Integrated Performance Analysis of Computer Systems (IPACS)

Benchmarks for Distributed Computer Systems

April 2006

Michael Krietemeyer and Matthias Merz (Eds.)

λογος

Foreword

When people speak about the performance analysis of computer systems, they often refer to simple benchmark tests that cover specific hardware characteristics. The well-known LINPACK benchmark for example, analyzes the floating point rate of execution for solving a linear system of equations. LINPACK summarizes different performance aspects (memory bandwidth and floating point operations) of a computer system to a single value (R_{max}). This allows building up a ranking system of HPC systems in an easy way, as it is demonstrated by the TOP500 supercomputer list. But summarizing performance of multiple categories into one score is highly subjective and could often be insufficient for an in-depth performance analysis.

New approaches cover major performance aspects of a computer. IDC, on the one hand, tries building a balanced rating system based on three categories (processor, memory, and interconnect performance). The HPC Challenges project, on the other hand, recommends the usage of six further benchmarks in addition to the LINPACK benchmark.

The IPACS project tries to provide a more comprehensive approach. The objective is to develop methods for measuring system performance on High Performance Computers based on low-level benchmarks, compute kernels, open source- and commercial application benchmarks. Additionally, IPACS covers the development of methods for performance modelling and prediction of commercial codes. In order to ensure an easy usability and a just-in-time analysis of benchmark results, the IPACS benchmarks are embedded into a benchmark environment consisting of a web based repository and a distributed benchmark-execution framework. By providing a new mechanism for an automatic installation and execution of benchmarks, IPACS also simplifies the performance analysis of computer system.

The IPACS client and execution framework as well as the IPACS benchmark suite are available for download at `www.ipacs-benchmark.org`. On this web site, you will also find our information retrieval component, that allows the search of benchmark results, the performance predication module as well as a kiviat diagram module, which allows an in-depth analysis of two or more high performance computers.

This book is structured into four parts: An introduction into the IPACS project, the results achieved by each project partner and a conclusion that identify future research topics. Finally, an appendix will provide a complete list of all IPACS publications and a comprehensive technical reference to the IPACS software environment.

We thankfully acknowledge the fruitful discussions with our colleagues from the IPACS project and others. In particular, our team says thank you to Hans-Werner Meuer, Djamshid Tavangarian, Alfred Geiger, and Erich Strohmaier for clarifying the requirements of the benchmark process. Our team also owes to thank all authors listed here for their work and the fact that they have kindly and actively supported our project and contributed to successfully finalizing this project. We appreciate and thank the executive director of the IT Center of the University of Mannheim, Hans-Günther Kruse, our colleges from the IT Center, the department of Information Systems III, and the Department of Mathematics and Computer Science of the University of Mannheim for their continuous support. Finally, we thank Klaus Thiele for improving the linguistic aspect of this publication.

The work on IPACS was supported by a grant from the German Federal Ministry of Education and Research (BMBF) in the program "High Performance and Grid Computing".

April 2006 Michael Krietemeyer and Matthias Merz

Contents

Part I

The IPACS Project

Chapter 1

IPACS Preface

With the continuing relevance of PC-clusters, SMP-clusters and the development of chip architectures into memory hierarchies and internal parallel processing, computer architectures have become more and more complex. Parallelism and the growing capacity of memory and storage have led to an increase of the growth of today's problems. But reliable and easy-to-use benchmarks, which moreover support the users in rating and evaluating parallel computer systems and help in the procurement of new computers, are still missing.

1.1 Context and Motivation

In this section, we will give some background information on performance analysis of computer systems before turning to related efforts in developments of integrated benchmarking environments.

With no doubt, the performance of computers, PCs and SMPs has been increased enormously in the last decades. This growth was driven not only by always raising clock rates but also by further developments of the chip architectures. Technologies like memory hierarchies, pipelines, or internal parallel processing and threading had significant impact. But these developments have also made computer architectures more and more complex and difficult to evaluate.

In addition to that, parallel clusters are widely used in the mean time and no longer restricted to a few HPC sites, enhancing the importance of parallelism and parallel performance for architectures as well as for programming techniques. This trend will become vitally in the near future since the increase of clock rates will reach its limit soon. With IBM's BlueGene/L, today already an architecture with a few ten-thousand processors is available. Even desktop machines will provide multiple cores, and with the new Cell architecture, a hybrid of cache and vector processors is on the market. Finally, with the implementation of computing and/or data grids, the meaning of distributed computing will be changed.

These developments immensely increase the complexity and variety of computer architectures. More and more, hardware features and components, like network connections

and protocols for parallel computing, have to be taken into account in order to estimate, rate and understand the performance of a system. This has also led to an increase of the complexity and difficulty of measuring the performance with benchmarks.

On the other hand, the increasing performance of computer systems, together with the growing capacity of memory and storage, has also improved the applicability of these architectures. The size of problems that can be treated or solved with clusters has grown enormously as well as precision and accuracy. Computer simulations are a valuable tool in more and more areas. With this development, information about performance also becomes important for more and more users, increasing the impact of benchmarking results. But the interpretation of such results or even the execution of benchmarks by a larger group of users is hampered by the complexity and the variety of architectures and benchmarks.

Among the vast number of benchmark programs, the TOP500 [1] list, based on the LINPACK benchmark [2], is the most publicly visible benchmark in the world. Its success comes first from the open availability of the source and supporting code together with the community validation of the results, secondly from the scalability of the LINPACK benchmark over all computer architectures within the last 25 years and thirdly from the interest of computer manufacturers in publishing the best LINPACK numbers for their systems as a competitive comparison. LINPACK numbers are for customers a prime corrective to the peek advertised performance (PAP) of computer systems, since a real benchmark program must be executed on an existing computer in order to obtain the performance numbers. Many other benchmark initiatives fail short on some of these three aspects:

- the benchmarks are only meaningful for certain architectures, hardware features or certain system sizes (e.g. NAS PB [3, 4] use only fixed problem sizes),
- to obtain and publish the benchmark one must be a member of an organization (and sometimes pay high membership fees) and follow certain procedures (e.g. SPEC [5]), TPC [6])
- or there is only an academic interest in the results of a benchmark (see the NETLIB link collection [2] with many dead projects).

However, the success of the LINPACK benchmark is also due to a limitation, it assesses the suitability of a computer system only by computing a solution to a dense and arbitrarily big system of linear equations. But many of today's applications incorporate new algorithms with different system stress patterns or algorithms based on new mathematical theories. Applications, which are not numerical and yet using parallel systems, are still not covered. With the growing relevance of computing grids, the situation will even become more aggravated in a few years. On the practical side, running the LINPACK with reasonable results by a benchmark professional is relatively easy, but for most new or young benchmarkers, it is very hard to tune and optimize the LINPACK software configuration to achieve good results.

The IPACS project [7] wants to improve this situation by augmenting LINPACK with a set of low-level and application benchmarks and in easing the execution of these benchmarks. In cooperation with colleagues at LBNL/NERSC, IPACS wants to define a new basis for benchmarks measuring system performance of distributed systems. These benchmarks should allow a realistic evaluation of performance, leading even to the prediction of performance, and are especially designed to be scalable and portable in order to facilitate their wide range and future use. The evaluation and selection or the development of augmenting benchmarks is part of other IPACS publications [8, 7] and some extend in Section 3.2. The usability of these benchmarks is further improved by providing a benchmark execution environment with online evaluation of benchmarking results to assist the inexperienced user.

1.2 The Goal of IPACS

As described above, the main goals and efforts of the IPACS project go in two directions: The development and compilation of benchmarks that are adequate for the complexity of today's and future systems, and the easing of the execution of these benchmarks and the interpretation of their results for a wide group of users.

For these goals, the benchmarks, which are developed, tested and propagated, comply to the following fundamental qualities:

- Scalable: The problem size is adaptable for a broad class of systems and should also jut out the next 10 years (Petaflop systems).
- Portable: The benchmarks should be public available. Moreover, installing and starting the benchmarks should be possible in a simple way without special efforts.
- Realistic: The benchmarks provide a framework for measuring system performance in a realistic, expressive, universally valid way.

This will guarantee the usability of these benchmarks for a wide variety of systems and the usefulness of their results. Furthermore, these results are used in this project in order to create the basis for forecasting the system performance for commercial applications in a simple way and particularly support specifically the industrial users stronger.

To cover all the relevant performance aspects of an architecture, the benchmarks are organized in the following groups:

- Low level benchmarks for the characterization of the computer or the Grid condition
- Open source benchmarks for industry near applications,
- Scalable benchmarks for commercial codes.

Additionally, "non-numeric applications", whose behavior, until now, was hardly examined, shall also be taken into account. And, finally, grid benchmarks are developed, including the environment for connecting and addressing grid services. Based on a component architecture, the IPACS benchmarks are also used as basis for these grid benchmarks. With this benchmark compilation, performance data can be collected on all levels of a given system for the purpose of allowing a deep insight into all main performance aspects. This insight is additionally used to develop performance modeling techniques for commercial application. Based on the benchmark results characterizing a given architecture, a model is presented that allows predicting the performance of complex applications. While the precision of such predictions is limited to $\approx 10\%$, the technique is easy to apply and should support the (industrial) user in interpreting and transferring benchmark data to his needs.

The second goal and essential element of the new benchmark suite is the simple usability of the benchmark by the user. In order to simplify the benchmarking process, a new mechanism is provided to enable an automatic installation and execution of benchmarks. A benchmark client has been developed that automatically allows the download of components from a repository and enables the upload of the results on the attached web server in interaction with the user. In addition to that, the user is supported and guided in executing the benchmark, i.e. defining the necessary options, building a job and submitting it to a queuing system. The results will be presented automatically via a web repository, allowing an easy comparison, sorting and analysis. This shall increase the acceptance of the benchmarks and form a data basis for performance measurements of distributed systems, that are valuable for experienced benchmarkers as well as for users of applications and architectures.

1.3 Related Work

Other benchmarking activities do not aim at such a highly automated benchmarking process cycle. There is one project, 'Repository in a Box', (RIB, [9]), which is a software package for creating web meta-data repositories which can contain metadata for benchmark suites for various application domains. This tool helps finding benchmarks or other software in a specific application domain that does not contain the benchmark code or benchmark results. The Performance Database Server ([10]) is a webserver which contains results of various benchmarks from Dhrystone to Linpack. The results from Linpack are mostly up to date, but other tables contain merely historical data. Data input seems to be sent via email to the maintainers. The goal of the Performance Evaluation Research Center ([11]) is a scientific understanding and improvement of the performance of HPC systems. Although they develop benchmarks and performance models for predictions (just as IPACS), facilitating the benchmarking and publishing process seems not to be intended. The HPC Challenge ([12]), together with the PaMaC project ([13]) also aim at a suitable benchmark suite which can comple-

ment the Linpack/TOP500 benchmark. The proposed benchmarks are primarily based on Linpack and its software infrastructure. The suite has not defined an I/O benchmark and there are no application benchmarks. The web-site contains an archive of benchmark results and provides a web-form to be filled out and submitted together with the benchmark result file. User validation is via email response with an activating URL. So the IPACS concept of integrating a benchmark code repository, a benchmark result repository and an automated process cycle contributes new ideas and experiences in the field of benchmarking.