Performance analysis of applications in parallel systems

Luisa Massari
Dipartimento di Informatica e Sistemistica
Università di Pavia
via Abbiategrasso, 209
I-27100 PAVIA, Italy
tel: +39 382 391369
e-mail: massari@ipvpel.unipv.it

1 Introduction

The evaluation and prediction of the performance of a system are strictly related to the specification of the workload submitted to the system itself. Indeed, performance indices depends on the resources physical characteristics, which are easily identifiable, and on the load elaborated by the system. Due to the difficulty in reproducing the execution of a real workload, the construction of a workload model is required [1]. The first step of this process is the definition of a set of parameters able to characterize the workload, i.e., to capture and reproduce the behavior of the system in terms of the resource consumptions required during its execution.

Based of this description, a workload model is then constructed, identifying workload components which are representative of the real workload.

In traditional sequential systems, parameters such as number of I/O operations and CPU required are sufficient to give insights into the real behavior of the application. In parallel systems, the identification of such parameters is a particularly important and complex process, due to the variety of hardware and software components that influence the multiprocessor performance. Indeed, in such environments, the static as well as the dynamic behavior of the application has to be reproduced.

In this paper, a sensitivity study on the parameters able to characterize the workload processed by multiprocessor systems is presented. In the next section, the most representative "static" parameters, derived from the analysis of the application, and "dynamic" parameters, obtained by monitoring the execution of the applications on a particular system, are presented.

In order to study the robustness of these parameters, a model of a parallel system, which allows the definition and simulation of a workload, has been developed. In this way, it is possible to correlate performance indices, such as response time, processor utilization and time spent waiting for processor allocation, with the parameters selected for the characterization of the application.

In Section 3, this model is proposed and some of the obtained results are analyzed.

2 Workload Parameters

In parallel systems the analysis of the algorithms execution requires the development of new methodologies and the definition of parameters able to capture the dynamic behavior of the algorithm itself.

These new metrics can be logically subdivided in "static" and "dynamic", based on the method through which they are obtained. Static metrics are derived analyzing the precedence graph, and the execution of the application on a real system is not required; they reflect the inherent parallelism of the application, and are architecture-independent. These parameters give a preliminary insigth into the algorithms characteristics; for example, the number of nodes in the precedence graph gives a knowledge of the algoritms granularity.

Dynamic indices, instead, require the execution of the application on a real system, and reflect the parallelism and the concurrency related to the particular architecture. These metrics can be described as a "curve" or as a single value.

Speedup, efficiency (see [2]) and efficacy (see [3]) are some of the dynamic "single value" indices which can be used for the characterization of a parallel application. Indices such as "execution profiles" and "execution signatures" will be considered for describing the workload.

An execution profile, which represents, as a function of time, the number of processors simultaneously busy, can be obtained by monitoring, during the execution of the application, the events related to processor activities. Fig. 1 shows an example of execution profile.

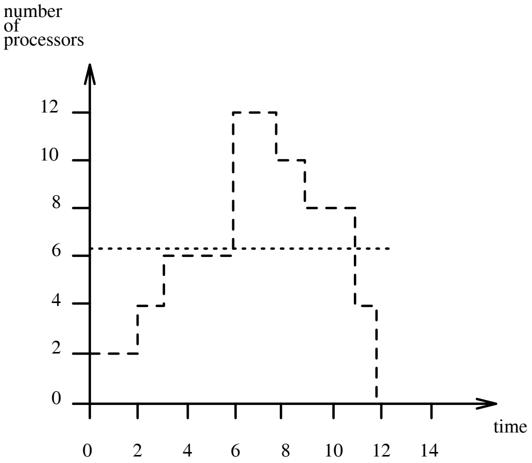

Figure 1: Execution profile.

This is a complex and expensive process because it requires the availability of software monitors, which may introduce overhead and distortions in the effective measurements. For this reason, it is sometimes more convenient to "approximate" the execution times and the number of processors required by the application by using average values, that is, mean execution time and average number of "busy" processors.

The dynamic behavior of an application can also be reproduced by means of its "execution signature" [4], which represents the execution time as a function of the number of processors allocated to the application. Note that the communication and computation components can be recognized within the execution time. In Fig. 2, the behavior of these two components, as a function of the number of allocated processors, is depicted.

3 System Models

Based on the previous described parameters, two models of parallel architectures are presented, both based on queueing network paradigms. The multiprocessor system is made up of a set of processors able to perform parallel activities, and it is connected to the external world through a front-end. Both the systems are represented by means of open models. Hence, they are completely defined by specifying the number of processors of the system, the arrival frequency of the applications, together with the parameters selected for their description.

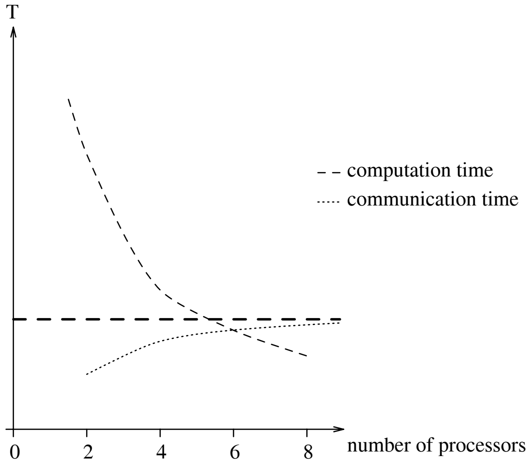

Figure 2: Execution signature of an application.

The system workload may also be represented by "classes" of applications having different characteristics. Each of them is constituted by a set of sequential units of computation.
The models differ in the logical organization of the processors, and in their associated activities.
The first model, depicted in Fig. 3, has been developed with the aim of studying the benefits of a characterization through "execution profiles", and the approximation eventually introduced by using "average values" for the number of processors and execution times.
In this case, the application is represented by a precedence graph, as shown in Fig. 4, where each task has a mean elaboration time associated to it.
Following the distribution of the number of processors required by the application, a dynamic allocation/deallocation of the processors has been adopted, in order to reproduce the detailed behavior of the profile of the application.

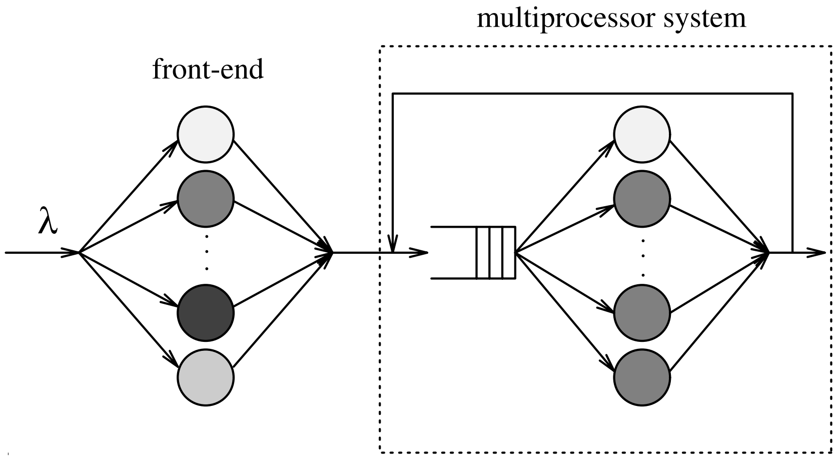

Figure 3: Model based on "execution profile".

Figs. 5.a and 5.b show a few of the experimental results obtained using the two different set of parameters. In particular, the processor utilizations and the waiting times are plotted as a function of the arrival rate of the applications which are considered to be statistically equal, that is, belonging to a single class. From the obtained performance indices, relative errors have been calculated, obtaining for the utilization, as an example, values ranging from 1% to 8%.

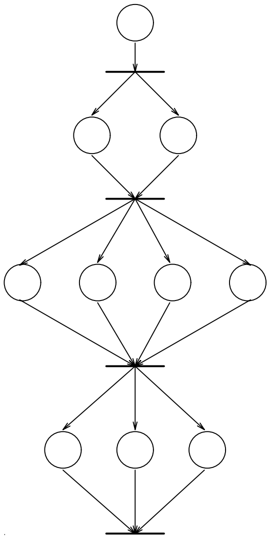

Figure 4: Application modelled through "execution profile".

A different characterization of a parallel application has been obtained by means of the "execution signature". In order to use this parameter, the processors have been logically subdivided into the model into communication and computation processors, as shown in Fig. 6.

Due to the complexity in managing different profiles, in this case the application has been characterized by means of the "average values" of computation and communication times (see Fig. 7), which vary depending on the number of processors allocated to the application, according to the specified signature.

An example of the results obtained is shown in Fig. 8, which plots processor utilizations due to communication and computation activities. The intersection point of the two curves corresponds to the "processor working set" of the application, which leads to an optimal system operating point [3]. When a lower or higher number of processors than the processor working set is allocated, the bottleneck of the system may become the computation or communication activities, respectively.

Some future evolutions of this study have been identified. In particular, the implementation of policies for scheduling of applications waiting for processor allocation and belonging to different classes have to be investigated. Indeed, the study of the relationships between the characteristics of the application and the system state, represented for example by the number of busy processors and their utilization, can optimize the utilization of the system itself.

References

[1] D. Ferrari, G. Serazzi, and A. Zeigner. *Measurement and Tuning of Computer Systems.* Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1983.

[2] D.L. Eager, J. Zahorjan, and E.D. Lazowska. Speedup Versus Efficiency in Parallel Systems. *IEEE Transactions on Computers*, 38(3):408–423, March 1989.

[3] D. Ghosal, G. Serazzi, and S.K. Tripathi. The Processor Working Set and its Use in Scheduling Multiprocessor Systems. *IEEE Transactions on Software Engineering*, 17(5):443–453, May 1991.

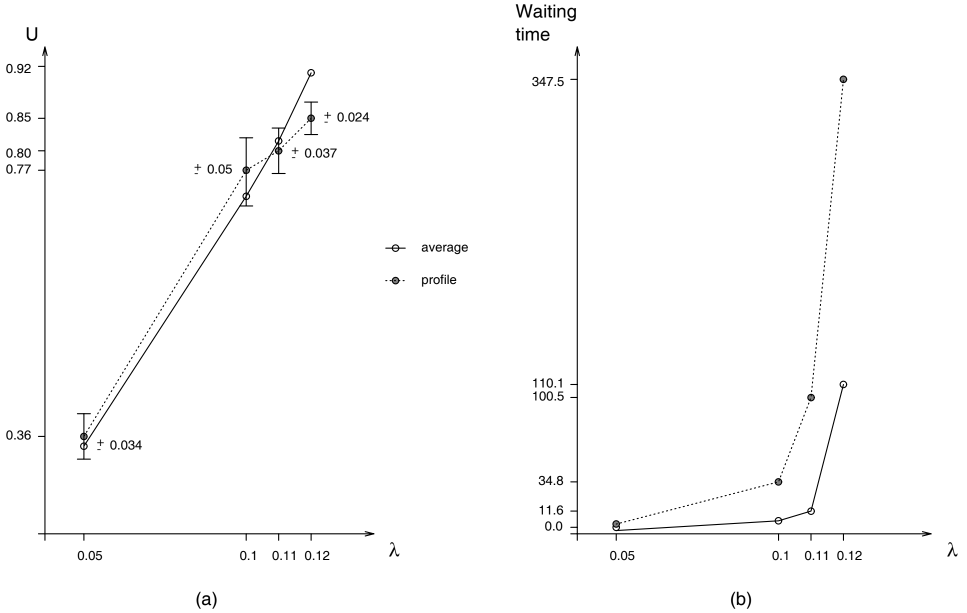

Figure 5: Comparison of performance indices obtained through "profiles" and "average values".

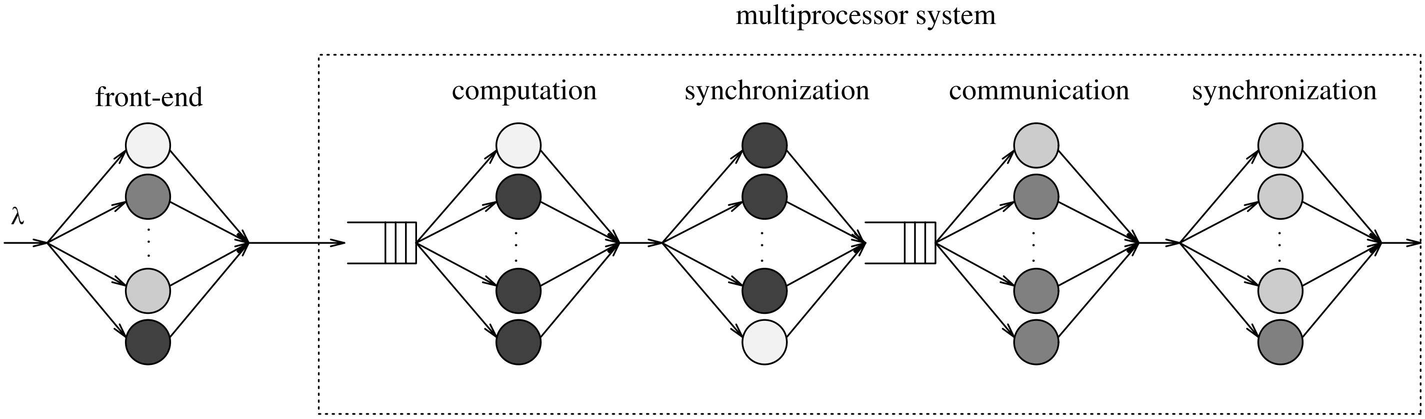

Figure 6: Model based on "execution signature".

[4] B.M. Carlson, T.D. Wagner, L.W. Dowdy, and P.H. Worley. Speedup properties of phases in the execution profile of distributed parallel programs. In R. Pooley and J. Hillston, editors, *Proc. 6th Int. Conf. on Modelling Techniques and Tools for Computer Performance Evaluation*, pages 83–95, Chippenham, Wiltshire, 1992. Antony Rowe Ltd.

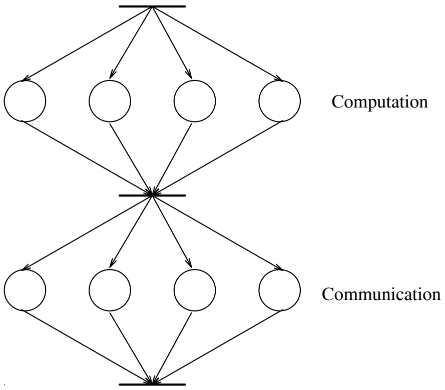

Figure 7: Application modelled through "execution signature".

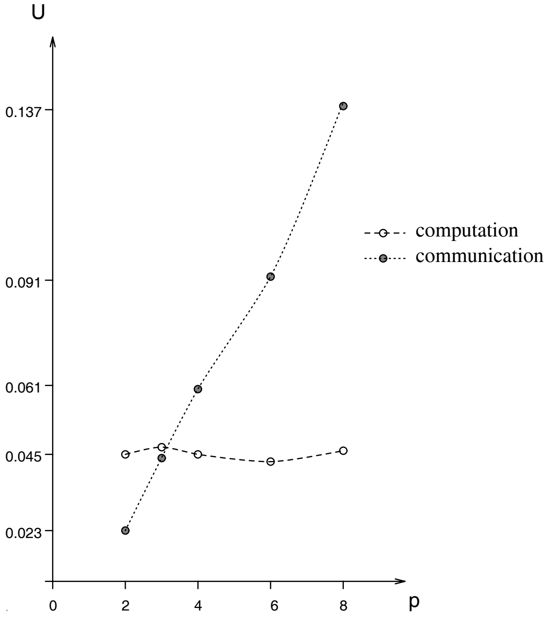

Figure 8: Processors utilization due to communication and computation activities.