Models for On-the-Fly Compensation of Measurement Overhead in Parallel Performance Profiling

Allen D. Malony and Sameer S. Shende

Performance Research Laboratory
Department of Computer and Information Science
University of Oregon, Eugene, OR, USA
{malony,sameer}@cs.uoregon.edu

Abstract. Performance profiling generates measurement overhead during parallel program execution. Measurement overhead, in turn, introduces intrusion in a program’s runtime performance behavior. Intrusion can be mitigated by controlling instrumentation degree, allowing a trade-off of accuracy for detail. Alternatively, the accuracy in profile results can be improved by reducing the intrusion error due to measurement overhead. Models for compensation of measurement overhead in parallel performance profiling are described. An approach based on rational reconstruction is used to understand properties of compensation solutions for different parallel scenarios. From this analysis, a general algorithm for on-the-fly overhead assessment and compensation is derived.

Keywords: Performance measurement and analysis, parallel computing, profiling, intrusion, overhead compensation.

1 Introduction

In parallel profiling, performance measurements are made during program execution. There is an overhead associated with performance measurement since extra code is being executed and hardware resources (processor, memory, network) consumed. When performance overhead affects the program execution, we speak of performance (measurement) intrusion. Performance intrusion, no matter how small, can result in performance perturbation [7] where the program’s measured performance behavior is “different” from its unmeasured performance. Whereas performance perturbation is difficult to assess, performance intrusion can be quantified by different metrics, the most important of which is dilation in program execution time. This type of intrusion is often reported as a percentage slowdown of total execution time, but the intrusion effects themselves will be distributed throughout the profile results.

Any performance profiling technique, be it based on statistical profiling methods (e.g., see [4, 14]) or measured profiling methods (e.g., see [2, 9]), will encounter measurement overhead and will also have limitations on what performance phenomena can and cannot be observed [7]. Until there is a systematic basis for
judging the validity of differing profiling techniques, it is more productive to focus on those challenges that a profiling method faces to improve the accuracy of its measurement. In this regard, we pose the question whether it is possible to compensate for measurement overhead in performance profiling. What we mean by this is to quantify measurement overhead and remove the overhead from profile calculations. (It is important to note we are not suggesting that by doing so we are “correcting” the effects of overhead on intrusion and perturbation.) Because performance overhead occurs in both measured and statistical profiling, overhead compensation is an important topic of study.

In our Euro-Par 2004 paper [8], we presented overhead compensation techniques that were implemented in the TAU performance system [9] and demonstrated with the NAS parallel benchmarks for both flat and callpath profile analysis. While our results showed improvement in NAS profiling accuracy, as measured by the error in total execution time compared to a non-instrumented run, the compensation models were deficient for parallel execution due to their inability to account for interprocess interactions and dependencies. The contribution of this paper is the modeling of performance overhead compensation in parallel profiling and the design of on-the-fly algorithms based on these models that might be implemented in practical profiling tools.

Section §2 briefly describes the basic models from [8] and how they fail. We discuss the issues that arise with overhead interdependency in parallel execution. In Section §3, we follow a strategy to model parallel overhead compensation for message-based parallel programs based on a rational reconstruction of compensation solutions for specific parallel case studies. From the rationally reconstructed models, a general on-the-fly algorithm for overhead analysis and compensation is derived. Conclusions and future work are given in Section §4.

2 Basic Models for Overhead Compensation

In our earlier work [8], we developed techniques for quantifying the overhead of performance profile measurements and correcting the profiling results to compensate for the measurement error introduced. This work was done for two types of profiles: flat profiles and profiles of routine calling paths. The techniques were implemented in the TAU profiling system [9] and demonstrated on the NAS parallel benchmarks. However, the models we developed were based on a local perspective of how measurement overhead impacted the program’s execution. Profiling measurements are, typically, performed for each program thread of execution. (Here we use the term “thread” in a general sense. Shared memory threads and distributed memory processes equally apply.) By a local perspective we mean one that only regards the overhead impact on the process (thread) where the profile measurement was made and overhead incurred.

Consider a message passing parallel program composed of multiple processes. Most profiling tools would produce a separate profile for each process, showing how time was spent in its measured events. Because the profile measurements are made locally to a process, it is reasonable, as a first step, to compensate