy loops, which occupy about 77% of the total computational cost. The following five techniques are applied: (1) To change partitioning method of parallelization threads: block partitioning were altered to cyclic partitioning. (2) Loop partitioning and prefetch: to apply cyclic partitioning and prefetch to a L1/L2 cash high demand miss rate loop for increasing memory throughput. (3) To merge several DO loops for sharing array: to reduce number of reading array element and increase performance. (4) To reduce reference frequency of list array. (5) To reduce IF sentences and to facilitate SIMD.

3. Improvement of MPI communication and file I/O system

The K computer has 88,128 nodes (Computational node: 82,944 nodes, I/O node: 5184 nodes). We faced three problems in the MPI communication and file I/O system of NHM in the K computer specifications. The main three problems and solutions are as follows: (1) Buffer error (MRQ Overflow): MRQ occurs in point-to-point MPI communication in case of using more than 50,000 nodes. The solution is to change point-to-point MPI communication to group communication. (2) The disc capacity and memory of each node is not enough for the high-resolution experiment output. To reduce output files size, we developed a parallel output system, and prepared a tool for unifying the parallel output files. (3) MPI parallelization was also applied to the preprocess tool (Figure 1) to prepare the initial condition.

4. Result

We modified 144 loops. The computational cost of each loop in the time integration loops are less than 1%. Figure 2 is a performance comparison between the original NHM and the tuned NHM for a forecast case with a 1600 x1100 km domain. The elapse of integration process is 15% reduced, and the peak performance is 5.7 % (23 % increasing). Table 1 shows that the weak scalability of the tuned NHM is more than 96 %. We also validated the NHM output using 800 nodes and 82,944 nodes of the K computer under the same experimental condition. Both outputs from each NHM are completely the same. The performance in practical experiment is shown in Table 2. In the parallel preprocessing, the total execution time for the 250 m resolution experiment is reduced to 1/20 compared with the serial processing.

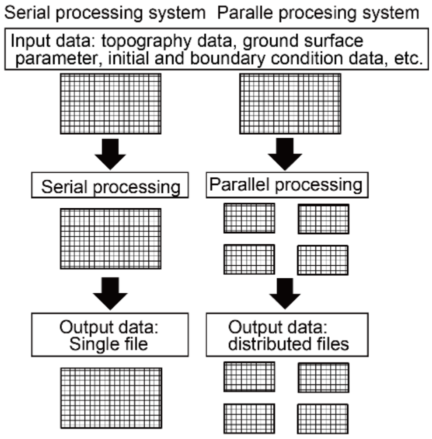
Serial processing system
Paralle procesing system
Input data: topography data, ground surface parameter, initial and boundary condition data, etc.
Serial processing
Parallel processing
Output data: Single file
Output data: distributed files

Fig. 1. Parallel preprocess system.

Table 1 Weak scalability of tuned NHM.

Number of CPUs	parallel performance
72	99%
288	
288	99%
1152	
1152	96%
4608	
4608	97%
18432	

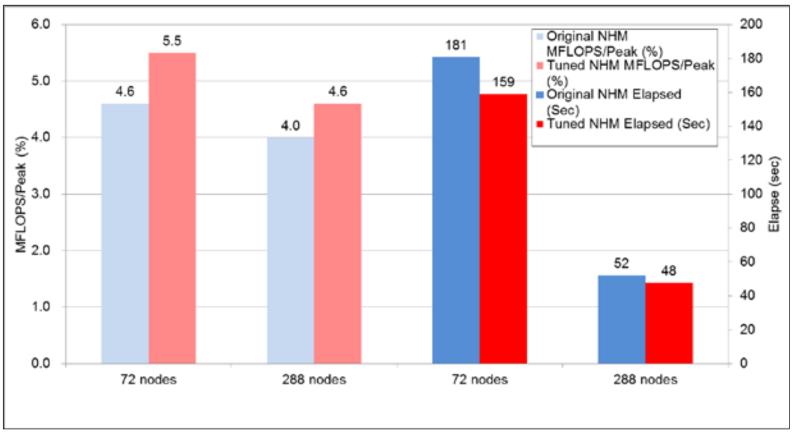

Fig. 2. The elapse and the peak performance of the time integration part.

Table 2 The elapse time and the peak performance in the practical experiments.

	Number of nodes	Parallel preprocessing	Tuned NHM	Peak performance(%)
2 km	72	0:03:23	0:32:11	4.70%
500 m	1152	0:25:48	4:12:55	2.74%
250 m	4608	0:59:43	18:57:34	2.49%

Acknowledgments: This research used the K computer (HPCI SPIRE FIELD3, ID. hp140220)