An Execution-Time Estimation Model for Heterogeneous Clusters

Graduate Advisor：Shuichi ICHIKAWA 003101 Yoshinori KISHIMOTO

1 Introduction

Many applications for parallel computers or homogeneous clusters suffer from load imbalance on heterogeneous clusters. It is simple to invoke multiple processes on fast processing elements (PEs) to alleviate load imbalance. This technique (**multiprocessing**) is widely applicable to many other applications.

It is not always preferable to use all PEs. To optimize the multiprocessing approach, it is necessary (1) to select optimal subset of PEs and (2) to determine the optimal number of processes on each PE. This problem is modeled as a combinatorial optimization problem to minimize the total execution time, where one must construct an objective function that estimates the total execution time from the given PE set and the given number of processes. In this study, execution-time estimation models are constructed from the measurement results of High Performance Linpack (HPL) [1] to estimate the actual optimal (or suboptimal) PE configuration.

2 Execution-Time Estimation Model

The estimation models are constructed from some small HPL trials. Since the orders of execution time are derived from the algorithm of HPL, constant factors are extracted from measurement results by the least-squares method. This kind of modeling technique is widely applicable to any other application.

In this study, we make the following assumptions to simplify our model: (1) Assume that the communication time is independent of the sender/receiver and (2) Apply the same M_i to PEs of the same specification. Such simplification may possibly lead to a slight discrepancy with reality, which must be examined empirically. The evaluation result will be found in Section 3.

Let G_i be the PEs of the same specification, P_i the number of processors on G_i, M_i the number of processes on PEs in G_i. The purpose of the model is to estimate execution time T_i from N, P, and M_i ($P = \sum P_i M_i$). The total execution time T estimated $T = \max_i(T_i)$.

The total execution time is estimated by the approximation formula (1) and (2), which are derived from the orders of computation and communication of HPL algorithm. In the following discussion, Equation (1) for a given set of P, M_i is called N-T model, and Equation (2) for a given set of M_i is called P-T model.

$$
\begin{aligned}
T_i(N)|_{P,M_i} &= k_0 N^3 + k_1 N^2 + k_2 N + k_3 && (1) \\
T_i(N,P)|_{M_i} &= k_4 P \cdot T_i(N)|_{P,M_i} + \\
&\quad k_5 \frac{1}{P} \cdot T_i(N)|_{P,M_i} + k_6 && (2)
\end{aligned}
$$

It is necessary to measure $T_i(N)$ of (at least) four different N to extract coefficients. If measurement set or interval of N is not enough, coefficients can not be extracted correctly (see Section 3). Since it is not sophisticated to manage many N-T models for each set of P and M_i, we tried to integrate N-T models for the same set of M_i into a P-T model. We have to measure (at least) three different P for each configuration to extract coefficients. When HPL is executed on a single PE_i, this case is distinct from the execution with multiple processors. Thus, the N-T model is used for $P = M_i$, while the P-T model is used for $P > M_i$. In this study, models are selectively used according to $P$and M_i as shown in Fig. 1.

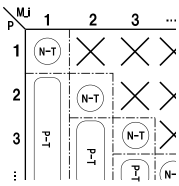

Figure 1: Binning

3 Model Evaluation

In this section, the estimation models are built and evaluated for a heterogeneous cluster, as shown in Table 1. In this study, measurements were made for every combination of N, P_i, and M_i. Since performance ratio G_2 to G_1 is 4 to 1, the range of M_1=1, ..., 6.

Table 1: HPL Execution Environment

PE	Athlon 1.33 GHz×1 (G_1), Intel Pentium-II 400 MHz×8 (G_2)
OS, Network	RedHat Linux7.0J (kernel 2.4.2), 100base-TX (Intel Pro100+)
Compiler	gcc 2.96, -DHPL_DETAILED_TIMING -fomit-frame-pointer -O3 -funroll-loops -W -Wall, MPICH–1.2.5, ATLAS 3.2.1

Table 2: Cluster Configuration Parameter

	Athlon		Pentium-II		Configurations
	P_1	M_1	P_2	M_2	
Parameter Extraction	1	1∼6	1∼8	1∼6	54
Performance Evaluation	0∼1	1∼6	0∼8	1	62

Table 3: Measurement set of N for N-T model

	Measurement set of N	total	Measurement
N9	400,600,800,1200,1600,2400,3200,4800,6400	486	6.4 [hour]
N5	400,800,1600,3200,6400	270	4.2 [hour]
NS	400,600,800,1200,1600	270	0.37 [hour]

Measurements were made for every combination of parameters, as shown in the "Parameter Extraction" in Table 2. In this study, I tried to reduce measurements for N. The estimation models from measurement set N9, N5, and NS are constructed and evaluated.

The models were constructed using the results of measurements from Table 3. Next, these models were applied to estimate the execution time of 62 possible configurations shown in the "Performance Evaluation" in Table 2 to find the optimal configurations for $N = 3200$, ..., 9600. Then, I measured the actual execution time for the same 62 possible configurations to determine the best configuration.

The errors of models from N9, N5, and NS against the measurement results are summarized in Table 4, where τ and $\hat{\tau}$ are the estimated execution time and the actual execution time of the estimated best configuration. $\hat{T}$ is the actual execution time of the actual best configuration. ϵ and e are the errors between τ and $\hat{\tau}$ against $\hat{T}$, respectively.

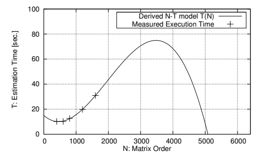

Figure 2: failed model

The error ϵ of N9 was less than 12.4%. The errors e of N9 and N5 were both less than 7.4%. It is not so far from the actual best configuration. The error ϵ of N5 was less than 15.0%. The measurement time N5 was less than N9. The error ϵ of NS was very big. For $N = 9600$, the estimation time τ was negative, because the extracted models were broken as shown in Fig. 2

These results shows that (1) 5 measurement sets of N seems enough, and (2) model construction fails if the measurement range of N is small.

Table 4: Errors of estimated best configuration

Size	$\epsilon = (\tau - \hat{T})/\hat{T}$			$e = (\hat{\tau} - \hat{T})/\hat{T}$		
N	N9	N5	NS	N9	N5	NS
3200	-0.018	0.019	-0.106	0.000	0.000	0.608
4800	-0.099	-0.080	-0.559	0.074	0.074	0.238
6400	-0.096	-0.095	-0.787	0.022	0.022	0.134
8000	-0.124	-0.146	-0.983	0.015	0.015	0.100
9600	-0.093	-0.139	-1.146	0.000	0.000	0.099

4 Conclusion

In this study, multiprocessing approach was examined to alleviate load imbalance in heterogeneous clusters. First, the estimation models were implemented from the measurement results of HPL. Then, these models were used to find the (sub-)optimal configuration. The error of derived models are sufficiently small, if they are constructed from enough measurements.

References

[1] A. Petitet et al. "HPL – A Portable Implementation of the High-Performance Linpack Benchmark for Distributed-Memory Computers," http://www.netlib.org/benchmark/hpl/.

[2] Y. Kishimoto, S. Ichikawa "The Execution Time Estimation Model for Heterogeneous Clusters and Its Evaluation," *IPSJ SIG Notes 2003–HPC–95*, pp. 161–166 (2003). (In Japanese).