



A methodology towards automatic performance analysis of parallel applications [☆]

Maria Calzarossa ^{*}, Luisa Massari, Daniele Tessera

Dipartimento di Informatica e Sistemistica, Università di Pavia, via Ferrata 1, I-27100 Pavia, Italy

Received 10 October 2001; received in revised form 15 December 2002; accepted 19 August 2003

Abstract

Tuning and debugging the performance of parallel applications is an iterative process consisting of several steps dealing with identification and localization of inefficiencies, repair, and verification of the achieved performance. In this paper, we address the analysis of the performance of parallel applications from a methodological viewpoint with the aim of identifying and localizing inefficiencies. Our methodology is based on performance metrics and criteria that highlight the properties of the applications and the load imbalance and dissimilarities in the behavior of the processors. A few case studies illustrate the application of the methodology.

© 2003 Elsevier B.V. All rights reserved.

Keywords: Parallel application; Performance analysis; Performance metrics; Load imbalance

1. Introduction

The performance achieved by a parallel application is the result of complex interactions between the hardware and software resources of the system where the application is being executed. The characteristics of the application, e.g., algorithmic structure, input parameters, problem size, influence these interactions by determining how the application exploits the available resources and the allocated processors.

[☆]This work has been supported by the Italian Ministry of Education, Universities and Research (MIUR) under the FIRB and Cofin Programmes and by the University of Pavia under the FAR Programme.

^{*}Corresponding author. Tel.: +39-0382-505-350; fax: +39-0382-505-373.

E-mail addresses: mcc@unipv.it (M. Calzarossa), massari@unipv.it (L. Massari), tessera@unipv.it (D. Tessera).

In this framework, tuning and debugging the performance of parallel applications become challenging issues [16].

A typical approach to address these issues is experimental, that is, based on instrumenting the application, monitoring its execution and analyzing its performance either on the fly or post-mortem. Many tools have been developed for this purpose (see e.g., [1,2,17,18]). These tools analyze the measurements collected at run-time and provide statistics and diagrams describing the performance of the application and of its activities, e.g., computation, communication, I/O. The major drawback of these tools is that they fail to assist users in mastering the complexity inherent in this analysis.

To overcome this drawback, various methodological approaches have been proposed and tools have been developed out of these approaches with the aim of identifying performance bottlenecks, that is, the code regions, e.g., routines, loops, of the applications critical from the performance viewpoint. The Poirot project [8] proposed a tool architecture to automatic diagnose parallel applications using a heuristic classification scheme. The Paradyn Parallel Performance tool [12] dynamically instruments the applications to automate bottleneck detection at run-time. The Paradyn Performance Consultant starts a hierarchical search of the bottlenecks and refines this search by using stack sampling [14] and by pruning the search space considering the behavior of the application during previous runs [9]. The Kappa-Pi tool [3] deals with a post-mortem automatic performance analysis of message passing applications based on PVM. The analysis of processor utilizations leads to the identification of performance bottlenecks classified by means of a rule based knowledge system. Aksum [4] automatically performs multiple runs of a parallel application and detects performance bottlenecks by comparing the performance achieved varying the problem size and the number of allocated processors.

In this paper, we address the analysis of the performance of parallel applications from a methodological viewpoint with the aim of identifying and localizing performance inefficiencies. We define new performance metrics and criteria that highlight the properties of the applications and the load imbalance and dissimilarities in the behavior of the allocated processors. These metrics rely on the measurements collected by monitoring at run-time the applications. The integration of this methodology into a performance tool will help users in interpreting the performance achieved by their applications.

The paper is organized as follows. Section 2 presents the methodology and introduces metrics and criteria for the evaluation of the overall behavior of a parallel application. Section 3 focuses on the behavior of the processors allocated to the application. Section 4 presents an application of the methodology on a few case studies. Finally, Section 5 summarizes the methodology and discusses its integration into a performance analysis tool.

2. Characterization of performance properties

Tuning and debugging the performance of a parallel application can be seen as an iterative process consisting of several steps, dealing with the identification and local-

متن کامل مقاله

دریافت فوری ←

ISIArticles

مرجع مقالات تخصصی ایران

- ✓ امکان دانلود نسخه تمام متن مقالات انگلیسی
- ✓ امکان دانلود نسخه ترجمه شده مقالات
- ✓ پذیرش سفارش ترجمه تخصصی
- ✓ امکان جستجو در آرشیو جامعی از صدها موضوع و هزاران مقاله
- ✓ امکان دانلود رایگان ۲ صفحه اول هر مقاله
- ✓ امکان پرداخت اینترنتی با کلیه کارت های عضو شتاب
- ✓ دانلود فوری مقاله پس از پرداخت آنلاین
- ✓ پشتیبانی کامل خرید با بهره مندی از سیستم هوشمند رهگیری سفارشات