WORKSHOP

ScalPerf'03: SCALABLE APPROACHES to HIGH PERFORMANCE and HIGH PRODUCTIVITY COMPUTING

April 13-17, 2003

Bertinoro International Center for Informatics
http://www.cs.unibo.it/bici/

ABSTRACTS

Non-linear non-simmetric FEM simulations: getting high performance computational kernels

Mauro Bianco
University of Padova

The simulation of complex physical phenomena is a challenging computational problem. The different competences needed to develop such an application are one of the reasons why it is difficult to obtain highly efficient programs to solve such problems. In this talk an experience of collaboration between structural engineers and computer engineers is shown for the development of high performance software for simulating concrete under high-temperature conditions. The physical problem is complex since concrete is a porous media in which many physical phenomena interact. The optimized computational kernel is, up to now, 240 times faster with respect to the original code even for a relatively small problem on a single processor, with numerical accuracy comparable with the accuracy parameters of the used architecture. This result has been obtained by taking into account the peculiarities of the physical problem, competences on algorithm design and code optimization to use computational resources efficiently.

Portability of memory maps across memory hierarchies

Gianfranco Bilardi
University of Padova

We investigate whether a given algorithm can be coded in a way efficiently portable across machines with different hierarchical memory systems. The machine models being used is the $a(x)$-HRAM (Hierarchical RAM), capturing the key property of a hierachical memory: the time to access a location x is a function $a(x)$ of the address, called the *access function.*

The *width decomposition* framework is proposed to provide a machine-independent characterization of temporal locality of a computation by a suitable set of *space reuse* parameters. Lower and upper bounds to the running time on an $a(x)$-HRAM can be derived in terms of such parameters (characterizing the computation) and of the access function $a(x)$ (characterizing the machine).

Using the width-decomposition framework, it is shown that, when the *schedule*, *i.e.*, the order by which operations are executed, is fixed, efficient portability is achievable. Specifically, we propose the *decomposition-tree* memory manager, which achieves time within a logarithmic factor of optimal on all HRAMs. We also propose an alternative *reoccurrence-width* memory manager, which achieves time within a constant factor of optimal for the class of *uniform* HRAMs, which includes all machines of practical interest.

We then introduce the *Pipelined Hierarchical Random Access Machine*, where memory can accept requests at a constant rate and satisfy each of the requests to the location x within $a(x)$ units of time. We consider a *pipelined decomposition-tree* memory management strategy and obtain a number of results on the time required to execute any sequence of N operations.

This work is in part joint with E. Peserico and in part with K. Ekanadham, P. Pattnaik.

Bulk Synchronous Parallel computing using idle times of workstation clusters

Olaf Bonorden
University of Paderborn

The Bulk Synchronous Parallel (BSP) model from L. Valiant directs the algorithm designer to create error-free communication efficient parallel algorithms. In contrast to the PRAM model the BSP cost model leads to coarse grained algorithms. These algorithms are very efficient on monolithic dedicated parallel machines with homogenous processors operating at the same speed.

The talk will present a system using virtual processors for running BSP programs on workstation clusters with dynamically changing processing times. E.g., this might be the idle times of a processors of a busy LAN. The virtual processors may migrate to adapt to changes in the load of the computers. One important topic is load balancing. Different scalable distributed algorithms for load balancing have been implemented. The evaluation of applications with different ratio of computation to communication shows the usability of the approach.

Status of Grid computing in Europe, and its readiness

Paolo Capiluppi
University of Bologna and INFN

The current status of some (out of many) Worldwide Grid Projects will be presented. Possible deployment of Grid to match the requirements of applications in the short and immediate term will be addressed. Some recent results of deployed architectures and measured performances of "running" grid project will also be discussed.

Computational requirements of large scale agent-based models

Filippo Castiglione
IAC-CNR

Agent-based modeling allows the description of very complex systems and it is, by now, widely employed in fields like biology, economy and finance. We describe two instances of this approach (a simulator of the Immune System response and a simulator of the Stock Market) that employ very similar technical solutions. The focus is on the respective computational features with special emphasis on the issues raised by the application of parallel processing techniques to this kind of simulations.

Scalability issues in material science and chemistry applications

Carlo Cavazzoni
CINECA

The presentation will focus on how Material Science and Chemistry applications have been parallelized in the last 10 years, and how today these applications are able to address capability through scalability on a limited number of processors (few hundreds). It will be shown that thanks to the availability of parallel computers (from cluster to supercomputers) the dimension of simulations has grown much more than the cpus, memory and disks speed. Then the challenge on how to sustain this grow rate in the next future will be discussed, starting from the fact that computer makers are now speaking of next generation computers with thousands and even million of processors. Finally it will be discussed if and how a grid-like approach is worth to this kind of applications.

The SB-PRAM project

Roman Dementiev
Max-Planck-Institut für Informatik

The SB-PRAM is a shared memory multiprocessor with almost uniform memory access. It uses multithreading in order to hide latency, a pipelined combining butterfly network in order to reduce hot spots, and address hashing in order to randomize network traffic and to reduce memory module congestion. A prototype of a 64 processor SB-PRAM has been completed. We present the architectural details of the SB-PRAM as well as performance measurements on this prototype.

Performance issues in algorithm engineering

Camil Demetrescu
University of Roma “La Sapienza”

Algorithm Engineering is concerned with the design, analysis, implementation, tuning, debugging and experimental evaluation of computer programs for solving algorithmic problems. It provides methodologies and tools for developing and engineering efficient algorithmic codes and aims at integrating and reinforcing traditional theoretical approaches for the design and analysis of algorithms and data structures.

This talk addresses performance issues in Algorithm Engineering, discussing cost models, implementation pitfalls, tuning, and experimental analysis of algorithmic codes.

Processor organizations for pipelined hierarchical memories

Kattamuri Ekanadham
IBM Research - T.J. Watson

We examine the problem of building systems in a scalable manner as technological advances force the designers to take a systematic view of communication costs as systems expand in space. An approach that has been extensively studied in the literature is to view the data storage spread out in space as a hierarchy, with increasing sizes and latencies as it stretches far out in space. The idea is to expose the locality in a computation to take advantage of automated movements of data in the hierarchy. Another avenue is to exploit the concurrency a memory system can offer and orchestrate the data movements accordingly. We briefly review pipelined hierachical memory designs that offer such advantage and proceed to look at how processors can use such memories. By looking at a computation basically as a collection of data movements in the processr-memory space, we derive lower bounds on the number of time steps any processor will have to spend for the computation. This is expressed in terms of the data dependences among the operations and spatial mapping of data, both of which are characteristics embedded in a given program specification. We present two alternative designs, one with and the other without speculation on certain aspects of the computation and derive upper bounds for the time steps they respectively take for the computation. The analysis exposes the dominant role played by the time taken to "locate" the data needed in a computation, in addition to the time taken for performing the arithmetic and logical operations.

Translating communication locality into locality of reference in parallel hierarchical machines

Carlo Fantozzi
University of Padova

Modern parallel machines exhibit a hierarchical structure in the communication network and in the memory subsystem; an efficient use of both relies on the exploitation of locality. Our work examines the interactions between communication locality (typical of the communication network) and locality of reference (typical of the memory subsystem). We show that a reduction in parallelism makes it possible to transform communication locality into temporal locality, thus incurring a slowdown merely proportional to the loss of parallelism. As a framework for our analysis, we introduce a sensible extension of the popular BSP model describing network hierarchy and memory hierarchy in an integrated fashion. Our technical result is a uniform scheme to simulate any computation for v processors on a v'-processor configuration with $v' < v$ and the same overall memory size. For a wide class of computations the simulation exhibits optimal $O(v/v')$ slowdown. As an important special case ($v' = 1$), our simulation is employed to obtain efficient hierarchy-conscious sequential algorithms from efficient fine-grained ones. Finally, some preliminary observations are presented concerning another important aspect of locality of reference, namely spatial locality.

The INFM parallel computing initiative

Chiara Marchetto
INFM

In the last ten years, the INFM has been promoting parallel computing among their researchers, with significant foundings and partnerships with other istitutions. It is also thanks to this initiative that the scientific applications using parallel computing have considerably grown, obtaining a scientific production of high value. In this talk an overview of the applications, the results, and the high performance resources used by the INFM scientific community will be given. It will be also discussed the requirements of new high performance parallel applications being developed for material sciences studies.

Impact of different memory hierarchies on a real-world code

Federico Massaioli
CASPUR

We present an analysis of memory accesses in a Computational Fluid Dynamic code, based on the Lattice Boltzmann Equation scheme. We find non-trivial relationships between different aspects of the hardware platform (architecture and implementation) and the tested memory access patterns. This applies to either serial or parallel versions.

Detailed measures on IBM POWER 3, IBM POWER 4, and on Intel IA-32 systems, and a few preliminary tests on Alpha EV7 architecture, are discussed. A predictive model of system memory hierarchies, accounting for the different behaviours observed, would be a relief for developers.

The work reported is joint with G. Amati of CASPUR.

How to use 170 trillion transistors

Jose Moreira
IBM Research - T.J. Watson

The IBM Blue Gene project aims to explore the limits of parallel processing through the development of a family of large scale computers with cellular architectures. With more than 100,000 processors, Blue Gene/L is the first machine in this family. Other architectures, with possibly even more processors, are being considered for future generations. In this talk, we will discuss why this is a good way to use the raw materials provided by continuing advances in microelectronics. We will describe the architecture of the Blue Gene/L machine, and how applications can exploit if effectively. We will also talk about architectural features being considered for future versions of Blue Gene.

A roadmap for the next generations of commercial multiprocessors

Pratap Pattnaik
IBM Research - T.J. Watson

The current evolution of the servers, to a large extent, is driven by the demands from an ever increasing data-centric society and by the rapid progress in the VLSI technology.

This talk will highlight the key considerations in today's multiprocessor design. Using IBM's Power family of processors as a case study, the talk will show the design trends in the modern commercial processors.

Temporal locality is not always portable

Enoch Peserico
U. Padova & MIT

Informally speaking, a computation exhibits "temporal locality" if data and operations can be organized in such a way that "most" memory accesses will involve only those "few" memory locations which in most memory systems today are much cheaper to access than the rest.

We show that, in general temporal locality cannot be leveraged in a "portable" way - that is, it is impossible to organize the sequence of the operations of some computations in single way that will "work well" on all memory systems. This result is somewhat surprising in the light of a growing literature of computations whose temporal locality is actually portable - it was in fact conjectured that this would be always the case!

Models for the design and implementation of efficient algorithms on parallel hierarchical machines

Andrea Pietracaprina
University of Padova

The effective exploitation of the massive computing power made available by current and future platforms requires the adoption of general computational models that support the design and analysis of efficient and portable algorithms, as well as the development of practical methodologies for profiling and optimizing applications on specific platforms. In this talk we will survey some of the most prominent directions that emerged in over a decade of active research on these issues. Specifically, we will present the Decomposable BSP model, a variant of Valiant's BSP, which explicitly accounts for the parallel and hierarchical nature of typical high-performance platforms, and argue that it exhibits higher effectiveness while maintaining generality. Moreover, we will discuss a methodology based on microbenchmarking and the use of hardware counters, to assess the relative impact of a number of architectural phenomena on the performance of applications.

The talk is based on joint works with: G. Bilardi, C. Fantozzi, G. Pucci and P. Sartore.

An experimental comparison of empirical and model-driven optimization

Keshav Pingali
Cornell University

Empirical optimization estimates the values of key optimization parameters by generating different program versions and running them on the actual hardware to determine which values give the best performance. In contrast, conventional compilers use architectural models to choose these parameters. It is widely believed that empirical optimization ismuch better than model-driven optimization but no actual studies exist to prove or disprove this belief.

In this talk, we describe some recent work in which we replaced the empirical optimization engine in ATLAS with a model-based optimization engine that used simple architectural models to estimate the same optimization parameters as ATLAS has, and compared the performance of the two systems on several common high-performance paltforms. Our experiments show that model-driven optimization is surprisingly effective - the performance gap between the two approaches is usually within 10%.

We conclude with a discuss of ongoing work.

This is joint work with David Padua's group at Urbana.

J&T: the node processor for the apeNEXT computer

Laura Sartori
University of Ferrara

Autonomic computing systems: the challenges of self-management

Walter F. Tichy
University of Karlsruhe

Advances in micro-electronics and software have led to distributed computings systems of unprecedented complexity. The labor costs for managing these systems outstrip the equipment costs by several factors, and the gap is widening. The vision of "Autonomic Computing Systems" says that computers should manage themselves rather than rely on human intervention. For this vision to become reality, autonomic systems need to configure themselves, adapt to changing environments, self-repair, self-protect, self-optimize, and learn from experience. While rudimentary forms of self-management exist, achieving a high-level of autonomic function is a major scientific and engineering challenge.

In this talk, I will explain the vision of autonomic computing and present examples of our research in this area. In particular, we are investigating how certain autonomic functions such as checkpoint/restart and reconfigurations can be added to programs with minimum effort, how to configure calls to services automatically, and how to let parallel applications tune their degree of parallelism.

Dedicated massively parallel computing for theoretical physics: results and perspectives of the APE project

Raffaele Tripiccione
University of Ferrara and INFN

The interactions between quarks are described in theoretical physics in the framework of Quantum Chromo-dynamics. This is a mathematical model of the physical system whose predictive power has so far never been found in disagreement with experimental data. Experimentally relevant data are extracted from the model by a variety of techniques.

A key role is played by numerical simulations of a discrete version of the theoretical model, known as Lattice Gauge Theory (LGT). The computing requirements of LGT are huge: at present, computer systems delivering sustained computing power of the order of hundreds of GFlops are used for simulation runs extending over periods of several weeks.

Faced with the need of very large computing resources, several research group in Europe, the US and Japan have developed and operated special purpose LGT computers, in the last 15 years. These systems have consistently provided computing performance for LGT applications comparable to state-of-the-art massively parallel systems, at a fraction of the price.

An LGT optimized computer architecture has a number of distinctive features, shaped by the underlying structure of the relevant algorithms:

- It is a massively parallel system, usually arranged as an n-dimensional mesh of processors. The relevant algorithms can be trivially mapped onto such an organization, with almost perfect load-balancing and a pattern of inter-processor communications that involves only first-neighbours in the processor mesh.
- The processor elements are optimized for floating point arithmetics.
- The structure of the code kernels is based on very long linear sequences of code, so deep pipelining is extremely effective for performance gains and branch control structures are not very important.
- Memory access can be performed in large blocks of consecutive addresses.
- Data transfer between processor can be limited to first neighbours, but latency should be kept low enough to allow transfer of short data packets.

The APE project has designed several generations of machines directly shaped by the lines discussed above. Early APE systems in the late eighties delivered about 1 Gflops peak performance, while the present generation (APEmille) touches the Tflops range of performance. apeNEXT is in an advanced development phase.

This talks describes in details apeNEXT, briefly recalling the physics motivations behind the project and focusing on the interaction between requirements and architecture.