A Hierarchical Approach to Workload Characterization for Parallel Systems*

M. Calzarossa¹, G. Haring², G. Kotsis², A. Merlo¹, D. Tessera¹

¹ Dipartimento di Informatica e Sistemistica, Università di Pavia
Via Abbiategrasso 209, I-27100 Pavia, Italy
e-mail: {mcc,merlo,tessera}@gilda.unipv.it

² Institut für Angewandte Informatik und Informationssysteme, Universität Wien
Lenaugasse 2/8, A-1080 Vienna, Austria
e-mail: {haring,gabi}@ani.univie.ac.at

Abstract. Performance evaluation studies are to be an integral part of the design and tuning of parallel applications. We propose a hierarchical approach to the systematic characterization of the workload of a parallel system, to be kept as modular and flexible as possible. The methodology is based on three different, but related, layers: the application, the algorithm, and the routine layer. For each of these layers different characteristics representing functional, sequential, parallel, and quantitative descriptions have been identified. These characteristics are specified in a system independent way to clearly separate between the workload description and the architecture description. Taking also architectural and mapping features into consideration, the hierarchical workload characterization can be applied to any type of performance studies.

1 Introduction

The main reason to use parallel systems is to get more and better performance, i.e. either to be able to solve larger problems or to solve given problems in a shorter time. So, in fact, performance is the driving force to develop new applications for parallel systems. The performance of such systems depends on many more aspects than that of a uniprocessor system, due to the more complex behavior of its hardware and software components. Therefore, from the performance point of view, there are more challenges but also deeper pitfalls [CS93].

Performance evaluation studies are to be an integral part of the design and tuning of applications, i.e. the workload, to reduce the development and performance debugging costs [CMM+94]. More or less for all these studies, but especially for performance predictions of applications under development, a monolithic approach is not appropriate. Due to the fact that workload, architecture, and mapping are the features with the largest impact on the performance of parallel systems, various performance prediction frameworks have been developed, where these aspects can be specified and modified in structured and

* This work was partially supported by the Austrian-Italian Cooperation 1994-96 Project Number 32, and by the 40% MURST Project.
flexible ways, as independent as possible from each other. Examples are the PAPS tool set [WKH93, WH94], the PRM [Fer90, Fer92] and the Mitchele-Thiel [Her92, MT93] approaches. But even these approaches use a rather "flat" description of the workload, reducing the understandability by increasing the representation complexity.

In this paper we propose a hierarchical approach to the systematic characterization of the workload of parallel systems able to manage the complexity of such a workload following the same principles as used in software engineering. Hierarchical descriptions of workloads have already proven their usability for representing workloads of interactive [Har86] and distributed [CHS88, RVH95] systems. Our hierarchical methodology is general enough to be easily adapted according to the objectives of the performance study. These objectives reach from getting "performance figures" for an existing application (or algorithm), which is composed of smaller components with known behavior, to predicting the performance of an application under development before implementing and executing it. In all these cases, a hierarchical approach is natural and helps to manage the complexity of the workload characterization and to drive the performance studies.

The methodology we present here can be applied both to functional and data parallel programming paradigms. From the architecture point of view, we can deal with distributed and shared memory, as well as virtual shared memory systems. Our characterization approach, which represents a logical view of the workload, is system independent, in the sense that the architectural characteristics come in when the final performance model is built. At that stage, either the communication or the memory access behavior will be modeled using explicit timings or hidden delays, according to the load and data distributions.

The paper is organized as follows. In Section 2 we introduce the underlying methodology of the proposed hierarchical approach and we define each layer in terms of its components. The characterization of each layer is discussed in Section 3. Section 4 concludes the paper with an outlook on future activities.

## 2 Methodology

Workload characterization studies are widely applied in the software design process and when a synthetic description of existing programs is required. Basic principles of software engineering technology, like abstraction and hierarchy, are taken to manage the complexity of the software engineering process or to better understand existing applications. In this section, the basic features of our hierarchical workload modeling methodology are pointed out. This approach corresponds to the user's or developer's mental model (view) of the software system.

Figure 1 (a) shows the mental view corresponding to the user's natural perception of a software system. Three possible levels of granularity are identified. Solving a specific "problem" requires the utilization of a few "methods" and