Proceedings of
24th IEEE International Parallel and Distributed Processing Symposium

IPDPS 2010 Advance Program Abstracts

Abstracts for both contributed papers and all workshops have been compiled to allow authors to check accuracy and so that visitors to this website may preview the papers to be presented at the conference. Full proceedings of the conference will be published on a cdrom to be distributed to registrants at the conference.

Contents

IEEE International Parallel & Distributed Processing Symposium

IPDPS 2010

Session 1
Algorithms for Network Management

Distributed Advance Network Reservation with Delay Guarantees

Niloofar Fazlollahi and David Starobinski
Dept. of Electrical and Computer Engineering
Boston University
Boston, USA
{nfazl,staro}@bu.edu

Abstract

New architectures have recently been proposed and deployed to support end-to-end advance reservation of network resources. These architectures rely on the use a centralized scheduler, which may be unpractical in large or administratively heterogeneous networks. In this work, we explore and demonstrate the feasibility of implementing distributed solutions for advance reservation. We introduce a new distributed, distance-vector algorithm, called Distributed Advance Reservation (DAR), that provably returns the earliest time possible for setting up a connection between any two nodes. Our main findings in this context are the following: (i) we prove that widest path routing and path switching (i.e, allowing a connection to switch between different paths) are necessary to guarantee earliest scheduling; (ii) we propose a novel approach for loop-free distributed widest path routing, leveraging the recently proposed DIV framework. Our routing results directly extend to on-demand QoS routing problems.

A General Algorithm for Detecting Faults under the Comparison Diagnosis Model

Iain A. Stewart
School of Engineering and Computing Sciences
Durham University
Durham, U.K.
Email: i.a.stewart@durham.ac.uk

Abstract

We develop a widely applicable algorithm to solve the fault diagnosis problem in certain distributed-memory multiprocessor systems in which there are a limited number of faulty processors. In particular, we prove that if the underlying graph forming the interconnection network has connectivity no less than its diagnosability δ and can be partitioned into enough connected components of large enough size then given a syndrome of test results under the comparison diagnosis model resulting from some set of faulty nodes of size at most δ, we can find the actual set of faulty nodes with time complexity $O(\Delta N)$, where Δ is the maximal degree of any node of the graph and N is the number of nodes.

Broadcasting on Large Scale Heterogeneous Platforms under the Bounded Multi-Port Model

Olivier Beaumont and Lionel Eyraud-Dubois
INRIA Bordeaux – Sud-Ouest
University of Bordeaux, LaBRI
Bordeaux, France
Email: obeaumon@labri.fr, eyraud@labri.fr

Shailesh Kumar Agrawal
INRIA Bordeaux – Sud-Ouest
Bordeaux, France
and
Department of CSE
Indian Institute of Technology Kanpur
Email: sagrawal@cse.iitk.ac.in

Abstract

We consider the problem of broadcasting a large message in a large scale distributed platform. The message must be sent from a source node, with the help of the receiving peers which may forward the message to other peers. In this context, we are interested in maximizing the throughput (i.e. the maximum streaming rate, once steady state has been reached).

The platform model does not assume that the topology of the platform is known in advance: we consider an Internet-like network, with complete potential connectivity. Furthermore, the model associates to each node local properties (incoming and outgoing bandwidth), and the goal is to build an overlay which will be used to perform the broadcast operation. We model contentions using the bounded multi-port model: a processor can be involved simultaneously in several communications, provided that its incoming and outgoing bandwidths are not exceeded. For the sake of realism, it is also necessary to bound the number of simultaneous connections that can be opened at a given node (i.e. its outdegree).

We prove that unfortunately, this additional constraint makes the problem of maximizing the overall throughput NP-Complete. On the other hand, we also propose a polynomial time algorithm to solve this problem, based on a slight resource augmentation on the outdegree of the nodes.

On the Importance of Bandwidth Control Mechanisms for Scheduling on Large Scale Heterogeneous Platforms

Olivier Beaumont and Hejer Rejeb
INRIA Bordeaux – Sud-Ouest
University of Bordeaux, LaBRI
Bordeaux, France

Abstract

We study three scheduling problems (file redistribution, independent tasks scheduling and broadcasting) on large scale heterogeneous platforms under the Bounded Multi-port Model. In this model, each node is associated to an incoming and outgoing bandwidth and it can be involved in an arbitrary number of communications, provided that neither its incoming nor its outgoing bandwidths are exceeded. This model well corresponds to modern networking technologies, it can be used when programming at TCP level and is also implemented in modern message passing libraries such as MPICH2. We prove, using the three above mentioned scheduling problems, that this model is tractable and that even very simple distributed algorithms can achieve optimal performance, provided that we can enforce bandwidth sharing policies. Our goal is to assert the necessity of such QoS mechanisms, that are now available in the kernels of modern operating systems, to achieve optimal performance. We prove that implementations of optimal algorithms that do not enforce prescribed bandwidth sharing can fail by a large amount if TCP contention mechanisms only are used. More precisely, for each considered scheduling problem, we establish upper bounds on the performance loss than can be induced by TCP bandwidth sharing mechanisms, we prove that these upper bounds are tight by exhibiting instances achieving them and we provide a set of simulations using SimGRID to analyze the practical impact of bandwidth control mechanisms.

Session 2
Scientific Computing with GPUs

Improving Numerical Reproducibility and Stability in Large-Scale Numerical Simulations on GPUs

Michela Taufer[1], Omar Padron[2], Philip Saponaro[1] and Sandeep Patel[3]
[1]Dept. of Computer & Inf. Sciences
University of Delaware
Email: {taufer, saponaro}@udel.edu
[2] New Jersey Center for Science, Tech., and Math. Education
Kean University
Email: padrono@kean.edu
[3]Dept. of Chemistry
University of Delaware
Email: patel@udel.edu

Abstract

The advent of general purpose graphics processing units (GPGPUs) brings about a whole new platform for running numerically intensive applications at high speeds. Their multi-core architectures enable large degrees of parallelism via a massively multi-threaded environment. Molecular dynamics (MD) simulations are particularly well-suited for GPUs because their computations are easily parallelizable. Significant performance improvements are observed when single precision floating-point arithmetic is used. However, this performance comes at the cost of accuracy: it is widely acknowledged that constant-energy (NVE) MD simulations accumulate errors as the simulation proceeds due to the inherent errors associated with integrators used for propagating the coordinates. A consequence of this numerical integration is the drift of potential energy as the simulation proceeds. Double precision arithmetic partially corrects this drifting, but is significantly slower than single precision, comparable to CPU performance. To address this problem, we extend the approaches of previous literature to improve numerical reproducibility and stability in MD simulations, while assuring efficiency and performance comparable to that when using the GPU hardware implementation of single precision arithmetic. We present development of a library of mathematical functions that use fast and efficient algorithms to fix the error produced by the equivalent operations performed by GPU. We successfully validate the library with a suite of synthetic codes emulating the MD behavior on GPUs.

Implementing the Himeno Benchmark with CUDA on GPU Clusters

Everett H. Phillips and Massimiliano Fatica
NVIDIA Corporation
Santa Clara, California, United States
ephillips@nvidia.com, mfatica@nvidia.com

Abstract

This paper describes the use of CUDA to accelerate the Himeno benchmark on clusters with GPUs. The implementation is designed to optimize memory bandwidth utilization. Our approach achieves over 83% of the theoretical peak bandwidth on a NVIDIA Tesla C1060 GPU and performs at over 50 GFlops. A multi-GPU implementation that utilizes MPI alongside CUDA streams to overlap GPU execution with data transfers allows linear scaling and performs at over 800 GFlops on a cluster with 16 GPUs. The paper presents the optimizations required to achieve this level of performance.

Direct Self-Consistent Field Computations on GPU Clusters

Guochun Shi and Volodymyr Kindratenko
National Center for Supercomputing Applications
University of Illinois at Urbana-Champaign
Urbana, IL, USA
{gshi,kindr}@ncsa.illinois.edu

Ivan Ufimtsev and Todd Martinez
Department of Chemistry
Stanford University
Stanford, CA, USA
{ufimtsev,todd.martinez}@stanford.edu

Abstract

We present an implementation of one of the direct self-consistent-field (DSCF) calculation techniques, the restricted Hartree-Fock method, on a high-performance computing cluster outfitted with graphics processing units (GPUs) and demonstrate its effectiveness and scalability up to 128 cluster nodes on molecules of as many as 1,732 atoms. We discuss the overall parallel application architecture that relies on message passing interface for distributing workload among GPU cluster nodes and POSIX threads to manage the use of GPUs internal to each node. This approach of combining coarse and fine-grain parallelism on a distributed memory system allows to perform DSCF calculations on molecules that up until now have been unattainable due to the excessive computational requirements.

Parallelization of Tau-Leap Coarse-Grained Monte Carlo Simulations on GPUs

Lifan Xu and Michela Taufer
Dept. of Computer & Inf. Sciences
University of Delaware
Email: xulifan, taufer@udel.edu

Stuart Collins and Dionisios G. Vlachos
Dept. of Chemical Engineering
University of Delaware
Email: vlachos@udel.edu

Abstract

The Coarse-Grained Monte Carlo (CGMC) method is a multi-scale stochastic mathematical and simulation framework for spatially distributed systems. CGMC simulations are important tools for studying phenomena such as catalysis, crystal growth, surface diffusion, phase transitions on single crystals, and cell membrane receptor dynamics. In parallel CGMC, the tau-leap method is used for parallel simulations that are executed on traditional CPU clusters in a master-slave setting. Unfortunately the communications between master and slaves negatively impact speedup and scalability.

In this paper, we explore the potentials of GPUs for the tau-leap method and we present an extensive performance evaluation that leads to the most suitable degree of parallelism for this method under different simulation profiles. We show how the efficient parallelization of the tau-leap method for GPUs includes (1) the redefinition of its data structures, (2) the redesign of its algorithm, and (3) the selection of the most appropriate degree of parallelism (i.e., fine-grained or course-gained) on a single GPU or multiple GPUs. Exceptional performance improvements can thus be achieved for this method.

Session 3

Data Storage and Memory Systems

DEBAR: A Scalable High-Performance De-duplication Storage System for Backup and Archiving

Tianming Yang[1], Hong Jiang[2], Dan Feng[1,3], Zhongying Niu[1], Ke Zhou [1] and Yaping Wan[1]
[1]School of Computer, Huazhong University of Science and Technology
Wuhan National Laboratory for Optoelectronics, Wuhan, 430074, China
Email: ytmzqyy@yahoo.cn, niuzhy@gmail.com
[3] Corresponding author: dfeng@hust.edu.cn
[2]Department of Computer Science and Engineering
University of Nebraska-Lincoln Lincoln, NE 68588, USA
Email: jiang@cse.unl.edu

Abstract

Driven by the increasing demand for large-scale and high-performance data protection, disk-based de-duplication storage has become a new research focus of the storage industry and research community where several new schemes have emerged recently. So far these systems are mainly inline de-duplication approaches, which are centralized and do not lend themselves easily to be extended to handle global de-duplication in a distributed environment. We present DEBAR, a de-duplication storage system designed to improve capacity, performance and scalability for de-duplication backup/archiving. DEBAR performs post-processing de-duplication, where backup streams are de-duplicated and cached on server-disks through an in-memory preliminary filter in phase I, and then completely de-duplicated in-batch in phase II. By decentralizing fingerprint lookup and update, DEBAR supports a cluster of servers to perform de-duplication backup in parallel, and is shown to scale linearly in both write throughput and physical capacity, achieving an aggregate throughput of 1.7GB/s and supporting a physical capacity of 2PB with 16 backup servers.

HPDA: A Hybrid Parity-based Disk Array for Enhanced Performance and Reliability

Bo Mao[1], Hong Jiang[2], Dan Feng[1], Suzhen Wu[1], Jianxi Chen[1], Lingfang Zeng[1] and Lei Tian[1 2]
1Wuhan National Laboratory for Optoelectronics
School of Computer Science and Technology
Huazhong University of Science and Technology, Wuhan, 430074, China
[2]Department of Computer Science & Engineering
University of Nebraska-Lincoln, USA
Corresponding author: dfeng@hust.edu.cn
{maobo.hust, suzhen66}@gmail.com, {jiang, tian}@cse.unl.edu, {chenjx, lfzeng, ltian}@hust.edu.cn

Abstract

A single flash-based Solid State Drive (SSD) can not satisfy the capacity, performance and reliability requirements of a modern storage system supporting increasingly demanding data-intensive computing applications. Applying RAID schemes to SSDs to meet these requirements, while a logical and viable solution, faces many challenges. In this paper, we propose a Hybrid Parity-based Disk Array architecture, HPDA, which combines a group of SSDs and two hard disk drives (HDDs) to improve the performance and reliability of SSD-based storage systems. In HPDA, the SSDs (data disks) and part of one HDD (parity disk) compose a RAID4 disk array. Meanwhile, a second HDD and the free space of the parity disk are mirrored to form a RAID1-style write buffer that temporarily absorbs the small write requests and acts as a surrogate set during recovery when a disk fails. The write data is reclaimed back to the data disks during the lightly loaded or idle periods of the system. Reliability analysis shows that the reliability of HPDA, in terms of MTTDL (Mean Time To Data Loss), is better than that of either pure HDD-based or SSD-based disk array. Our prototype implementation of HPDA and performance evaluations show that HPDA significantly outperforms either HDD-based or SSD-based disk array.

Fine-Grained QoS Scheduling for PCM-based Main Memory Systems

Ping Zhou[1], Yu Du[2], Youtao Zhang[2] and Jun Yang[1]

[1]{piz7, juy9}@pitt.edu
Electrical and Computer Engineering Department
University of Pittsburgh
Pittsburgh, PA 15261

[2]{fisherdu, zhangyt}@cs.pitt.edu
Computer Science Department
University of Pittsburgh
Pittsburgh, PA 15260

Abstract

With wide adoption of chip multiprocessors (CMPs) in modern computers, there is an increasing demand for large capacity main memory systems. The emerging PCM (Phase Change Memory) technology has unique power and scalability advantages and is regarded as a promising candidate among new memory technologies. When scheduling a mix of applications of different priority levels, it is often important to provide tunable QoS (Quality-of-Service) for the applications with high priority. However due to the slow PCM cell access, and the destructive interferences among concurrent applications, existing memory scheduling schemes lack the flexibility to tune QoS in a wide range, in particular to the level close or equal to that of stand-alone execution. In this paper we propose a novel QoS scheduling scheme that utilizes *request preemption* and *row buffer partition* that enable QoS tuning at a fine-granularity. That is, they can tune the request queuing time and the PCM bank service time for the high priority requests. Our experimental results show that the proposed scheme achieves 1.7× ∼10× QoS tuning range while introducing negligible area and energy overheads.

Performance Impact of Resource Contention in Multicore Systems

Robert Hood[1], Haoqiang Jin, Piyush Mehrotra, Johnny Chang[1], Jahed Djomehri[1],
Sharad Gavali, Dennis Jespersen, Kenichi Taylor[2] and Rupak Biswas
NAS Divsion.NASA Ames Research Center
Moffett Field, CA, USA
[1]Computer Sciences Corportation [2]Silicon Graphics International
Email: {robert.hood,haoqiang.jin,piyush.mehrotra,johnny.chang,jahed.djomehri,
sharad.gavali,dennis.jespersen,rupak.biswas}@nasa.gov, kenichi@sgi.com

Abstract

Resource sharing in commodity multicore processors can have a significant impact on the performance of production applications. In this paper we use a differential performance analysis methodology to quantify the costs of contention for resources in the memory hierarchy of several multicore processors used in high-end computers. In particular, by comparing runs that bind MPI processes to cores in different patterns, we can isolate the effects of resource sharing. We use this methodology to measure how such sharing affects the performance of four applications of interest to NASA—OVERFLOW, MITgcm, Cart3D, and NCC. We also use a subset of the HPCC benchmarks and hardware counter data to help interpret and validate our findings. We conduct our study on high-end computing platforms that use four different quadcore microprocessors—Intel Clovertown, Intel Harpertown, AMD Barcelona, and Intel Nehalem-EP. The results help further our understanding of the requirements these codes place on their production environments and also of each computer's ability to deliver performance.

Session 4
Fault Tolerance

Improving the Performance of Hypervisor-Based Fault Tolerance

Jun Zhu, Wei Dong, Zhefu Jiang, Xiaogang Shi and Zhen Xiao
School of Electronics Engineering
& Computer Science
Peking University, Beijing, China
{zhujun, dongwei, jzf, sxg, xiaozhen}@net.pku.edu.cn

Xiaoming Li
State Key Laboratory of Advanced Optical
Communication Systems & Networks
Peking University, Beijing, China
lxm@pku.edu.cn

Abstract

Hypervisor-based fault tolerance (HBFT), a checkpoint-recovery mechanism, is an emerging approach to sustaining mission-critical applications. Based on virtualization technology, HBFT provides an economic and transparent solution. However, the advantages currently come at the cost of substantial overhead during failure-free, especially for memory intensive applications.

This paper presents an in-depth examination of HBFT and options to improve its performance. Based on the behavior of memory accesses among checkpointing epochs, we introduce two optimizations, *read fault reduction* and *write fault prediction*, for the memory tracking mechanism. These two optimizations improve the mechanism by 31.1% and 21.4% respectively for some application. Then, we present *software-superpage* which efficiently maps large memory regions between virtual machines (VM). By the above optimizations, HBFT is improved by a factor of 1.4 to 2.2 and it achieves a performance which is about 60% of that of the native VM.

Supporting Fault Tolerance in a Data-Intensive Computing Middleware

Tekin Bicer, Wei Jiang and Gagan Agrawal
Department of Computer Science and Engineering
The Ohio State University
Columbus, OH 43210

Abstract

Over the last 2-3 years, the importance of data-intensive computing has increasingly been recognized, closely coupled with the emergence and popularity of map-reduce for developing this class of applications. Besides programmability and ease of parallelization, fault tolerance is clearly important for data-intensive applications, because of their long running nature, and because of the potential for using a large number of nodes for processing massive amounts of data. Fault-tolerance has been an important attribute of map-reduce as well in its Hadoop implementation, where it is based on replication of data in the file system.

Two important goals in supporting fault-tolerance are low overheads and efficient recovery. With these goals, this paper describes a different approach for enabling data-intensive computing with fault-tolerance. Our approach is based on an API for developing data-intensive computations that is a variation of map-reduce, and it involves an explicit programmer-declared reduction object. We show how more efficient fault-tolerance support can be developed using this API. Particularly, as the reduction object represents the state of the computation on a node, we can periodically cache the reduction object from every node at another location and use it to support failure-recovery.

We have extensively evaluated our approach using two data-intensive applications. Our results show that the overheads of our scheme are extremely low, and our system outperforms Hadoop both in absence and presence of failures.

A High-Performance Fault-Tolerant Software Framework for Memory on Commodity GPUs

Naoya Maruyama and Akira Nukada
GSIC, Tokyo Institute of Technology
JST CREST
Email: naoya,nukada@matsulab.is.titech.ac.jp

Satoshi Matsuoka
GSIC, Tokyo Institute of Technology
National Institute of Informatics
JST CREST
Email: matsu@is.titech.ac.jp

Abstract

As GPUs are increasingly used to accelerate HPC applications by allowing more flexibility and programmability, their fault tolerance is becoming much more important than before when they were used only for graphics. The current generation of GPUs, however, does not have standard error detection and correction capabilities, such as SEC-DED ECC for DRAM, which is almost always exercised in HPC servers. We present a high-performance software framework to enhance commodity off-the-shelf GPUs with DRAM fault tolerance. It combines data coding for detecting bit-flip errors and checkpointing for recovering computations when such errors are detected. We analyze performance of data coding in GPUs and present optimizations geared toward memory-intensive GPU applications. We present performance studies of the prototype implementation of the framework and show that the proposed framework can be realized with negligible overheads in compute intensive applications such as N-body problem and matrix multiplication, and as low as 35% in a highly-efficient memory intensive 3-D FFT kernel.

Scalable Failure Recovery for High-performance Data Aggregation

Dorian C. Arnold
Department of Computer Science
University of New Mexico
Albuquerque, New Mexico USA
darnold@cs.unm.edu

Barton P. Miller
Computer Sciences Department
University of Wisconsin
Madison, Wisconsin USA
darnold@cs.unm.edu

Abstract

Many high-performance tools, applications and infrastructures, such as Paradyn, STAT, TAU, Ganglia, SuperMon, Astrolabe, Borealis, and MRNet, use data aggregation to synthesize large data sets and reduce data volumes while retaining relevant information content. Hierarchical or tree-based overlay networks (TBONs) are often used to execute data aggregation operations in a scalable, piecewise fashion. In this paper, we present state compensation, a scalable failure recovery model for high-bandwidth, low-latency TBON computations. By leveraging inherently redundant state information found in many TBON computations, state compensation avoids explicit state replication (for example, process checkpoints and message logging) and incurs no overhead in the absence of failures. Further, when failures do occur, state compensation uses a weak data consistency model and localized protocols that allow processes to recover from failures independently and responsively.

Based on a formal specification of our data aggregation model, we have validated state compensation and identified its assumptions and limitations: state compensation requires that data aggregation operations be associative, commutative and idempotent. In this paper, we describe the fundamental state compensation concepts and a prototype implementation integrated into the MRNet TBON infrastructure. Our experiments with this framework suggest that for TBONs supporting up to millions of application processes, state compensation can yield millisecond recovery latencies and inconsequential application perturbation.

Session 5
Sorting

High Performance Comparison-Based Sorting Algorithm on Many-Core GPUs

Xiaochun Ye[1], Dongrui Fan[1], Wei Lin[1], Nan Yuan[1] and Paolo Ienne[1,2]
[1]Key Laboratory of Computer System and Architecture
Institute of Computing Technology (ICT)
Chinese Academy of Sciences, Beijing, China
yexiaochun, fandr, linwei, yuannan@ict.ac.cn
[2]Ecole Polytechnique Fdrale de Lausanne (EPFL)
School of Computer and Communication Sciences
CH-1015 Lausanne, Switzerland
Paolo.Ienne@epfl.ch

Abstract

Sorting is a kernel algorithm for a wide range of applications. In this paper, we present a new algorithm, GPU-Warpsort, to perform comparison-based parallel sort on Graphics Processing Units (GPUs). It mainly consists of a bitonic sort followed by a merge sort.

Our algorithm achieves high performance by efficiently mapping the sorting tasks to GPU architectures. Firstly, we take advantage of the synchronous execution of threads in a warp to eliminate the barriers in bitonic sorting network. We also provide sufficient homogeneous parallel operations for all the threads within a warp to avoid branch divergence. Furthermore, we implement the merge sort efficiently by assigning each warp independent pairs of sequences to be merged and by exploiting totally coalesced global memory accesses to eliminate the bandwidth bottleneck.

Our experimental results indicate that GPU-Warpsort works well on different kinds of input distributions, and it achieves up to 30optimized comparison-based GPU sorting algorithm on input sequences with millions of elements.

GPU Sample Sort

Nikolaj Leischner, Vitaly Osipov and Peter Sanders
Dept. of Computer Sciences
Karlsruhe Institute of Technology
Karlsruhe, Germany
Email: nikolaj.leischner@student.kit.edu; osipov,sanders@kit.edu

Abstract

In this paper, we present the design of a sample sort algorithm for manycore GPUs. Despite being one of the most efficient comparison-based sorting algorithms for distributed memory architectures its performance on GPUs was previously unknown. For uniformly distributed keys our sample sort is at least 25% and on average 68% faster than the best comparison-based sorting algorithm, GPU Thrust merge sort, and on average more than 2 times faster than GPU quicksort. Moreover, for 64-bit integer keys it is at least 63% and on average 2 times faster than the highly optimized GPU Thrust radix sort that directly manipulates the binary representation of keys. Our implementation is robust to different distributions and entropy levels of keys and scales almost linearly with the input size. These results indicate that multi-way techniques in general and sample sort in particular achieve substantially better performance than two-way merge sort and quicksort.

Highly Scalable Parallel Sorting

Edgar Solomonik and Laxmikant V. Kalé
Department of Computer Science
University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA
E-mail: solomon2@illinois.edu, kale@illinois.edu

Abstract

Sorting is a commonly used process with a wide breadth of applications in the high performance computing field. Early research in parallel processing has provided us with comprehensive analysis and theory for parallel sorting algorithms. However, modern supercomputers have advanced rapidly in size and changed significantly in architecture, forcing new adaptations to these algorithms. To fully utilize the potential of highly parallel machines, tens of thousands of processors are used. Efficiently scaling parallel sorting on machines of this magnitude is inhibited by the communication-intensive problem of migrating large amounts of data between processors. The challenge is to design a highly scalable sorting algorithm that uses minimal communication, maximizes overlap between computation and communication, and uses memory efficiently. This paper presents a scalable extension of the Histogram Sorting method, making fundamental modifications to the original algorithm in order to minimize message contention and exploit overlap. We implement Histogram Sort, Sample Sort, and Radix Sort in Charm++ and compare their performance. The choice of algorithm as well as the importance of the optimizations is validated by performance tests on two predominant modern supercomputer architectures: XT4 at ORNL (Jaguar) and Blue Gene/P at ANL (Intrepid).

Session 6
Scheduling

A Scheduling Framework for Large-Scale, Parallel, and Topology-Aware Applications

Valentin Kravtsov, Pavel Bar, David Carmeli and Assaf Schuster
Computer Science Department
Technion – Israel Institute of Technology
Technion City, Haifa, Israel
{svali_ds, bpavel, carmelid, assaf}@cs.technion.ac.il

Martin Swain
Systems Biology Research Group
University of Ulster
Cromore Road, Coleraine, BT52 1SA,
Northern Ireland, UK
mt.swain@ulster.ac.uk

Abstract

Scheduling of large-scale, distributed topology-aware applications requires that not only the properties of the requested machines be considered, but also the properties of the machines' interconnections. This requirement severely complicates the scheduling process, as even a matching between a single multi-processors task and available machines in a single time slot becomes an NP-complete problem with no polynomial approximation. In this paper we propose a complete scheduling framework for multi-cluster, heterogeneous environments that provides, in practice, an efficient solution for the scheduling of topology-aware applications. The proposed framework is very flexible as it is composed of pluggable components and can be easily configured to support a variety of scheduling policies. We also describe three novel scheduling and coallocation algorithms that were developed and plugged into the framework. The proposed scheduling framework was integrated into the QosCosGrid system, where it is used as the main decision-making module.

Load Regulating Algorithm for Static-Priority Task Scheduling on Multiprocessors

Risat Mahmud Pathan and Jan Jonsson
Department of Computer Science and Engineering
Chalmers University of Technology, Sweden
Email:{risat,janjo}@chalmers.se

Abstract

This paper proposes a fixed-priority partitioned scheduling algorithm for periodic tasks on multiprocessors. A new technique for assigning tasks to processors is developed and the schedulability of the algorithm is analyzed for worst-case performance. We prove that, if the workload (utilization) of a given task set is less than or equal to 55.2% of the total processing capacity on m processors, then all tasks meet their deadlines. During task assignment, the total work load is regulated to the processors in such a way that a subset of the processors are guaranteed to have an individual processor load of at least 55.2%. Due to such load regulation, our algorithm can be used efficiently as an admission controller for online task scheduling. And this online algorithm is scalable with increasing number of cores in chip multiprocessors.

In addition, our scheduling algorithm possesses two properties that may be important for the system designer. The first one guarantees that if task priorities are fixed before task assignment they do not change during task assignment and execution, thereby facilitating debugging during development and maintenance of the system. The second property guarantees that at most m/2 tasks are split, thereby keeping the run-time overhead as caused by task splitting low.

Scheduling Algorithms for Linear Workflow Optimization

K. Agrawal[1], A. Benoit[2], L. Magnan[2] and Y. Robert[2]
[1]Washington University in St. Louis, USA [2]ENS Lyon et Université de Lyon, France
kunal@cse.wustl.edu, {Anne.Benoit|Loic.Magnan|Yves.Robert}@ens-lyon.fr

Abstract

Pipelined workflows are a popular programming paradigm for parallel applications. In these workflows, the computation is divided into several stages, and these stages are connected to each other through first-in first-out channels. In order to execute these workflows on a parallel machine, we must first determine the *mapping* of the stages onto the various processors on the machine. After finding the mapping, we must compute the *schedule*, i.e., the order in which the various stages execute on their assigned processors. In this paper, we assume that the mapping is given and explore the latter problem of scheduling, particularly for *linear* workflows. Linear workflows are those in which dependencies between stages can be represented by a linear graph. The objective of the scheduling algorithm is either to minimize the period (the inverse of the throughput), or to minimize the latency (response time), or both. We consider two realistic execution models: the one-port model (all operations are serialized) and the multi-port model (bounded communication capacities and communication/computation overlap). In both models, finding a schedule to minimize the latency is easy. However, computing the schedule to minimize the period is NP-hard in the one-port model, but can be done in polynomial time in the multi-port model. We also present an approximation algorithm to minimize the period in the one-port model. Finally, the bi-criteria problem, which consists in finding a schedule respecting a given period and a given latency, is NP-hard in both models.

Hypergraph-based Task-Bundle Scheduling Towards Efficiency and Fairness in Heterogeneous Distributed Systems

Han Zhao, Xinxin Liu and Xiaolin Li
Scalable Software Systems Laboratory
Department of Computer Science
Oklahoma State University
Stillwater, OK 74078, USA
Email: {haz, xinxin, xiaolin}@cs.okstate.edu

Abstract

This paper investigates scheduling loosely coupled task-bundles in highly heterogeneous distributed systems. Two allocation quality metrics are used in pay-per-service distributed applications: efficiency in terms of social welfare, and fairness in terms of envy-freeness. The first contribution of this work is that we build a unified hypergraph scheduling model under which efficiency and fairness are compatible with each other. Second, in the scenario of budget-unawareness, we formulate a strategic algorithm design for distributed negotiations among autonomous self-interested computing peers and prove its convergence to complete local efficiency and envy-freeness. Third, we add budget limitation to the allocation problem and propose a class of hill-climbing heuristics in favor of different performance metrics. Finally we conduct extensive simulations to validate the performance of all the proposed algorithms. The results show that the decentralized hypergraph scheduling method is scalable, and yields desired allocation performance in various scenarios.

Session 7

Performance/Scalability Improvement for Scientific Applications

Improving the Performance of Uintah: A Large-Scale Adaptive Meshing Computational Framework

Justin Luitjens
School of Computing
University of Utah
Salt Lake City, Utah 84112
luitjens@cs.utah.edu

Martin Berzins
School of Computing
University of Utah
Salt Lake City, Utah 84112
mb@sci.utah.edu

Abstract

Uintah is a highly parallel and adaptive multi-physics framework created by the Center for Simulation of Accidental Fires and Explosions in Utah. Uintah, which is built upon the Common Component Architecture, has facilitated the simulation of a wide variety of fluid-structure interaction problems using both adaptive structured meshes for the fluid and particles to model solids. Uintah was originally designed for, and has performed well on, about a thousand processors. The evolution of Uintah to use tens of thousands processors has required improvements in memory usage, data structure design, load balancing algorithms and cost estimation in order to improve strong and weak scalability up to 98,304 cores for situations in which the mesh used varies adaptively and also cases in which particles that represent the solids move from mesh cell to mesh cell.

Optimizing and Tuning the Fast Multipole Method for State-of-the-Art Multicore Architectures

Aparna Chandramowlishwaran[1,2], Samuel Williams[1],
Leonid Oliker[1], Ilya Lashuk[2], George Biros[2] and Richard Vuduc[2]
[1]CRD, Lawrence Berkeley National Laboratory, Berkeley, CA 94720
[2]College of Computing, Georgia Institute of Technology, Atlanta, GA

Abstract

This work presents the first extensive study of singlenode performance optimization, tuning, and analysis of the fast multipole method (FMM) on modern multicore systems. We consider single- and double-precision with numerous performance enhancements, including low-level tuning, numerical approximation, data structure transformations, OpenMP parallelization, and algorithmic tuning.

Among our numerous findings, we show that optimization and parallelization can improve doubleprecision performance by 25× on Intel's quad-core Nehalem, 9.4× on AMD's quad-core Barcelona, and 37.6× on Sun's Victoria Falls (dual-sockets on all systems). We also compare our single-precision version against our prior state-of-the-art GPU-based code and show, surprisingly, that the most advanced multicore architecture (Nehalem) reaches parity in both performance and power efficiency with NVIDIA's most advanced GPU architecture.

Parallelization of DQMC Simulation for Strongly Correlated Electron Systems

Che-Rung Lee
Dept. of Computer Science
National Tsing-Hua University
Hsin-Chu, Taiwan
cherung@cs.nthu.edu.tw

I-Hsin Chung
IBM T.J. Watson Research Center
Yorktown Heights, NY USA
ihchung@us.ibm.com

Zhaojun Bai
Dept. of of Computer Science
University of California, Davis
Davis, CA USA
bai@cs.ucdavis.edu

Abstract

Determinant Quantum Monte Carlo (DQMC) simulation has been widely used to reveal macroscopic properties of strong correlated materials. However, parallelization of the DQMC simulation is extremely challenging duo to the serial nature of underlying Markov chain and numerical stability issues. We extend previous work with novelty by presenting a hybrid granularity parallelization (HGP) scheme that combines algorithmic and implementation techniques to speed up the DQMC simulation. From coarse-grained parallel Markov chain and task decompositions to fine-grained parallelization methods for matrix computations and Green's function calculations, the HGP scheme explores the parallelism on different levels and maps the underlying algorithms onto different computational components that are suitable for modern high performance heterogeneous computer systems. Practical techniques, such as communication and computation overlapping, message compression and load balancing are also considered in the proposed HGP scheme. We have implemented the DQMC simulation with the HGP scheme on an IBM Blue Gene/P system. The effectiveness of the new scheme is demonstrated through both theoretical analysis and performance results. Experiments have shown over a factor of 80 speedups on an IBM Blue Gene/P system with 1,014 computational processors.

Parallel I/O Performance: From Events to Ensembles

Andrew Uselton[1], Mark Howison[1], Nicholas J. Wright[1], David Skinner[1],
Noel Keen[1], John Shalf[1], Karen L. Karavanic[2] and Leonid Oliker[1]
[1]CRD/NERSC, Lawrence Berkeley National Laboratory Berkeley, CA 94720
[2]Portland State University, Portland, OR 97207-0751

Abstract

Parallel I/O is fast becoming a bottleneck to the research agendas of many users of extreme scale parallel computers. The principle cause of this is the concurrency explosion of high-end computation, coupled with the complexity of providing parallel file systems that perform reliably at such scales. More than just being a bottleneck, parallel I/O performance at scale is notoriously variable, being influenced by numerous factors inside and outside the application, thus making it extremely difficult to isolate cause and effect for performance events. In this paper, we propose a statistical approach to understanding I/O performance that moves from the analysis of performance events to the exploration of performance ensembles. Using this methodology, we examine two I/O-intensive scientific computations from cosmology and climate science, and demonstrate that our approach can identify application and middleware performance deficiencies — resulting in more than 4× run time improvement for both examined applications.

Session 8
Network Architecture and Algorithms

Achieve Constant Performance Guarantees using Asynchronous Crossbar Scheduling without Speedup

Deng Pan
Florida International University
Miami, FL

Kia Makki
TUM
Miami, FL

Niki Pissinou
Florida International University
Miami, FL

Abstract

Buffered crossbar switches are special crossbar switches with a small exclusive buffer at each crosspoint of the crossbar. They demonstrate unique advantages, such as variable length packet handling and distributed scheduling, over traditional unbuffered crossbar switches. The current main approach for buffered crossbar switches to provide performance guarantees is to emulate push-in-first-out output queued switches. However, such an approach has several drawbacks, and in particular it has difficulty in providing tight constant performance guarantees. To address the issue, we propose in this paper the guaranteed-performance asynchronous packet scheduling (GAPS) algorithm for buffered crossbar switches. GAPS intends to provide tight performance guarantees, and requires no speedup. It directly handles variable length packets without segmentation and reassembly, and makes scheduling decisions in a distributed manner. We show by theoretical analysis that GAPS achieves constant performance guarantees. We also prove that GAPS has a bounded crosspoint buffer size of $3L$, where L is the maximum packet length. Finally, we present simulation data to verify the analytical results and show the effectiveness of GAPS.

Distributive Waveband Assignment in Multi-granular Optical Networks

Yang Wang
Department of Computer Science
Georgia State University
Atlanta, Georgia 30303
Email: ywang39@student.gsu.edu

Xiaojun Cao
Department of Computer Science
Georgia State University
Atlanta, Georgia 30303
Email: cao@cs.gsu.edu

Abstract

To handle the challenge of increasing node size in wavelength routing networks (WRNs), waveband switching (WBS) is introduced to group multiple wavelengths together as a band or fiber and switch them using a single port whenever possible. Literature studies with the off-line traffic confirm that WBS can effectively reduce the port count and cost as well as control complexity. In the cases with online traffic, both the port reduction and call blocking probability should be considered due to the unknown traffic pattern and limited resources. In this work, we first analyze a reconfigurable switching architecture and the blocking probability in WBS networks. Based on the analysis, we then propose a novel dynamic graph-based waveband assignment algorithm in conjunction with adaptive routing. The proposed algorithm employs the ant optimization techniques to reduce ports and blocking probability in the network with online traffic in a distributed manner. Our simulation results show that our graph-based waveband assignment algorithm combined with the adaptive routing scheme can achieve the best performance when compared to other schemes. Our study also shows that even with limited resources, WBS can achieve port saving and an allowable blocking probability.

QoS Aware BiNoC Architecture

Shih-Hsin Lo[1], Ying-Cherng Lan[1], Hsin-Hsien Yeh[1], Wen-Chung Tsai[1], Yu-Hen Hu[2] and Sao-Jie Chen[1,3]
[1]Graduate Institute of Electronics Engineering, National Taiwan University,
Taipei, Taiwan, ROC, {r96943130, f94943068, r98943095, d95943036}@ntu.edu.tw, csj@cc.ee.ntu.edu.tw
[2]Department of Electrical and Computer Engineering, University of Wisconsin, Madison
Madison, WI53706, USA, hu@engr.wisc.edu
[3]Graduate Institute of Electrical Engineering, National Taiwan University, Taipei Taiwan, ROC

Abstract

A quality-of-service (QoS) aware, bi-directional channel NoC (BiNoC) architecture is proposed to support guarantee-service (GS) traffic while reducing packet delivery latency. By incorporating dynamically self-reconfigured bidirectional communication channels between adjacent routers, BiNoC architecture promises more flexibility for various traffic flow patterns. A novel inter-router communication protocol is proposed that prioritizes bandwidth arbitration in favor of high priority GS traffic flows. Multiple virtual channels with prioritized routing policy are also implemented to facilitate data transmission with QoS considerations. Combining these architectural innovations, the QoS aware BiNoC promises reduced latency of packet delivery and more efficient channel resource utilizations. Cycle-accurate simulations demonstrate significant performance advantage over conventional unidirectional NoC architecture equipped with hard-wired unidirectional channels.

First Experiences with Congestion Control in InfiniBand Hardware

Ernst Gunnar Gran,
Magne Eimot, Sven-Arne Reinemo,
Tor Skeie, Olav Lysne Member, IEEE
Simula Research Laboratory
Fornebu, Norway
Email: {ernstgr, magneei, svenar,
tskeie, olavly}@simula.no

Lars Paul Huse
Sun Microsystems
Email: Lars.Paul.Huse@sun.com

Gilad Shainer
Mellanox Technologies
Email: Shainer@Mellanox.com

Abstract

In lossless interconnection networks congestion control (CC) can be an effective mechanism to achieve high performance and good utilization of network resources. Without CC, congestion in one node may grow into a congestion tree that can degrade the performance severely. This degradation can affect not only contributors to the congestion, but also throttles innocent traffic flows in the network. The InfiniBand standard describes CC functionality for detecting and resolving congestion. The InfiniBand CC concept is rich in the way that it specifies a set of parameters that can be tuned in order to achieve effective CC. There is, however, limited experience with the InfiniBand CC mechanism. To the best of our knowledge, only a few simulation studies exist. Recently, InfiniBand CC has been implemented in hardware, and in this paper we present the first experiences with such equipment. We show that the implemented InfiniBand CC mechanism effectively resolves congestion and improves fairness by solving the parking lot problem, if the CC parameters are appropriately set. By conducting extensive testing on a selection of the CC parameters, we have explored the parameter space and found a subset of parameter values that leads to efficient CC for our test scenarios. Furthermore, we show that the InfiniBand CC increases the performance of the well known HPC Challenge benchmark in a congested network.

Session 9
Software Support for Using GPUs

Object-Oriented Stream Programming using Aspects

Mingliang Wang and Manish Parashar
NSF Center for Autonomic Computing
The Applied Software System Laboratory (TASSL)
Rutgers University
94 Brett Road, Piscataway, NJ 08854, USA
{mlwang, parashar}@cac.rutgers.edu

Abstract

High-performance parallel programs that efficiently utilize heterogeneous CPU+GPU accelerator systems require tuned coordination among multiple program units. However, using current programming frameworks such as CUDA leads to tangled source code that combines code for the core computation with that for device and computational kernel management, data transfers between memory spaces, and various optimizations. In this paper, we propose a programming system based on the principles of Aspect-Oriented Programming, to un-clutter the code and to improve programmability of these heterogeneous parallel systems. Specifically, we use standard C++ to describe the core computations and aspects to encapsulate all other support parts. An aspect-weaving compiler is then used to combine these two pieces of code to generate a final program. The system modularizes concerns that are hard to manage using conventional programming frameworks such as CUDA, has a small impact on existing program structure as well as performance, and as a result, simplifies the programming of accelerator-based heterogeneous parallel systems. We also present an options pricing and an n-body simulation example program to demonstrate that programs written using this system can be successfully translated to CUDA programs for NVIDIA GPU hardware and to OpenCL programs for multicore CPUs with comparable performance. For both examples, the performance of the translated code achieved ~80% of the hand-coded CUDA programs.

Optimal Loop Unrolling For GPGPU Programs

Giridhar Sreenivasa Murthy, Mahesh Ravishankar, Muthu Manikandan Baskaran and P. Sadayappan
Department of Computer Science and Engineering
The Ohio State University, Columbus, Ohio, USA
{murthyg, ravishan, baskaran, saday}@cse.ohio-state.edu

Abstract

Graphics Processing Units (GPUs) are massively parallel, many-core processors with tremendous computational power and very high memory bandwidth. With the advent of general purpose programming models such as NVIDIA's CUDA and the new standard OpenCL, general purpose programming using GPUs (GPGPU) has become very popular. However, the GPU architecture and programming model have brought along with it many new challenges and opportunities for compiler optimizations. One such classical optimization is loop unrolling. Current GPU compilers perform limited loop unrolling.

In this paper, we attempt to understand the impact of loop unrolling on GPGPU programs. We develop a semi-automatic, compile-time approach for identifying optimal unroll factors for suitable loops in GPGPU programs. In addition, we propose techniques for reducing the number of unroll factors evaluated, based on the characteristics of the program being compiled and the device being compiled to. We use these techniques to evaluate the effect of loop unrolling on a range of GPGPU programs and show that we correctly identify the optimal unroll factors. The optimized versions run up to 70 percent faster than the unoptimized versions.

Speculative Execution on Multi-GPU Systems

Gregory Diamos and Sudhakar Yalamanchili
School of Electrical and
Computer Engineering
Georgia Institute of Technology
Atlanta, Georgia 30332–0250
gregory.diamos@gatech.edu sudha@ece.gatech.edu

Abstract

The lag of parallel programming models and languages behind the advance of heterogeneous many-core processors has left a gap between the computational capability of modern systems and the ability of applications to exploit them. Emerging programming models, such as CUDA and OpenCL, force developers to explicitly partition applications into components (kernels) and assign them to accelerators in order to utilize them effectively. An accelerator is a processor with a different ISA and micro-architecture than the main CPU. These static partitioning schemes are effective when targeting a system with only a single accelerator. However, they are not robust to changes in the number of accelerators or the performance characteristics of future generations of accelerators.

In previous work, we presented the Harmony execution model for computing on heterogeneous systems with several CPUs and accelerators. In this paper, we extend Harmony to target systems with multiple accelerators using control speculation to expose parallelism. We refer to this technique as Kernel Level Speculation (KLS). We argue that dynamic parallelization techniques such as KLS are sufficient to scale applications across several accelerators based on the intuition that there will be fewer distinct accelerators than cores within each accelerator. In this paper, we use a complete prototype of the Harmony runtime that we developed to explore the design decisions and trade-offs in the implementation of KLS. We show that KLS improves parallelism to a sufficient degree while retaining a sequential programming model. We accomplish this by demonstrating good scaling of KLS on a highly heterogeneous system with three distinct accelerator types and ten processors.

Dynamic Load Balancing on Single- and Multi-GPU Systems

Long Chen[1], Oreste Villa[2], Sriram Krishnamoorthy[2] and Guang R. Gao[1]
[1]Department of Electrical & Computer Engineering
University of Delaware
Newark, DE 19716
{lochen, ggao}@capsl.udel.edu

[2]High Performance Computing
Pacific Northwest National Laboratory
Richland, WA 99352
{oreste.villa, sriram}@pnl.gov

Abstract

The computational power provided by many-core graphics processing units (GPUs) has been exploited in many applications. The programming techniques currently employed on these GPUs are not sufficient to address problems exhibiting irregular, and unbalanced workload. The problem is exacerbated when trying to effectively exploit multiple GPUs concurrently, which are commonly available in many modern systems. In this paper, we propose a task-based dynamic load-balancing solution for single- and multi-GPU systems. The solution allows load balancing at a finer granularity than what is supported in current GPU programming APIs, such as NVIDIA's CUDA. We evaluate our approach using both micro-benchmarks and a molecular dynamics application that exhibits significant load imbalance. Experimental results with a single- GPU configuration show that our fine-grained task solution can utilize the hardware more efficiently than the CUDA scheduler for unbalanced workload. On multi- GPU systems, our solution achieves near-linear speedup, load balance, and significant performance improvement over techniques based on standard CUDA APIs.

Session 10

Performance Prediction and Benchmarking Tools

Servet: A Benchmark Suite for Autotuning on Multicore Clusters

Jorge González-Domínguez, Guillermo L. Taboada, Basilio B. Fraguela, María J. Martín and Juan Touriño
Computer Architecture Group
Department of Electronics and Systems
University of A Coruña, Spain
Email: {jgonzalezd, taboada, basilio.fraguela, mariam, juan}@udc.es

Abstract

The growing complexity in computer system hierarchies due to the increase in the number of cores per processor, levels of cache (some of them shared) and the number of processors per node, as well as the high-speed interconnects, demands the use of new optimization techniques and libraries that take advantage of their features.

In this paper Servet, a suite of benchmarks focused on detecting a set of parameters with high influence in the overall performance of multicore systems, is presented. These benchmarks are able to detect the cache hierarchy, including their size and which caches are shared by each core, bandwidths and bottlenecks in memory accesses, as well as communication latencies among cores. These parameters can be used by autotuned codes to increase their performance in multicore clusters. Experimental results using different representative systems show that Servet provides very accurate estimates of the parameters of the machine architecture.

KRASH: Reproducible CPU Load Generation on Many-Core Machines

Swann Perarnau and Guillaume Huard
INRIA Moais research team, CNRS LIG lab.
Grenoble University, France
Email: perarnau,huard@imag.fr

Abstract

In this article we present KRASH, a tool for reproducible generation of system-level CPU load. This tool is intended for use in shared memory machines equipped with multiple CPU cores which are usually exploited concurrently by several users. The objective of KRASH is to enable parallel application developers to validate their resources use strategies on a partially loaded machine by replaying an observed load in concurrence with their application. To reach this objective, we present a method for CPU load generation which behaves as realistically as possible: the resulting load is similar to the load that would be produced by concurrent processes run by other users. Nevertheless, contrary to a simple run of a CPU-intensive application, KRASH is not sensitive to system scheduling decisions. The main benefit brought by KRASH is this reproducibility: no matter how many processes are present in the system the load generated by our tool strictly respects a given load profile. This last characteristic proves to be hard to achieve using simple methods because the system scheduler is supposed to share the resources fairly among running processes.

Our first contribution is a method that cooperates with the system scheduler to produce a CPU load that conforms to a desired load profile. We argue that this cooperation with the system scheduler is mandatory in the generator to reach a good reproducibility, a high precision and a low intrusiveness. Taking advantage of Linux kernel capabilities, we implemented this method in KRASH (Kernel for Reproduction and Analysis of System Heterogeneity). We have run experiments that show that KRASH provides a precise reproduction of the desired load and that it induces a very low overhead on the system. Our second contribution is a qualitative and quantitative study that compares KRASH to other tools dealing with system-level CPU load generation. To our knowledge, KRASH is the only tool that implements the generation of a dynamic load profile (a load varying with time). When used to generate a constant load, KRASH result is among the most realistic ones. Furthermore, KRASH provides more flexibility than other tools.

Power-aware MPI Task Aggregation Prediction for High-End Computing Systems

Dong Li[1], Dimitrios S. Nikolopoulos[3], Kirk Cameron[1], Bronis R. de Supinski[2] and Martin Schulz[2]
[1]Virginia Tech, Blacksburg, VA, USA, {lid,cameron}@cs.vt.edu
[2]Lawrence Livermore National Lab, Livermore, CA, USA, {bronis,schulzm}@llnl.gov
[3]FORTH-ICS and University of Crete, Heraklion, Crete, GREECE, dsn@ics.forth.gr

Abstract

Emerging large-scale systems have many nodes with several processors per node and multiple cores per processor. These systems require effective task distribution between cores, processors and nodes to achieve high levels of performance and utilization. Current scheduling strategies distribute tasks between cores according to a count of available cores, but ignore the execution time and energy implications of task aggregation (i.e., grouping multiple tasks within the same node or the same multicore processor). Task aggregation can save significant energy while sustaining or even improving performance. However, choosing an effective task aggregation becomes more difficult as the core count and the options available for task placement increase. We present a framework to predict the performance effect of task aggregation in both computation and communication phases and its impact in terms of execution time and energy of MPI programs. Our results for the NPB 3.2 MPI benchmark suite show that our framework provides accurate predictions leading to substantial energy saving through aggregation (64.87% on average and up to 70.03%) with tolerable performance loss (under 5%).

Session 11
Resource Allocation

Varying Bandwidth Resource Allocation Problem with Bag Constraints

Venkatesan T. Chakaravarthy, Vinayaka Pandit, Yogish Sabharwal, and Deva P. Seetharam
IBM Research - India,
New Delhi, INDIA
Email: {vechakra, pvinayak, ysabharwal, dseetharam}@in.ibm.com

Abstract

We consider the problem of scheduling jobs on a pool of machines. Each job requires multiple machines on which it executes in parallel. For each job, the input specifies release time, deadline, processing time, profit and the number of machines required. The total number of machines may be different at different points of time. A feasible solution is a subset of jobs and a schedule for them such that at any timeslot, the total number of machines required by the jobs active at the timeslot does not exceed the number of machines available at that timeslot. We present an $O(\log(B_{max}/B_{min}))$- approximation algorithm, where B_{max} and B_{min} are the maximum and minimum available bandwidth (maximum and minimum number of machines available over all the timeslots). Our algorithm and the approximation ratio are applicable for more a general problem that we call the Varying bandwidth resource allocation problem with bag constraints (BAGVBRAP). The BAGVBRAP problem is a generalization of some previously studied scheduling and resource allocation problems.

Decentralized Resource Management for Multi-core Desktop Grids

Jaehwan Lee, Pete Keleher and Alan Sussman
UMIACS and Department of Computer Science
University of Maryland
College Park, MD, USA
{jhlee, keleher, als}@cs.umd.edu

Abstract

The majority of CPUs now sold contain multiple computing cores. However, current desktop grid computing systems either ignore the multiplicity of cores, or treat them as distinct, independent machines. The latter approach ignores the resource contention present between cores in a single CPU, while the former approach fails to take advantage of significant computing power.

We propose a decentralized resource management framework for exploiting multi-core nodes in peer-to-peer grids. We present two new load-balancing schemes that explicitly account for the resource sharing and contention of multiple cores, and propose a simple simulation model that can represent a continuum of resource sharing among cores of a CPU. We use simulation to confirm that our two algorithms match jobs with computing nodes efficiently, and balance load during the lifetime of the computing jobs.

Dynamic Fractional Resource Scheduling for HPC Workloads

Mark Stillwell[1], Frédéric Vivien[1,2] and Henri Casanova[1]
[1]Department of Information and Computer Sciences
University of Hawai'i at Mānoa, Honolulu, U.S.A.
[2]INRIA, France

Abstract

We propose a novel job scheduling approach for homogeneous cluster computing platforms. Its key feature is the use of virtual machine technology for sharing resources in a precise and controlled manner. We justify our approach and propose several job scheduling algorithms. We present results obtained in simulations for synthetic and real-world High Performance Computing (HPC) workloads, in which we compare our proposed algorithms with standard batch scheduling algorithms. We find that our approach widely outperforms batch scheduling. We also identify a few promising algorithms that perform well across most experimental scenarios. Our results demonstrate that virtualization technology coupled with lightweight scheduling strategies affords dramatic improvements in performance for HPC workloads.

ADEPT Scalability Predictor in Support of Adaptive Resource Allocation

Arash Deshmeh
School of Computer Science
University of Windsor
Windsor, Canada
deshmeh@uwindsor.ca

Jacob Machina
School of Computer Science
University of Windsor
Windsor, Canada
machina@uwindsor.ca

Angela Sodan
School of Computer Science
University of Windsor
Windsor, Canada
acsodan@uwindsor.ca

Abstract

Adaptive resource allocation with different numbers of machine nodes provides more flexibility and significantly better potential performance for local job and grid scheduling. With the emergence of parallel computing in every-day life on multi-core systems, such schedulers will likely increase in practical relevance. A major reason why adaptive schedulers are not yet practically used is lacking knowledge of the scalability curves of the applications. Existing white-box approaches for scalability prediction are too expensive to apply them routinely. We present ADEPT, a speedup and runtime prediction tool, which is inexpensive and easy-to-use. ADEPT employs a black-box model and can be practically applied at large scale without user or administrator involvement. ADEPT requires neither program analysis and measurements nor user guesses but makes highly accurate predictions with only few observations of application runtime over different numbers of nodes/cores. ADEPT performs efficient model fitting by introducing an envelope-derivation technique to constrain the search. Additionally, ADEPT is capable of handling deviations from the underlying model by detection and automatic correction of anomalies via a fluctuation metric and by considering specific scalability patterns via multi-phase modeling. ADEPT also performs reliability judgment with potential proposal for placement of additional observations. Using MPI and OpenMP implementations of the NAS benchmarks and seven real applications, we demonstrate the effectiveness and high prediction accuracy of ADEPT for both speedup and runtime prediction, including interpolative and extrapolative cases, and show the capability of ADEPT to successfully handle special cases.

Session 12
Image Processing and Data Mining

Exploiting the Forgiving Nature of Applications for Scalable Parallel Execution

Jiayuan Meng[1,2], Anand Raghunathan[1,3], Srimat Chakradhar[1], and Surendra Byna[1]
[1]NEC Laboratories America, Princeton, NJ
[2]Department of Computer Science, University of Virginia, Charlottesville, VA
[3]School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN

Abstract

It is widely believed that most Recognition and Mining (RM) workloads can easily take advantage of parallel computing platforms because these workloads are data-parallel. Contrary to this popular belief, we present RM workloads for which conventional parallel implementations scale poorly on multi-core platforms. We identify off-chip memory transfers and overheads in the parallel runtime library as the primary bottlenecks that limit speedups to be well below the ideal linear speedup expected for data-parallel workloads. To achieve improved parallel scalability, we identify and exploit several interesting properties of RM workloads — sparsity of model updates, low spatial locality among model updates, presence of insignificant computations, and the inherently self-healing nature of these algorithms in the presence of errors. We leverage these domain-specific characteristics to improve parallel scalability in two major ways. First, we utilize data dependency relaxation to simultaneously execute multiple training iterations in parallel, thereby increasing the granularity of the parallel tasks and significantly lowering the run-time overheads of fine-grained threading. Second, we strategically drop selected computations that are insignificant to the accuracy of the final result, but account for a disproportionately large amount of off-chip (memory and coherence) traffic.

Through the application of the proposed techniques, we show that much higher speedups are possible on multi-core platforms for two important RM applications — document search using semantic indexing, and eye detection in images using generalized learning vector quantization. On an 8-core platform, we achieve application speedups of 5.5X and 7.3X compared to sequential implementations. Compared to conventional parallel implementations of these applications using Intel's TBB, the proposed techniques result in 4.3X and 4.9X improvements. Although the optimized parallel implementations are not numerically equivalent to the sequential implementations, the output quality is shown to be comparable (and within the margin of variation produced by processing the input data in a different order). We also explore error mitigation techniques that can be used to ensure that the accuracy of results is not compromised.

Fisheye Lens Distortion Correction on Multicore and Hardware Accelerator Platforms

Konstantis Daloukas, Christos D. Antonopoulos and Nikolaos Bellas
Department of Computer and Communications Engineering,
University of Thessaly, Volos, Greece
{kodalouk, cda, nbellas}@uth.gr

Sek M. Chai
Motorola, Inc.
Schaumburg, IL, USA
sek.chai@motorola.com

Abstract

Wide-angle (fisheye) lenses are often used in virtual reality and computer vision applications to widen the field of view of conventional cameras. Those lenses, however, distort images. For most real-world applications the video stream needs to be transformed, at real-time (20 frames/sec or better), back to the natural-looking, central perspective space.

This paper presents the implementation, optimization and characterization of a fisheye lens distortion correction application on three platforms: a conventional, homogeneous multicore processor by Intel, a heterogeneous multicore (Cell BE), and an FPGA implementing an automatically generated streaming accelerator. We evaluate the interaction of the application with those architectures using both high- and low-level performance metrics. In macroscopic terms, we find that today? mainstream conventional multicores are not effective in supporting real-time distortion correction, at least not with the currently commercially available core counts. Architectures, such as the Cell BE and FPGAs, offer the necessary computational power and scalability, at the expense of significantly higher development effort. Among these three platforms, only the FPGA and a fully optimized version of the code running on the Cell processor can provide real-time processing speed. In general, FPGAs meet the expectations of performance, flexibility, and low overhead. General purpose multicores are, on the other hand, much easier to program.

Large-Scale Multi-Dimensional Document Clustering on GPU Clusters

Yongpeng Zhang and Frank Mueller
Dept. of Computer Science
North Carolina State University
Raleigh, NC 27695-7534
Email:mueller@cs.ncsu.edu

Xiaohui Cui and Thomas Potok
Oak Ridge National Laboratory
Computational Sciences and Engineering Division
Oak Ridge, TN 37831
Email:cuix@ornl.gov

Abstract

Document clustering plays an important role in data mining systems. Recently, a flocking-based document clustering algorithm has been proposed to solve the problem through simulation resembling the flocking behavior of birds in nature. This method is superior to other clustering algorithms, including k-means, in the sense that the outcome is not sensitive to the initial state. One limitation of this approach is that the algorithmic complexity is inherently quadratic in the number of documents. As a result, execution time becomes a bottleneck with large number of documents.

In this paper, we assess the benefits of exploiting the computational power of Beowulf-like clusters equipped with contemporary Graphics Processing Units (GPUs) as a means to significantly reduce the runtime of flocking-based document clustering. Our framework scales up to over one million documents processed simultaneously in a sixteen-node moderate GPU cluster. Results are also compared to a four-node cluster with higher-end GPUs. On these clusters, we observe 30X-50X speedups, which demonstrate the potential of GPU clusters to efficiently solve massive data mining problems. Such speedups combined with the scalability potential and accelerator-based parallelization are unique in the domain of document-based data mining, to the best of our knowledge.

eScience in the Cloud: A MODIS Satellite Data Reprojection and Reduction Pipeline in the Windows Azure Platform

Jie Li and Marty Humphrey
Department of Computer Science
University of Virginia
Charlottesville, VA USA
{jl3yh, humphrey}@cs.virginia.edu

Deb Agarwal and Keith Jackson
Advanced Computing for Science Department
Lawrence Berkeley National Lab
Berkeley, CA USA
{daagarwal, krjackson}@lbl.gov

Catharine van Ingen
eScience Group
Microsoft Research
San Francisco, CA USA
vaningen@microsoft.com

Youngryel Ryu
Dept. Environmental Science, Policy and Management
University of California, Berkeley
Berkeley, CA USA
yryu@berkeley.edu

Abstract

The combination of low-cost sensors, low-cost commodity computing, and the Internet is enabling a new era of data-intensive science. The dramatic increase in this data availability has created a new challenge for scientists: how to process the data. Scientists today are envisioning scientific computations on large scale data but are having difficulty designing software architectures to accommodate the large volume of the often heterogeneous and inconsistent data. In this paper, we introduce a particular instance of this challenge, and present our design and implementation of a MODIS satellite data reprojection and reduction pipeline in the Windows Azure cloud computing platform. This cloud-based pipeline is designed with a goal of hiding data complexities and subsequent data processing and transformation from end users. This pipeline is highly flexible and extensible to accommodate different science data processing tasks, and can be dynamically scaled to fulfill scientists various computational requirements in a cost-efficient way. Experiments show that by running a practical large-scale science data processing job in the pipeline using 150 moderately-sized Azure virtual machine instances, we were able to produce analytical results in nearly 90X less time than was possible with a high-end desktop machine. To our knowledge, this is one of the first eScience applications to use the Windows Azure platform.

Session 13
Transactional Memory

Locality-Aware Adaptive Grain Signatures for Transactional Memories

Woojin Choi and Jeff Draper
Information Sciences Institute
University of Southern California
Marina del Rey, CA 90292, USA
{woojinch, draper}@ISI.EDU

Abstract

Transactional Memory (TM) has attracted considerable attention because it promises to increase programmer productivity by making it easier to write correct parallel programs. To maintain correctness in the face of concurrency, detecting conflicts among simultaneously running transactions is an essential element. Hardware signatures have been proposed as an area-efficient mechanism for conflict detection. A signature can summarize an unbounded amount of addresses and misses no conflicts, but could falsely declare conflicts even when no true conflict exists (false positives) due to aliasing and occupancy. Previous signature designs assume that false positives are destructive to performance and attempt to reduce the total number of false positives.

In this paper, we show that some false positives can be helpful to performance by triggering the early abortion of a transaction which would encounter a true conflict later anyway. Based on this observation, we propose an adaptive grain signature to improve performance by dynamically changing the range of address keys based on the history. With the use of adaptive grain signatures, we can increase the number of performance-friendly false positives as well as decrease the number of performance-destructive false positives.

Dynamic Analysis of the Relay Cache-Coherence Protocol for Distributed Transactional Memory

Bo Zhang and Binoy Ravindran
Department of Electrical and Computer Engineering
Virginia Tech, Blacksburg, VA 24061, USA
{alexzbzb,binoy}@vt.edu

Abstract

Transactional memory is an alternative programming model for managing contention in accessing shared in-memory data objects. Distributed transactional memory (TM) promises to alleviate difficulties with lock-based (distributed) synchronization and object performance bottlenecks in distributed systems. In distributed TM systems, both the management and consistency of a distributed transactional object are ensured by a cache-coherence protocol. The Relay protocol is a cache-coherence protocol that operates on a fixed spanning tree. The protocol efficiently reduces the total number of abortions for a given set of transactions. We analyze the Relay protocol for a set of transactions which are dynamically generated in a given time period, and compare the protocol's time complexity against that of an optimal offline clairvoyant algorithm. We show that Relay is $O(\log D)$-competitive, where D is the diameter of the spanning tree, for a set of transactions that request the same object, given the condition that the maximum local execution time of transactions is sufficiently small.

Runtime Checking of Serializability in Software Transactional Memory

Arnab Sinha and Sharad Malik
Dept. of Electrical Engineering, Princeton University
{sinha,sharad}@princeton.edu

Abstract

Ensuring the correctness of complex implementations of software transactional memory (STM) is a daunting task. Attempts have been made to formally verify STMs, but these are limited in the scale of systems they can handle [1], [2], [3] and generally verify only a model of the system, and not the actual system. In this paper we present an alternate attack on checking the correctness of an STM implementation by verifying the execution runs of an STM using a checker that runs in parallel with the transaction memory system. With future many-core systems predicted to have hundreds and even thousands of cores [4], it is reasonable to utilize some of these cores for ensuring the correctness of the rest of the system. This will be needed anyway given the increasing likelihood of dynamic errors due to particle hits (soft errors) and increasing fragility of nanoscale devices. These errors can only be detected at runtime. An important correctness criterion that is the subject of verification is the serializability of transactions. While checking transaction serializability is NP-complete, practically useful subclasses such as interchange-serializability (DSR) are efficiently computable [5]. Checking DSR reduces to checking for cycles in a transaction ordering graph which captures the access order of objects shared between transaction instances. Doing this concurrent to the main transaction execution requires minimizing the overhead of capturing object accesses, and managing the size of the graph, which can be as large as the total number of dynamic transactions and object accesses. We discuss techniques for minimizing the overhead of access logging which includes time-stamping, and present techniques for on-the-fly graph compaction that drastically reduce the size of the graph that needs to be maintained, to be no larger than the number of threads. We have implemented concurrent serializability checking in the Rochester Software Transactional Memory (RSTM) system [6]. We present our practical experiences with this including results for the RSTM, STAMP [7] and synthetic benchmarks. The overhead of concurrent checking is a strong function of the transaction length. For long transactions this is negligible. Thus the use of the proposed method for continuous runtime checking is acceptable. For very short transactions this can be significant. In this case we see the applicability of the proposed method for debugging.

Consistency in Hindsight: A Fully Decentralized STM Algorithm

Annette Bieniusa
Department of Computer Science
University of Freiburg, Germany
bieniusa@informatik.uni-freiburg.de

Thomas Fuhrmann
Department of Computer Science
Technical University of Munich, Germany
fuhrmann@net.in.tum.de

Abstract

Software transactional memory (STM) algorithms often rely on centralized components to achieve atomicity, isolation and consistency. In a distributed setting, centralized components are undesirable as they impair scalability. This paper presents *Decent STM*, a fully decentralized object-based STM algorithm. It relies on mostly immutable data structures, which are well-suited for replication and migration. It is the first decentralized STM implementing snapshot isolation semantics. A novel randomized consensus protocol guarantees consistency of the mutable parts. Transactions may proceed tentatively before consensus has been reached. Object versioning ensures consistency in hindsight. Thus, atomic code sections never block during execution. The evaluation of benchmarks shows that the guaranteed success of reads more than compensates for the higher conflict rate during commit.

Session 14

Tools for Performance and Correctness Analysis

Identifying Ad-hoc Synchronization for Enhanced Race Detection

Ali Jannesari and Walter F. Tichy
Karlsruhe Institute of Technology (KIT)
76131 Karlsruhe, Germany
Email: {jannesari, tichy}@kit.edu

Abstract

Parallel programs contain a surprising number of ad-hoc synchronization operations. Ad-hoc synchronization operations are loops that busy-wait on condition variables. Current race detectors produce unnecessary warnings (false positives) when ad-hoc synchronization is used. False positives are also generated when programmers use synchronization primitives that are unknown to race detectors, for instance when programmers switch libraries. These shortcomings may result in an overwhelming number of false positives, dissuading programmers from using race detectors.

This paper shows that ad-hoc synchronization operations can be detected automatically. The method requires no user intervention such as annotations and has been implemented in the race detector Helgrind$^+$. Evaluation results on various benchmarks confirm that Helgrind$^+$ is aware of all synchronizations in programs, reliably reports true races, and produces few false alarms. A surprising result is that with the new technique, Helgrind$^+$ can analyze synchronization libraries, so special knowledge about these libraries is not needed in the detector.

Improving the Performance of Program Monitors with Compiler Support in Multi-Core Environment

Guojin He and Antonia Zhai
Department of Computer Science and Engineering
University of Minnesota
Minneapolis, MN 55455, USA
{guojinhe, zhai}@cs.umn.edu

Abstract

Dynamic program execution monitors allow programmers to observe and verify an application while it is running. Instrumentation-based dynamic program monitors often incur significant performance overhead due to instrumentation. Special hardware supports have been proposed to reduce this overhead. However, these supports mostly target specific monitoring requirements and thus have limited applicability. Recently, with multi-core processors becoming mainstream, executing the monitored program and the monitor simultaneously on separate cores has emerged as an attractive option. However, communication between the two often becomes the new performance bottleneck due to large amounts of information forwarded to the monitor.

In this paper, we present compiler techniques that aim to minimize the communication overhead. Our proposal is based on the observations that a monitor only requires specific information from the monitored programs and some information can be easily computed by the monitor from data that have already been communicated. We developed a code generator and optimization techniques to decide the set of data items to forward and the set to compute, so that the total execution time of the monitor is minimized. Our compiler can optimize a variety of monitors with diverse monitoring requirements, taking as input the control flow graph of the monitored program and the set of data that needs verification.

Using a static binary rewriter, we evaluate the performance impact of the proposed compiler techniques on the SPEC2006 integer benchmarks for two intensive monitoring tasks: taint-propagation and memory bug detection. Comparing to instrumentation-based monitors, the proposed techniques can bring down the performance overhead of the two monitors from 10.6x and 9.0x to 2.36x and 2.17x, respectively.

On-line detection of large-scale parallel application's structure

German Llort, Juan Gonzalez, Harald Servat, Judit Gimenez and Jesus Labarta
Barcelona Supercomputing Center
Universitat Politècnica de Catalunya
Barcelona, Spain
Email: {gllort, jgonzale, harald, judit, jesus}@bsc.es

Abstract

With larger and larger systems being constantly deployed, trace-based performance analysis of parallel applications has become a challenging task. Even if the amount of performance data gathered per single process is small, traces rapidly become unmanageable when merging together the information collected from all processes.

In general, an efficient analysis of such a large volume of data is subject to a previous filtering step that directs the analyst's attention towards what is meaningful to understand the observed application behavior. Furthermore, the iterative nature of most scientific applications usually ends up producing repetitive information. Discarding irrelevant data aims at reducing both the size of traces, and the time required to perform the analysis and deliver results.

In this paper, we present an on-line analysis framework that relies on clustering techniques to intelligently select the most relevant information to understand how the application behaves, while keeping the volume of performance data at a reasonable size.

Adaptive Sampling-Based Profiling Techniques for Optimizing the Distributed JVM Runtime

King Tin Lam, Yang Luo, Cho-Li Wang
Department of Computer Science
The University of Hong Kong
Hong Kong
{ktlam, clwang, yluo}@cs.hku.hk

Abstract

Extending the standard Java virtual machine (JVM) for cluster-awareness is a transparent approach to scaling out multi-threaded Java applications. While this clustering solution is gaining momentum in recent years, efficient runtime support for fine-grained object sharing over the distributed JVM remains a challenge. The system efficiency is strongly connected to the global object sharing profile that determines the overall communication cost. Once the sharing or correlation between threads is known, access locality can be optimized by collocating highly correlated threads via dynamic thread migrations. Although correlation tracking techniques have been studied in some page-based software DSM systems, they would entail prohibitively high overheads and low accuracy when ported to fine-grained object-based systems. In this paper, we propose a lightweight sampling-based profiling technique for tracking inter-thread sharing. To preserve locality across migrations, we also propose a stack sampling mechanism for profiling the set of objects which are tightly coupled with a migrant thread. Sampling rates in both techniques can vary adaptively to strike a balance between preciseness and overhead. Such adaptive techniques are particularly useful for applications whose sharing patterns could change dynamically. The profiling results can be exploited for effective thread-to-core placement and dynamic load balancing in a distributed object sharing environment. We present the design and preliminary performance result of our distributed JVM with the profiling implemented. Experimental results show that the profiling is able to obtain over 95% accurate global sharing profiles at a cost of only a few percents of execution time increase for fine- to medium-grained applications.

Session 15
Parallel Linear Algebra I

Algorithmic Cholesky Factorization Fault Recovery

Doug Hakkarinen and Zizhong Chen
Department of Mathematical and Computer Sciences
Colorado School of Mines,
Golden, CO, USA
{dhakkari, zchen}@mines.edu

Abstract

Modeling and analysis of large scale scientific systems often use linear least squares regression, frequently employing Cholesky factorization to solve the resulting set of linear equations. With large matrices, this often will be performed in high performance clusters containing many processors. Assuming a constant failure rate per processor, the probability of a failure occurring during the execution increases linearly with additional processors. Fault tolerant methods attempt to reduce the expected execution time by allowing recovery from failure. This paper presents an analysis and implementation of a fault tolerant Cholesky factorization algorithm that does not require checkpointing for recovery from fail-stop failures. Rather, this algorithm uses redundant data added in an additional set of processors. This differs from previous works with algorithmic methods as it addresses fail-stop failures rather than fail-continue cases. The implementation and experimentation using ScaLAPACK demonstrates that this method has decreasing overhead in relation to overall runtime as the matrix size increases, and thus shows promise to reduce the expected runtime for Cholesky factorizations on very large matrices.

Analyzing the Soft Error Resilience of Linear Solvers on Multicore Multiprocessors

Konrad Malkowski, Padma Raghavan and Mahmut Kandemir
Department of Computer Science and Engineering
The Pennsylvania State University
University Park, PA 16802, USA
{malkowsk, raghavan, kandemir}@cse.psu.edu

Abstract

As chip transistor densities continue to increase, soft errors (bit flips) are becoming a significant concern in networked multiprocessors with multicore nodes. Large cache structures in multicore processors are especially susceptible to soft errors as they occupy a significant portion of the chip area. In this paper, we consider the impacts of soft errors in caches on the resilience and energy efficiency of sparse linear solvers. In particular, we focus on two widely used sparse iterative solvers, namely Conjugate Gradient (CG) and Generalized Minimum Residuals (GMRES). We propose two adaptive schemes, (i) a *Write Eviction Hybrid ECC (WEH-ECC)* scheme for the L1 cache and (ii) a *Prefetcher Based Adaptive ECC (PBA-ECC)* scheme for the L2 cache, and evaluate the energy and reliability trade-offs they bring in the context of GMRES and CG solvers. Our evaluations indicate that *WEH-ECC* reduces the CG and GMRES soft error vulnerability by a factor of 18 to 220 in L1 cache, relative to an unprotected L1 cache, and energy consumption by 16%, relative to a cache with strong protection. The *PBA-ECC* scheme reduces the CG and GMRES soft error vulnerability by a factor of 9×10^3 to 8.6×10^9, relative to an unprotected L2 cache, and reduces the energy consumption by 8.5%, relative to a cache with strong ECC protection. Our energy overheads over unprotected L1 and L2 caches are 5% and 14% respectively.

Parallel Architecture for Meaning Comparison

Suneil Mohan, Amitava Biswas, Aalap Tripathy, Jagannath Pannigrahy and Rabi Mahapatra
Department of Computer Science and Engineering,
Texas A&M University,
College Station, Texas, USA
{suneil,amitabi,aalap,jagannath,rabi}@cse.tamu.edu

Abstract

In this paper we present a fine grained parallel architecture that performs meaning comparison using vector cosine similarity (dot product). Meaning comparison assigns a similarity value to two objects (e.g. text documents) based on how similar their meanings (represented as two vectors) are to each other. The novelty of our design is the fine grained parallelism which is not exploited in available hardware based dot product processor designs and can not be achieved in traditional server class processors like the Intel Xeon. We compare the performance of our design against that of available hardware based dot product processors as well a server class processor using optimum software code performing the same computation. We show that our hardware design can achieve a speedup of 62,000 times compared to an available hardware design and a speedup of 8866 times with 33% (1.5 times) less power consumption, compared to software code running on Intel Xeon processor for 1024 basis vectors. Our design can significantly reduce the amount of servers required for similarity comparison in a distributed search engine. Thus it can enable reduction in energy consumption, investment, operational costs and floor area in search engine data centers. This design can also be deployed for other applications which require fast dot product computation.

Plenary Session
Best Papers

Extreme Scale Computing: Modeling the Impact of System Noise in Multicore Clustered Systems

Seetharami Seelam, Liana Fong and Asser Tantawi
IBM T. J. Watson Research Center,
{sseelam,llfong,tantawi}@us.ibm.com

John Lewars, John Divirgilio and Kevin Gildea
IBM Systems and Technology Group,
{jlewars,jdivirgi,gildeak}@us.ibm.com

Abstract

System noise or Jitter is the activity of hardware, firmware, operating system, runtime system, and management software events. It is shown to disproportionately impact application performance in current generation large-scale clustered systems running general-purpose operating systems (GPOS). Jitter mitigation techniques such as co-scheduling jitter events across operating systems improve application performance but their effectiveness on future petascale systems is unknown. To understand if existing co-scheduling solutions enable scalable petascale performance, we construct two complementary jitter models based on detailed analysis of system noise from the nodes of a large-scale system running a GPOS. We validate these two models using experimental data from a system consisting of 128 GPOS instances with 4096 CPUs. Based on our models, we project a minimum slowdown of 2.1%, 5.9%, and 11.5% for applications executing on a similar one petaflop system running 1024 GPOS instances and having global synchronization operations once every 1000 msec, 100 msec, and 10 msec, respectively. Our projections indicate that additional system noise mitigation techniques are required to contain the impact of jitter on multi-petaflop systems, especially for tightly synchronized applications.

Oblivious Algorithms for Multicores and Network of Processors

Rezaul Alam Chowdhury[1], Francesco Silvestri[2], Brandon Blakeley[1] and Vijaya Ramachandran[1]

[1]Department of Computer Sciences
University of Texas
Austin, TX 78712, USA
{shaikat,blakeley,vlr}@cs.utexas.edu

[2]Department of Information Engineering
University of Padova
Via Gradenigo 6/B, 35131 Padova, Italy
silvest1@dei.unipd.it

Abstract

We address the design of algorithms for multicores that are oblivious to machine parameters. We propose HM, a multicore model consisting of a parallel shared-memory machine with hierarchical multi-level caching, and we introduce a multicore-oblivious (MO) approach to algorithms and schedulers for HM. An MO algorithm is specified with no mention of any machine parameters, such as the number of cores, number of cache levels, cache sizes and block lengths. However, it is equipped with a small set of instructions that can be used to provide hints to the run-time scheduler on how to schedule parallel tasks. We present efficient MO algorithms for several fundamental problems including matrix transposition, FFT, sorting, the Gaussian Elimination Paradigm, list ranking, and connected components. The notion of an MO algorithm is complementary to that of a network-oblivious (NO) algorithm, recently introduced by Bilardi et al. for parallel distributed-memory machines where processors communicate point-to-point. We show that several of our MO algorithms translate into efficient NO algorithms, adding to the body of known efficient NO algorithms.

Analyzing and Adjusting User Runtime Estimates to Improve Job Scheduling on the Blue Gene/P

Wei Tang[1], Narayan Desai[2], Daniel Buettner[3] and Zhiling Lan[1]
[1]Department of Computer Science, Illinois Institute of Technology
Chicago, IL 60616, USA
{wtang6, lan}@iit.edu
[2]Mathematics and Computer Science Division
[3]Argonne Leadership Computing Facility
Argonne National Laboratory, Argonne, IL 60439, USA
[2]desai@mcs.anl.gov
[3]buettner@alcf.anl.gov

Abstract

Backfilling and short-job-first are widely acknowledged enhancements to the simple but popular first-come, first-served job scheduling policy. However, both enhancements depend on user-provided estimates of job runtime, which research has repeatedly shown to be inaccurate. We have investigated the effects of this inaccuracy on backfilling and different queue prioritization policies, determining which part of the scheduling policy is most sensitive. Using these results, we have designed and implemented several estimation-adjusting schemes based on historical data. We have evaluated these schemes using workload traces from the Blue Gene/P system at Argonne National Laboratory. Our experimental results demonstrate that dynamically adjusting job runtime estimates can improve job scheduling performance by up to 20%.

Performance Evaluation of Concurrent Collections on High-Performance Multicore Computing Systems

Aparna Chandramowlishwaran[1], Kathleen Knobe[2] and Richard Vuduc[1]
[1]College of Computing, Georgia Institute of Technology, Atlanta, GA
[2]Software Solutions Group, Intel Corporation, Hudson, MA

Abstract

This paper is the first extensive performance study of a recently proposed parallel programming model, called Concurrent Collections (CnC). In CnC, the programmer expresses her computation in terms of application-specific operations, partially-ordered by semantic scheduling constraints. The CnC model is well-suited to expressing asynchronous-parallel algorithms, so we evaluate CnC using two dense linear algebra algorithms in this style for execution on state-of-the-art multicore systems: (i) a recently proposed asynchronous-parallel Cholesky factorization algorithm, (ii) a novel and non-trivial "higher-level" partly-asynchronous generalized eigensolver for dense symmetric matrices.

Given a well-tuned sequential BLAS, our implementations match or exceed competing multithreaded vendor-tuned codes by up to 2.6×. Our evaluation compares with alternative models, including ScaLAPACK with a shared memory MPI, OpenMP, Cilk++, and PLASMA 2.0, on Intel Harpertown, Nehalem, and AMD Barcelona systems. Looking forward, we identify new opportunities to improve the CnC language and runtime scheduling and execution.

Session 16
P2P Algorithms

A Hybrid Interest Management Mechanism for Peer-to-Peer Networked Virtual Environments

Ke Pan, Wentong Cai, Xueyan Tang, Suiping Zhou and Stephen John Turner
Parallel and Distributed Computing Center
School of Computer Engineering
Nanyang Technological University
Singapore 639798
{pank0001, aswtcai, asxytang, asspzhou, assjturner}@ntu.edu.sg

Abstract

An Interest Management (IM) mechanism eliminates irrelevant status updates transmitted in Networked Virtual Environments (NVE). However, IM itself involves both computation and communication overhead, of which the latter is the focus of this paper. Traditionally, there are area-based and cell-based IM mechanisms. This paper proposes a hybrid IM mechanism for peer-to-peer NVEs, that utilizes the cell-based mechanism to reduce Area-Of-Interest (AOI) updates in the area-based mechanism so as to reduce its communication overhead. To compare the new mechanism with the two traditional approaches, a multiplayer game scenario is simulated. The performance results show that, compared to the traditional mechanisms, the hybrid mechanism reduces the upload bandwidth consumption by more than 25.28 percent, reduces the overhead ratio from more than 67.54 percent to only 25.17 percent, and allows more than 5000 players in the Internet to join the same game with today's network upload bandwidth.

Attack-Resistant Frequency Counting

Bo Wu, Jared Saia
Department of Computer Science
University of New Mexico
Albuquerque, NM 87131-1386, USA
Email: {bowu, saia}@cs.unm.edu

Valerie King
Department of Computer Science
University of Victoria
Victoria, BC, Canada V8W 3P6
Email: val@uvic.ca

Abstract

We present collaborative peer-to-peer algorithms for the problem of approximating frequency counts for popular items distributed across the peers of a large-scale network. Our algorithms are attack-resistant in the sense that they function correctly even in the case where an adaptive and computationally unbounded adversary causes up to a 1/3 fraction of the peers in the network to suffer Byzantine faults. Our algorithms are scalable in the sense that all resource costs are polylogarithmic. Specifically, latency is $O(\log n)$; the number of messages and number of bits sent and received by each peer is $O(\log^2 n)$ per item; and number of neighbors of each peer is $O(\log^2 n)$. Our motivation for addressing this problem is to provide a tool for the following three applications: worm and virus detection; spam detection; and distributed data-mining.

To the best of our knowledge, our algorithms are the first attack-resistant and scalable algorithms for this problem. Moreover, surprisingly, our algorithms seem to be the first attack-resistant algorithms for *any* data mining problem.

Overlays with Preferences: Approximation Algorithms for Matching with Preference Lists

Giorgos Georgiadis
Department of Computer Science and Engineering
Chalmers University of Technology
S-412 96 Göteborg, Sweden
Email: georgiog@chalmers.se
Fax: +46-31-7723663

Marina Papatriantafilou
Department of Computer Science and Engineering
Chalmers University of Technology
S-412 96 Göteborg, Sweden
Email: ptrianta@chalmers.se
Fax: +46-31-7723663

Abstract

A key property of overlay networks, that is going to play an important part in future networking solutions, is the peers' ability to establish connections with other peers based on some suitability metric related to e.g. the node's distance, interests, recommendations, transaction history or available resources. Each node may choose individually an appropriate metric and try to connect or be matched with the available peers that it considers best. When there are no preference cycles among the peers, it has been proven that a stable matching exists, where peers have maximized the individual satisfaction gleaned of their choices. However, no such guarantees are currently being given for the cases where cycles may exist and known methods may not be able to resolve "oscillations" in preference-based connectivity and reach stability. In this work we employ the use of node satisfaction to move beyond classic stable matchings and towards the overlay network context. We present a simple yet powerful distributed algorithm that uses aggregate satisfaction as an optimization metric. The algorithm is a generalization of an approximation one-to-one matching algorithm, into the many-to-many case. We prove that the total satisfaction achieved by our algorithm is a constant factor approximation of the maximum total satisfaction in the network, depending also on the maximum number of possible connections of a peer in the overlay.

Analysis of Durability in Replicated Distributed Storage Systems

Sriram Ramabhadran
Department of Computer Science & Engineering
University of California, San Diego
La Jolla, CA 92093-0404, USA
sriram@cs.ucsd.edu

Joseph Pasquale
Department of Computer Science & Engineering
University of California, San Diego
La Jolla, CA 92093-0404, USA
pasquale@cs.ucsd.edu

Abstract

In this paper, we investigate the roles of replication vs. repair to achieve durability in large-scale distributed storage systems. Specifically, we address the fundamental questions: How does the lifetime of an object depend on the degree of replication and rate of repair, and how is lifetime maximized when there is a constraint on resources? In addition, in real systems, when a node becomes unavailable, there is uncertainty whether this is temporary or permanent; we analyze the use of timeouts as a mechanism to make this determination. Finally, we explore the importance of memory in repair mechanisms, and show that under certain cost conditions, memoryless systems, which are inherently less complex, perform just as well.

Session 17

Parallel Solutions for String and Sequence Problems

Scalable Multi-Pipeline Architecture for High Performance Multi-Pattern String Matching

Weirong Jiang, Yi-Hua E. Yang and Viktor K. Prasanna
Ming Hsieh Department of Electrical Engineering
University of Southern California
Los Angeles, CA 90089, USA
Email: {weirongj, yeyang, prasanna}@usc.edu

Abstract

Multi-pattern string matching remains a major performance bottleneck in network intrusion detection and anti-virus systems for high-speed deep packet inspection (DPI). Although Aho-Corasick deterministic finite automaton (AC-DFA) based solutions produce deterministic throughput and are widely used in today's DPI systems such as Snort and ClamAV, the high memory requirement of AC-DFA (due to the large number of state transitions in AC-DFA) inhibits efficient hardware implementation to achieve high performance. Some recent work has shown that the AC-DFA can be reduced to a character trie that contains only the forward transitions by incorporating pipelined processing. But they have limitations in either handling long patterns or extensions to support multi-character input per clock cycle to achieve high throughput. This paper generalizes the problem and proves formally that a linear pipeline with H stages can remove all cross transitions to the top H levels of a AC-DFA. A novel and scalable pipeline architecture for memory-efficient multi-pattern string matching is then presented. The architecture can be easily extended to support multi-character input per clock cycle by mapping a compressed AC-DFA onto multiple pipelines. Simulation using Snort and ClamAV pattern sets shows that a 8-stage pipeline can remove more than 99% of the transitions in the original AC-DFA. The implementation on a state-of-the-art field programmable gate array (FPGA) shows that our architecture can store on a single FPGA device the full set of string patterns from the latest Snort rule set. Our FPGA implementation sustains 10+ Gbps throughput, while consuming a small amount of on-chip logic resources. Also desirable scalability is achieved: the increase in resource requirement of our solution is sub-linear with the throughput improvement.

Head-Body Partitioned String Matching for Deep Packet Inspection with Scalable and Attack-Resilient Performance

Yi-Hua E. Yang
Ming Hsieh Dept. of Electrical Eng.
University of Southern California
Email: yeyang@usc.edu

Viktor K. Prasanna
Ming Hsieh Dept. of Electrical Eng.
University of Southern California
Email: prasanna@usc.edu

Chenqian Jiang
Ming Hsieh Dept. of Electrical Eng.
University of Southern California
Email: chenqiaj@usc.edu

Abstract

Dictionary-based string matching (DBSM) is a critical component of Deep Packet Inspection (DPI), where thousands of malicious patterns are matched against high-bandwidth network traffic. Deterministic finite automata constructed with the Aho-Corasick algorithm (AC-DFA) have been widely used for solving this problem. However, the state transition table (STT) of a large-scale DBSM AC-DFA can span hundreds of megabytes of system memory, whose limited bandwidth and long latency could become the performance bottleneck We propose a novel partitioning algorithm which converts an AC-DFA into a "head" and a "body" parts. The head part behaves as a traditional ACDFA that matches the pattern prefixes up to a predefined length; the body part extends any head match to the full pattern length in parallel body-tree traversals. Taking advantage of the SIMD instructions in modern x86-64 multi-core processors, we design compact and efficient data structures packing multi-path and multi-stride pattern segments in the body-tree. Compared with an optimized AC-DFA solution, our head-body matching (HBM) implementation achieves 1.2x to 3x throughput performance when the input match (attack) ratio varies from 2% to 32%, respectively. Our HBM data structure is over 20x smaller than a fully-populated AC-DFA for both Snort and ClamAV dictionaries. The aggregated throughput of our HBM approach scales almost 7x with 8 threads to over 10 Gbps in a dual-socket quad-core Opteron (Shanghai) server.

Parallel de novo Assembly of Large Genomes from High-Throughput Short Reads

B.G. Jackson[1], M. Regennitter[1], X. Yang[1], P.S. Schnable[2] and S. Aluru[1]
[1]Department of Electrical and Computer Engineering
[2]Department of Agronomy
Iowa State University, Ames, IA, USA

Abstract

The advent of high-throughput short read technology is revolutionizing life sciences by providing an inexpensive way to sequence genomes at high coverage. Exploiting this technology requires the development of a *de novo* short read assembler, which is an important open problem that is garnering significant research effort. Current methods are largely limited to microbial organisms, whose genomes are two to three orders of magnitude smaller than complex mammalian and plant genomes. In this paper, we present the design and development of a parallel *de novo* short read assembler that can scale to large genomes with high coverage. Our approach is based on the string graph formulation. Input reads are mapped to short paths, and the genome is reconstructed as a superpath anchored by distance constraints inferred from read pairs. Our method can handle a mixture of multiple read sizes and multiple paired read distances. We present parallel algorithms for string graph construction, string graph compaction, graph based error detection and removal, and computing aggregate summarization of paired read links across graph edges. Using this, we navigate the final graph structure to reproduce large contiguous sequences from the underlying genome. We present a validation of our framework on experimental and simulated data from multiple known genomes and present scaling results on IBM Blue Gene/L.

Efficient Parallel Algorithms for Maximum-Density Segment Problem

Xue Wang, Fasheng Qiu and Sushil K. Prasad
Computer Science
Georgia State University
Atlanta, USA
{xwang14, fqiu1}@student.gsu.edu, sprasad@gsu.edu

Guantao Chen
Mathematics and Statistics
Georgia State University
Atlanta, USA
gchen@gsu.edu

Abstract

One of the fundamental problems involving DNA sequences is to find high density segments of certain widths, for example, those regions with intensive guanine and cytosine (GC). Formally, given a sequence, each element of which has a value and a width, the maximum-density segment problem asks for the segment with the maximum density while satisfying minimum and possibly maximum width constraints. While several linear-time sequential algorithms have emerged recently due to its primitive-like utility, to our knowledge, no nontrivial parallel algorithm has yet been proposed for this topical problem. In this paper, we propose an $O(\log^2 n)$-time CREW PRAM algorithm using n processors to solve the generalized maximum-density problem, with a minimum width constraint and non-uniform widths. Besides, we describe an efficient implementation of the parallel algorithm on manycore GPUs (nVIDIA GeForce GTX 280), taking advantage of the full programmability of CUDA. This algorithm can process up to million-size sequence within a second using an nVIDIA GeForce GTX 280, thus demonstrating the practicality of this algorithm as a basic primitive for scientists. This may also indicate suitability of modern GPU architectures as implementation platform for certain PRAM algorithms.

Session 18
Energy-aware Task Management

Hybrid MPI/OpenMP Power-Aware Computing

Dong Li[1], Bronis R. de Supinski[2], Martin Schulz[2], Kirk Cameron[1] and Dimitrios S. Nikolopoulos[3]

[1]Virginia Tech
Blacksburg, VA, USA
{lid,cameron}@cs.vt.edu

[2]Lawrence Livermore National Lab
Livermore, CA, USA
{bronis,schulzm}@llnl.gov

[3]FORTH-ICS and University of Crete
Heraklion, Crete, GREECE
dsn@ics.forth.gr

Abstract

Power-aware execution of parallel programs is now a primary concern in large-scale HPC environments. Prior research in this area has explored models and algorithms based on dynamic voltage and frequency scaling (DVFS) and dynamic concurrency throttling (DCT) to achieve power-aware execution of programs written in a single programming model, typically MPI or OpenMP. However, hybrid programming models combining MPI and OpenMP are growing in popularity as emerging large-scale systems have many nodes with several processors per node and multiple cores per processor. In this paper we present and evaluate solutions for power-efficient execution of programs written in this hybrid model targeting large-scale distributed systems with multicore nodes. We use a new power-aware performance prediction model of hybrid MPI/OpenMP applications to derive a novel algorithm for power-efficient execution of realistic applications from the ASC Sequoia and NPB MZ benchmarks. Our new algorithm yields substantial energy savings (4.18% on average and up to 13.8%) with either negligible performance loss or performance gain (up to 7.2%).

Performance and Energy Optimization of Concurrent Pipelined Applications

Anne Benoit, Paul Renaud-Goud and Yves Robert
LIP, Ecole Normale Supérieure de Lyon, 46 Allée d'Italie, 69364 Lyon Cedex 07, France
Université de Lyon - UMR 5668 - CNRS - ENSL - UCBL - INRIA

Abstract

In this paper, we study the problem of finding optimal mappings for several independent but concurrent workflow applications, in order to optimize performance-related criteria together with energy consumption. Each application consists in a linear chain graph with several stages, and processes successive data sets in pipeline mode, from the first to the last stage. We study the problem complexity on different target execution platforms, ranking from fully homogeneous platforms to fully heterogeneous ones. The goal is to select an execution speed for each processor, and then to assign stages to processors, with the aim of optimizing several concurrent optimization criteria. There is a clear trade-off to reach, since running faster and/or more processors leads to better performance, but the energy consumption is then very high. Energy savings can be achieved at the price of a lower performance, by reducing processor speeds or enrolling fewer resources. We consider two mapping strategies: in one-to-one mappings, a processor is assigned a single stage, while in interval mappings, a processor may process an interval of consecutive stages of the same application. For both mapping strategies and all platform types, we establish the complexity of several multi-criteria optimization problems, whose objective functions combine period, latency and energy criteria. In particular, we exhibit cases where the problem is NP-hard with concurrent applications, while it can be solved in polynomial time for a single application. Also, we demonstrate the difficulty of performance/energy trade-offs by proving that the tri-criteria problem is NP-hard, even with a single application on a fully homogeneous platform.

Robust Control-theoretic Thermal Balancing for Server Clusters

Yong Fu[1], Chenyang Lu[1] and Hongan Wang[2]
[1]Deptartment of Computer Science and Engineering, Washington University {fuy, lu}@cse.wustl.edu
[2]Institute of Software, Chinese Academy of Sciences wha@iel.iscas.ac.cn

Abstract

Thermal management is critical for clusters because of the increasing power consumption of modern processors, compact server architectures and growing server density in data centers. Thermal balancing mitigates hot spots in a cluster through dynamic load distribution among servers. This paper presents two *Control-theoretical Thermal Balancing (CTB)* algorithms that dynamically balance the temperatures of different servers based on online measurements. CTB features controllers rigorously designed based on optimal control theory and a difference equation model that approximates the thermal dynamics of clusters. Control analysis and simulation results demonstrate that CTB achieves robust thermal balancing under a wide range of uncertainties: (1) when different tasks incur different power consumptions on the CPUs, (2) when servers experience different ambient temperatures, and (3) when servers experience thermal faults.

A Simple Thermal Model for Multi-core Processors and Its Application to Slack Allocation

Zhe Wang, Sanjay Ranka
Dept. of Computer and Information Science and Engineering
University of Florida
Gainesville, USA
Email: {zhwang, sanjay}@cise.ufl.edu

Abstract

Power density and heat density of multicore processor system are increasing exponentially with Moore's Law. High temperature on chip greatly affects its reliability, and the cost of packaging and cooling system increases exponentially with power consumption. For a multicore processor, the peak temperature of a block depends on its own power density as well as power density of other blocks on chip.

In this paper, we have developed a simple thermal model, called Matrix Model (MM), that can be used to derive temperature profiles for all the cores of a multicore processor. We theoretically demonstrate the correctness and efficiency of MM. Our simulation results show that the model is comparable to the HotSpot Model for predicting the peak temperature. Besides having lower computational cost, the MM is succinct (a single matrix) and can be used to derive algorithms for a variety of scenarios. We use this model to develop a novel slack allocation algorithm for a workflow represented by Directed Acyclic Graph on a multicore processor.

Session 19

Parallel Operating Systems and System Software

GenerOS: An Asymmetric Operating System Kernel for Multi-core Systems

Qingbo Yuan, Jianbo Zhao, Mingyu Chen and Ninghui Sun
Key Laboratory of Computer System and Architecture
Institute of Compute Technology, Chinese Academy of Sciences
Beijing, China
{yuanbor, zhaojianbo, cmy, snh}@ncic.ac.cn

Abstract

Due to complex abstractions implemented over shared data structures protected by locks, conventional symmetric multi-threaded operating system kernel such as Linux is hard to achieve high scalability on the emerging multi-core architectures, which integrate more and more cores on a single die. This paper presents GenerOS - a general asymmetric operating system kernel for multi-core systems. In principal, GenerOS partitions processing cores into application core, kernel core and interrupt core, each of which is dedicated to a specified function. In implementation, we conduct a delicate modification to Linux kernel and provide the same interface as Linux kernel so that GenerOS is compatible with legacy applications. The better performance of GenerOS mainly benefits from: (1) Applications run on their own cores with minimal interrupt and kernel support; (2) Every kernel service is encapsulated into a serial process so that there will be fewer contentions than conventional symmetric kernel; (3) A slim schedule policy is used in the kernel core to support schedule between system calls with low overhead.

Experiments with two typical workloads on 16-core AMD machine show that GenerOS behaves better than original Linux kernel when there are more processing cores (19.6% for TPCH using oracle database management system and 42.8% for httperf using apache web server).

Palacios and Kitten: New High Performance Operating Systems For Scalable Virtualized and Native Supercomputing

John Lange[1], Kevin Pedretti[2], Trammell Hudson[2], Peter Dinda[1], Zheng Cui[3], Lei Xia[1], Patrick Bridges[3],
Andy Gocke[1], Steven Jaconette[1], Mike Levenhagen[2] and Ron Brightwell[2]
[1]Northwestern University, Department of Electrical Engineering and Computer Science
Email: {jarusl,pdinda,leixia,agocke,jaconette}@northwestern.edu
[2]Sandia National Laboratories, Scalable System Software Department
Email: {ktpedre,mjleven,rbbrigh}@sandia.gov, hudson@osresarch.net
[3]University of New Mexico, Department of Computer Science
Email: {zheng,bridges}@cs.unm.edu

Abstract

Palacios is a new open-source VMM under development at Northwestern University and the University of New Mexico that enables applications executing in a virtualized environment to achieve scalable high performance on large machines. Palacios functions as a modularized extension to Kitten, a high performance operating system being developed at Sandia National Laboratories to support large-scale supercomputing applications. Together, Palacios and Kitten provide a thin layer over the hardware to support full-featured virtualized environments alongside Kitten's lightweight native environment. Palacios supports existing, unmodified applications and operating systems by using the hardware virtualization technologies in recent AMD and Intel processors. Additionally, Palacios leverages Kitten's simple memory management scheme to enable low-overhead pass-through of native devices to a virtualized environment. We describe the design, implementation, and integration of Palacios and Kitten. Our benchmarks show that Palacios provides near native (within 5%), scalable performance for virtualized environments running important parallel applications. This new architecture provides an incremental path for applications to use supercomputers, running specialized lightweight host operating systems, that is not significantly performance-compromised.

MMT: Exploiting Fine-Grained Parallelism in Dynamic Memory Management

Devesh Tiwari, Sanghoon Lee, James Tuck and Yan Solihin
Department of Electrical and Computer Engineering
North Carolina State University
Raleigh, USA
{devesh.dtiwari,shlee5,jtuck,solihin}@ncsu.edu

Abstract

Dynamic memory management is one of the most expensive but ubiquitous operations in many C/C++ applications. Additional features such as security checks, while desirable, further worsen memory management overheads. With advent of multicore architecture, it is important to investigate how dynamic memory management overheads for sequential applications can be reduced. In this paper, we propose a new approach for accelerating dynamic memory management on multicore architecture, by offloading dynamic management functions to a separate thread that we refer to as memory management thread (MMT).We show that an efficient MMT design can give significant performance improvement by extracting parallelism while being agnostic to the underlying memory management library algorithms and data structures. We also show how parallelism provided by MMT can be beneficial for high overhead memory management tasks, for example, security checks related to memory management. We evaluate MMT on heap allocation-intensive benchmarks running on an Intel core 2 quad platform for two widely-used memory allocators: Doug Lea's and PHKmalloc allocators. On average, MMT achieves a speedup ratio of 1.19× for both allocators, while both the application and memory management libraries are unmodified and are oblivious to the parallelization scheme. For PHKmalloc with security checks turned on, MMT reduces the security check overheads from 21% to just 1% on average.

Optimization of Applications with Non-blocking Neighborhood Collectives via Multisends on the Blue Gene/P Supercomputer

Sameer Kumar[1] Philip Heidelberger[1] Dong Chen[1] Michael Hines[2]
{sameerk, philiph, chendong}@us.ibm.com
[1]IBM T.J. Watson Research Center Yorktown Heights, NY, 10598
michael.hines@yale.edu
[1]Department of Computer Science, Yale University, New Haven, CT, USA

Abstract

We explore the multisend interface as a data mover interface to optimize applications with neighborhood collective communication operations. One of the limitations of the current MPI 2.1 standard is that the vector collective calls require counts and displacements (zero and non-zero bytes) to be specified for all the processors in the communicator. Further, all the collective calls in MPI 2.1 are blocking and do not permit overlap of communication with computation in the same thread of execution. However, multisends are non-blocking calls that permit overlap of computation and communication. We present the record replay persistent optimization to the multisend interface that minimizes the processor overhead of initiating the collective. We present four different case studies with the multisend API on Blue Gene/P (i) 3D-FFT, (ii) 4D nearest neighbor exchange as used in Quantum Chromodynamics, (iii) NAMD and (iv) neural network simulator NEURON. Performance results show 1.9x speedup with 32^3 3D-FFTs, 1.9x speedup for 4D nearest neighbor exchange with the 2^4 problem, 1.6x speedup in NAMD and almost 3x speedup in NEURON with 256K cells and 1k connections/cell.

Session 20
Parallel Graph Algorithms I

A Multi-Source Label-Correcting Algorithm for the All-Pairs Shortest Paths Problem

Hiroki Yanagisawa
IBM Research - Tokyo
IBM Japan, Ltd.
Kanagawa, Japan
Email: yanagis@jp.ibm.com

Abstract

The All-Pairs Shortest Paths (APSP) problem seeks the shortest path distances between all pairs of vertices, and is one of the most fundamental graph problems. In this paper, a fast algorithm with a small working space for the APSP problem on sparse graphs is presented, which first divides the vertices into sets of vertices with each set having a constant number of vertices and then solves the multi-source shortest paths (MSSP) problem for each set in parallel. For solving the MSSP problems, we give a multi-source label-correcting algorithm, as an extension of a label-correcting algorithm for the single-source shortest path problem. Our algorithm uses fewer operations on the priority queue than an implementation based on Dijkstra's algorithm. Our experiments showed that an implementation of our algorithm with SIMD instructions achieves an order of magnitude speedup for real-world geometric graphs compared to an implementation based on Dijkstra's algorithm.

Parallel Computation of Best Connections in Public Transportation Networks

Daniel Delling
Microsoft Research Silicon Valley,
1065 La Avenida, Mountain View, CA 94043.
dadellin@microsoft.com

Bastian Katz
Department of Computer Science,
Karlsruhe Institute of Technology,
76128 Karlsruhe, Germany.
katz@kit.edu

Thomas Pajor
Department of Computer Science,
Karlsruhe Institute of Technology,
76128 Karlsruhe, Germany.
pajor@kit.edu

Abstract

Exploiting parallelism in route planning algorithms is a challenging algorithmic problem with obvious applications in mobile navigation and timetable information systems. In this work, we present a novel algorithm for the so-called one-to-all *profile-search* problem in public transportation networks. It answers the question for all fastest connections between a given station S and any other station at any time of the day in a single query. This algorithm allows for a very natural parallelization, yielding excellent speed-ups on standard multi-core servers. Our approach exploits the facts that first, time-dependent travel-time functions in such networks can be represented as a special class of piecewise linear functions, and that second, only few connections from S are useful to travel far away. Introducing the *connection-setting* property, we are able to extend DIJKSTRA's algorithm in a sound manner. Furthermore, we also accelerate station-to-station queries by preprocessing important connections within the public transportation network. As a result, we are able to compute all relevant connections between two random stations in a complete public transportation network of a big city (Los Angeles) on a standard multi-core server in less than 55 ms on average.

Dynamically Tuned Push-Relabel Algorithm for the Maximum Flow Problem on CPU-GPU-Hybrid Platforms

Zhengyu He and Bo Hong
School of Electrical and Computer Engineering
Georgia Institute of Technology
zhengyu.he,bohong@gatech.edu

Abstract

The maximum flow problem is a fundamental graph theory problem with many important applications. Maxflow algorithms based on the push-relabel method are known to have better complexity bound and faster practical execution speed than others. However, existing push-relabel algorithms are designed for uniprocessors or parallel processors that support locking primitives, thus making it very difficult to apply the push-relabel technique to CUDA-based GPUs. In this paper, we present a first generic parallel push-relabel algorithm for CUDA devices. We model the parallelization efficiency of the algorithm, which reveals that, for a given input graph, the level of parallelism varies during the execution of the algorithm. To maximize the execution efficiency, we develop a dynamically tuned algorithm that utilizes both CPU and GPU by adaptively switching between the two computing units during run time. We show that algorithm finds the maximum flow with $O(|V|^2|E|)$ operations (summed over both the CPU and the GPU). Extensive experimental results show that the new algorithm is up to 2 times faster than the push-relabel algorithm by Goldberg et al.

A Novel Application of Parallel Betweenness Centrality to Power Grid Contingency Analysis

Shuangshuang Jin, Zhenyu Huang, Yousu Chen, Daniel Chavarría-Miranda,
John Feo and Pak Chung Wong
Pacific Northwest National Laboratory
Richland WA 99352, USA
Email: {shuangshuang.jin, zhenyu.huang, yousu.chen, daniel.chavarria, john.feo, pak.wong}@pnl.gov

Abstract

In Energy Management Systems, contingency analysis is commonly performed for identifying and mitigating potentially harmful power grid component failures. The exponentially increasing combinatorial number of failure modes imposes a significant computational burden for massive contingency analysis. It is critical to select a limited set of high-impact contingency cases within the constraint of computing power and time requirements to make it possible for real-time power system vulnerability assessment. In this paper, we present a novel application of parallel betweenness centrality to power grid contingency selection. We cross-validate the proposed method using the model and data of the western US power grid, and implement it on a Cray XMT system - a massively multithreaded architecture - leveraging its advantages for parallel execution of irregular algorithms, such as graph analysis. We achieve a speedup of 55 times (on 64 processors) compared against the single-processor version of the same code running on the Cray XMT. We also compare an OpenMP-based version of the same code running on an HP Superdome shared-memory machine. The performance of the Cray XMT code shows better scalability and resource utilization, and shorter execution time for large-scale power grids. This proposed approach has been evaluated in PNNL's Electricity Infrastructure Operations Center (EIOC). It is expected to provide a quick and efficient solution to massive contingency selection problems to help power grid operators to identify and mitigate potential widespread cascading power grid failures in real time.

Session 21
Parallel Linear Algebra II

Adapting Communication-Avoiding LU and QR Factorizations to Multicore Architectures

Simplice DONFACK
INRIA Saclay-Ile de France
Bat 490,Universite Paris-Sud 11
91405 Orsay France
simplice.donfack@lri.fr

Laura GRIGORI
INRIA Saclay-Ile de France
Bat 490,Universite Paris-Sud 11
91405 Orsay France
Laura.Grigori@inria.fr

Alok KUMAR GUPTA
BCCS, Bergen Norway-5075
alok.gupta@bccs.uib.no

Abstract

In this paper we study algorithms for performing the LU and QR factorizations of dense matrices. Recently, two communication optimal algorithms have been introduced for distributed memory architectures, referred to as communication avoiding CALU and CAQR. In this paper we discuss two algorithms based on CAQR and CALU that are adapted to multicore architectures. They combine ideas to reduce communication from communication avoiding algorithms with asynchronism and dynamic task scheduling. For matrices that are tall and skinny, that is, they have many more rows than columns, the two algorithms outperform the corresponding algorithms from Intel MKL vendor library on a dual-socket, quad-core machine based on Intel Xeon EMT64 processor and on a four-socket, quad-core machine based on AMD Opteron processor. For these matrices, multithreaded CALU outperforms the corresponding routine dgetrf from Intel MKL library up to a factor of 2.3 and the corresponding routine dgetrf from ACML library up to a factor of 5, while multithreaded CAQR outperforms by a factor of 5.3 the corresponding dgeqrf routine from MKL library.

QR Factorization of Tall and Skinny Matrices in a Grid Computing Environment

Emmanuel Agullo[1], Camille Coti[2], Jack Dongarra[1], Thomas Herault [3] and Julien Langou [4]
[1] Dpt of Electrical Engineering and Computer Science, University of Tennessee
[2]INRIA Saclay-Île de France, F-91893 Orsay, France
[3]Univ Paris Sud, University of Tennesse, LRI, INRIA
[4]Dpt of Mathematical and Statistical Sciences, University of Colorado Denver
eagullo@eecs.utk.edu, coti@lri.fr, dongarra@eecs.utk.edu,
herault@lri.fr, julien.langou@ucdenver.edu

Abstract

Previous studies have reported that common dense linear algebra operations do not achieve speed up by using multiple geographical sites of a computational grid. Because such operations are the building blocks of most scientific applications, conventional supercomputers are still strongly predominant in high-performance computing and the use of grids for speeding up large-scale scientific problems is limited to applications exhibiting parallelism at a higher level. We have identified two performance bottlenecks in the distributed memory algorithms implemented in ScaLAPACK, a state-of-the-art dense linear algebra library. First, because ScaLAPACK assumes a homogeneous communication network, the implementations of ScaLAPACK algorithms lack locality in their communication pattern. Second, the number of messages sent in the ScaLAPACK algorithms is significantly greater than other algorithms that trade flops for communication. In this paper, we present a new approach for computing a QR factorization C one of the main dense linear algebra kernels C of tall and skinny matrices in a grid computing environment that overcomes these two bottlenecks. Our contribution is to articulate a recently proposed algorithm (Communication Avoiding QR) with a topology-aware middleware (QCG-OMPI) in order to confine intensive communications (ScaLAPACK calls) within the different geographical sites. An experimental study conducted on the Grid'5000 platform shows that the resulting performance increases linearly with the number of geographical sites on large-scale problems (and is in particular consistently higher than ScaLAPACK's).

Tile QR Factorization with Parallel Panel Processing for Multicore Architectures

Bilel Hadri[1], Hatem Ltaief[1], Emmanuel Agullo[1] and Jack Dongarra[1,2,3]
[1]Department of Electrical Engineering and Computer Science
University of Tennessee, Knoxville,Tennessee, USA
Email: {hadri, ltaief, eagullo, dongarra}@eecs.utk.edu
[2]Computer Science and Mathematics Division
Oak Ridge National Laboratory
Oak Ridge, Tennessee, USA
[3]School of Mathematics & School of Computer Science
University of Manchester,England

Abstract

To exploit the potential of multicore architectures, recent dense linear algebra libraries have used tile algorithms, which consist in scheduling a Directed Acyclic Graph (DAG) of tasks of fine granularity where nodes represent tasks, either panel factorization or update of a block-column, and edges represent dependencies among them. Although past approaches already achieve high performance on moderate and large square matrices, their way of processing a panel in sequence leads to limited performance when factorizing tall and skinny matrices or small square matrices. We present a new fully asynchronous method for computing a QR factorization on shared-memory multicore architectures that overcomes this bottleneck. Our contribution is to adapt an existing algorithm that performs a panel factorization in parallel (named Communication-Avoiding QR and initially designed for distributed-memory machines), to the context of tile algorithms using asynchronous computations. An experimental study shows significant improvement (up to almost 10 times faster) compared to state-of-the-art approaches. We aim to eventually incorporate this work into the Parallel Linear Algebra for Scalable Multi-core Architectures (PLASMA) library.

Linpack Evaluation on a Supercomputer with Heterogeneous Accelerators

Toshio Endo
Graduate School of Information Science and Engineering
Tokyo Institute of Technology
Tokyo, Japan
endo@is.titech.ac.jp

Akira Nukada
Global Scientific Information and Computing Center
Tokyo Institute of Technology
Tokyo, Japan
nukada@matsulab.is.titech.ac.jp

Satoshi Matsuoka
Global Scientific Information and Computing Center
Tokyo Institute of Technology
/National Institute of Informatics
Tokyo, Japan
matsu@is.titech.ac.jp

Naoya Maruyama
Global Scientific Information and Computing Center
Tokyo Institute of Technology
Tokyo, Japan
naoya@matsulab.is.titech.ac.jp

Abstract

We report Linpack benchmark results on the TSUBAME supercomputer, a large scale heterogeneous system equipped with NVIDIA Tesla GPUs and ClearSpeed SIMD accelerators. With all of 10,480 Opteron cores, 640 Xeon cores, 648 ClearSpeed accelerators and 624 NVIDIA Tesla GPUs, we have achieved 87.01TFlops, which is the third record as a heterogeneous system in the world. This paper describes careful tuning and load balancing method required to achieve this performance. On the other hand, since the peak speed is 163 TFlops, the efficiency is 53%, which is lower than other systems. This paper also analyses this gap from the aspect of system architecture.

Session 22
Caches and Caching

Adapting Cache Partitioning Algorithms to Pseudo-LRU Replacement Policies

Kamil Kędzierski, Miquel Moreto
Technical University of Catalonia (UPC)
and Barcelona Supercomputing Center (BSC)
Barcelona, Spain
{kkedzier, mmoreto}@ac.upc.edu

Francisco J. Cazorla
Spanish National Research
Council (IIIA-CSIC) and BSC
Barcelona, Spain
francisco.cazorla@bsc.es

Mateo Valero
UPC
and BSC
Barcelona, Spain
mateo@ac.upc.edu

Abstract

Recent studies have shown that cache partitioning is an efficient technique to improve throughput, fairness and Quality of Service (QoS) in CMP processors. The cache partitioning algorithms proposed so far assume Least Recently Used (LRU) as the underlying replacement policy. However, it has been shown that the true LRU imposes extraordinary complexity and area overheads when implemented on high associativity caches, such as last level caches. As a consequence, current processors available on the market use pseudo-LRU replacement policies, which provide similar behavior as LRU, while reducing the hardware complexity. Thus, the presented so far LRU-based cache partitioning solutions cannot be applied to real CMP architectures.

This paper proposes a complete partitioning system for caches using the pseudo-LRU replacement policy. In particular, the paper focuses on the pseudo-LRU implementations proposed by Sun Microsystems and IBM, called Not Recently Used (NRU) and Binary Tree (BT), respectively. We propose a high accuracy profiling logic and a cache partitioning hardware for both schemes. We evaluate our proposals' hardware costs in terms of area and power, and compare them against the LRU partitioning algorithm.

Overall, this paper presents two hardware techniques to adapt the existing cache partitioning algorithms to real replacement policies. The results show that our solutions impose negligible performance degradation with respect to the LRU.

Exploiting Set-Level Non-Uniformity of Capacity Demand to Enhance CMP Cooperative Caching

Dongyuan Zhan, Hong Jiang and Sharad C. Seth
Department of Computer Science & Engineering,
University of Nebraska C Lincoln, Lincoln, NE 68588
{dzhan, jiang, seth}@cse.unl.edu

Abstract

As the Memory Wall remains a bottleneck for Chip Multiprocessors (CMP), the effective management of CMP last level caches becomes of paramount importance in minimizing expensive offchip memory accesses. For the CMPs with private last level caches, Cooperative Caching (CC) has been proposed to enable capacity sharing among private caches by spilling an evicted block from one cache to another. But this eviction-driven CC does not necessarily promote the cache performance since it implicitly favors the applications full of block evictions regardless of their real capacity demand. The recent Dynamic Spill-Receive (DSR) paradigm improves CC by prioritizing applications with higher benefit from extra capacity in spilling blocks. However, the DSR paradigm only exploits the coarse-grained application-level difference in capacity demand, making it less effective as the non-uniformity exists at a much finer level.

This paper (i) highlights the observation of cache set-level nonuniformity of capacity demand, and (ii) presents a novel L2 cache design, named SNUG (Set-level Non-Uniformity identifier and Grouper), to exploit the fine-grained non-uniformity to further enhance the effectiveness of cooperative caching. By utilizing a per-set shadow tag array and saturating counter, SNUG can identify whether a set should either spill or receive blocks; by using an index-bit flipping scheme, SNUG can group peer sets for spilling and receiving in an flexible way, capturing more opportunities for cooperative caching. We evaluate our design through extensive execution-driven simulations on Quad-core CMP systems. Our results show that for 6 classes of workload combinations our SNUG cache can improve the CMP throughput by up to 22.3%, with an average of 13.9% over the baseline configuration, while the state-of-the-art DSR scheme can only achieve an improvement by up to 14.5% and 8.4% on average.

Masking I/O Latency using Application Level I/O Caching and Prefetching on Blue Gene Systems

Seetharami Seelam, I-Hsin Chung, John Bauer and Hui-Fang Wen
IBM Thomas J. Watson Research Center
Yorktown Heights, New York 10598 USA
{sseelam,ihchung,bauerj,hfwen}@us.ibm.com

Abstract

In this paper, we present an application-level I/O caching, prefetching, asynchronous system to hide access latency experienced by HPC applications. Our solution of user controllable caching and prefetching system maintains a file-IO cache in the user space of the application, analyzes the I/O access patterns, prefetches requests, and performs write-back of dirty data to storage asynchronously. So each time the application needs the data it does not have to pay the full I/O latency penalty in going to the storage and getting the required data.

We have implemented this caching and asynchronous access system on the Blue Gene (BG/L and BG/P) systems. We present experimental results with NAS BT, MADbench, and WRF benchmarks. The results on BG/P system demonstrate that our method hides access latency, enhances application I/O access time by as much as 100%, and improves WRF execution time over 10%.

Intra-Application Cache Partitioning

Sai Prashanth Muralidhara, Mahmut Kandemir and Padma Raghavan
Department of Computer Science and Engineering
Pennsylvania State University, University Park, PA 16802, USA
{smuralid, kandemir, raghavan}@cse.psu.edu

Abstract

Efficient management of shared on-chip resources such as the shared level 2 (L2) cache has become an important problem with the emergence of chip multiprocessors (CMPs). Partitioning the shared cache in chip multiprocessors (CMPs) among concurrently executing applications can provide important benefits such as throughput improvement, fairness guarantees, and quality of service (QoS) enhancements. In this paper, we pose an interesting related question, which is, if partitioning the shared cache space among concurrently executing threads of the same application can enhance the application performance. We address this problem by identifying and speeding up the slowest thread, also termed as the critical path thread, during each execution interval since the overall performance of a multithreaded application is determined by the critical path thread. To do so, we propose a dynamic, runtime system based, cache partitioning scheme that partitions the shared cache space dynamically among the individual threads of a given application. In a nutshell, we wish to take some cache space away from the faster threads and give it to the critical path thread at each execution interval. We show that speeding up the critical path thread this way, results in overall performance enhancement of the application execution in the long term. Our experimental evaluation indicates that, the proposed dynamic cache partitioning scheme yields benefits up to 15% over a shared cache with no partitions, up to 23% over a statically partitioned cache (private cache) and up to 20% over a throughput-oriented scheme.

Session 23
Thread Scheduling

SLAW: a Scalable Locality-aware Adaptive Work-stealing Scheduler

Yi Guo, Jisheng Zhao, Vincent Cave and Vivek Sarkar
Department of Computer Science, Rice University, Houston, Texas 77005
Email: {yguo,jz10,vc8,vsarkar}@rice.edu

Abstract

This paper introduces SLAW, a Scalable Locality-aware Adaptive Work-stealing scheduler. The SLAW scheduler is designed to address two common limitations in current *work-stealing* schedulers: use of a fixed task scheduling policy and locality-obliviousness due to randomized stealing.

Past work has demonstrated the pros and cons of using fixed scheduling policies, such as *work-first* and *help-first*, in different cases without a clear win for one policy over the other. The SLAW scheduler addresses this limitation by supporting both work-first and help-first policies simultaneously. It does so by using an *adaptive* approach that selects a scheduling policy on a per-task basis at runtime. The SLAW scheduler also establishes bounds on the stack and heap space needed to store tasks. The experimental results for the benchmarks studied in this paper show that SLAW's adaptive scheduler achieves 0.98× to 9.2× speedup over the help-first scheduler and 0.97× to 4.5× speedup over the work-first scheduler for 64-thread executions, thereby establishing the robustness of using an adaptive approach instead of a fixed policy. In contrast, the help-first policy is 9.2× slower than work-first in the worst case for a fixed help-first policy, and the work-first policy is 3.7× slower than help-first in the worst case for a fixed work-first policy. Further, for large irregular recursive parallel computations, the adaptive scheduler runs with bounded stack usage and achieves performance (and supports data sizes) that cannot be delivered by the use of any single fixed policy.

It is also known that work-stealing schedulers can be cache-unfriendly for some applications due to randomized stealing. The SLAW scheduler is designed for programming models where locality hints are provided to the runtime by the programmer or compiler, and achieves *locality-awareness* by grouping workers into *places*. Locality awareness can lead to improved performance by increasing temporal data reuse within a worker and among workers in the same place. Our experimental results show that locality-aware scheduling can achieve up to 2.6× speedup over locality-oblivious scheduling, for the benchmarks studied in this paper.

Executing Task Graphs Using Work-Stealing

Kunal Agrawal
Washington University in St Louis
St Louis, MO 63130, USA

Charles E. Leiserson
Massachusetts Institute of Technology
Cambridge, MA 02139, USA

Jim Sukha
Massachusetts Institute of Technology
Cambridge, MA 02139, USA

Abstract

Nabbit is a work-stealing library for execution of task graphs with arbitrary dependencies which is implemented as a library for the multithreaded programming language Cilk++. We prove that Nabbit executes static task graphs in parallel in time which is asymptotically optimal for graphs whose nodes have constant in-degree and out-degree. To evaluate the performance of Nabbit, we implemented a dynamic program representing the Smith-Waterman algorithm, an irregular dynamic program on a two-dimensional grid. Our experiments indicate that when task-graph nodes are mapped to reasonably sized blocks, Nabbit exhibits low overhead and scales as well as or better than other scheduling strategies. The Nabbit implementation that solves the dynamic program using a task graph even manages in some cases to outperform a divide-and-conquer implementation for directly solving the same dynamic program. Finally, we extend both the Nabbit implementation and the completion-time bounds to handle dynamic task graphs, that is, graphs whose nodes and edges are created on the fly at runtime.

Structuring Execution of OpenMP Applications for Multicore Architectures

François Broquedis, Olivier Aumage, Brice Goglin,
Samuel Thibault, Pierre-Andr Wacrenier and Raymond Namyst
INRIA, LaBRI, Université of Bordeaux
F-33405 Talence – France
Email: {broquedi,aumage,goglin,thibault,wacren,namyst}@labri.fr

Abstract

The now commonplace multi-core chips have introduced, by design, a deep hierarchy of memory and cache banks within parallel computers as a tradeoff between the user friendliness of shared memory on the one side, and memory access scalability and efficiency on the other side. However, to get high performance out of such machines requires a dynamic mapping of application tasks and data onto the underlying architecture. Moreover, depending on the application behavior, this mapping should favor cache affinity, memory bandwidth, computation synchrony, or a combination of these. The great challenge is then to perform this hardware-dependent mapping in a portable, abstract way.

To meet this need, we propose a new, hierarchical approach to the execution of OpenMP threads onto multicore machines. Our ForestGOMP runtime system dynamically generates structured trees out of OpenMP programs. It collects relationship information about threads and data as well. This information is used together with scheduling hints and hardware counter feedback by the scheduler to select the most appropriate threads and data distribution. ForestGOMP features a highlevel platform for developing and tuning portable threads schedulers. We present several applications for which we developed specific scheduling policies that achieve excellent speedups on 16-core machines.

Oversubscription on Multicore Processors

Costin Iancu, Steven Hofmeyr, Filip Blagojević and Yili Zheng
Lawrence Berkeley National Laboratory
Berkeley, USA
{cciancu,shofmeyr,fblagojevic,yzheng}@lbl.gov

Abstract

Existing multicore systems already provide deep levels of thread parallelism; hybrid programming models and composability of parallel libraries are very active areas of research within the scientific programming community. As more applications and libraries become parallel, scenarios where multiple threads compete for a core are unavoidable. In this paper we evaluate the impact of task oversubscription on the performance of MPI, OpenMP and UPC implementations of the NAS Parallel Benchmarks on UMA and NUMA multi-socket architectures. We evaluate explicit thread affinity management against the default Linux load balancing and discuss sharing and partitioning system management techniques. Our results indicate that oversubscription provides beneficial effects for applications running in competitive environments. Sharing all the available cores between applications provides better throughput than explicit partitioning. Modest levels of oversubscription improve system throughput by 27% and provide better performance isolation of applications from their co-runners: best overall throughput is always observed when applications share cores and each is executed with multiple threads per core. Rather than "resource" symbiosis, our results indicate that the determining behavioral factor when applications share a system is the granularity of the synchronization operations.

Session 24
Distributed Algorithms

A Scalable Algorithm for Maintaining Perpetual System Connectivity in Dynamic Distributed Systems

Tarun Bansal
Department of Computer Science
Ohio State University
Columbus, OH 43202, USA
Email: bansal@cse.ohio-state.edu

Neeraj Mittal
Department of Computer Science
The University of Texas at Dallas
Richardson, TX 75080, USA
Email: neerajm@utdallas.edu

Abstract

We investigate the problem of maintaining a topology with small degree as well as small diameter in a dynamic distributed system such that the system always stays connected and processes that wish to leave the system can do so quickly. Perpetual system connectivity is necessary to solve many important problems in dynamic distributed systems, including atomic broadcast and stable property detection, that need strict (deterministic) guarantees about system connectivity to be solvable. To our knowledge, in all existing topology maintenance algorithms for asynchronous distributed systems that provide perpetual system connectivity, either: (i) the topology has large worst-case degree and/or diameter, (ii) a process may experience high worst-case delay when leaving the system, or (iii) processes cannot join and/or leave concurrently.

In this paper, we present a spanning tree maintenance algorithm that satisfies the following desirable properties. First, the spanning tree has small maximum degree of $O(1)$ and small maximum diameter of $O(\log N)$, where N denotes the maximum size of the system. Second, any process can leave the system within $O(\log N)$ time even in the presence of concurrent arrivals and departures. Third, the system always stays connected. We show using a simple knowledge-based argument that, in any algorithm that maintains perpetual connectivity such that the topology has either worst-case diameter of $\Omega(\log N)$ or worst-case degree of $O(1)$, the departure of a process may be delayed by $\Omega(\log \log N)$ time in the worst-case.

Algorithmic Mechanisms for Internet-based Master-Worker Computing with Untrusted and Selfish Workers

Antonio Fernández Anta
LADyR, GSyC
Universidad Rey Juan Carlos
Madrid, Spain
anto@gsyc.es

Chryssis Georgiou
Dept. of Computer Science
University of Cyprus
Nicosia, Cyprus
chryssis@cs.ucy.ac.cy

Miguel A. Mosteiro
LADyR, GSyC, Universidad Rey Juan Carlos
Madrid, Spain and
Dept. of Computer Science, Rutgers University
Piscataway, NJ, USA
mosteiro@cs.rutgers.edu

Abstract

We consider Internet-based master-worker computations, where a master processor assigns, across the Internet, a computational task to a set of untrusted worker processors, and collects their responses; examples of such computations are the "@home" projects such as SETI. Prior work dealing with Internet-based task computations has either considered only rational, or only malicious and altruistic workers. Altruistic workers always return the correct result of the task, malicious workers always return an incorrect result, and rational workers act based on their self-interest. However, in a massive computation platform, such as the Internet, it is expected that all three type of workers coexist. Therefore, in this work we study Internet-based master-worker computations in the presence of Malicious, Altruistic, and Rational workers. A stochastic distribution of the workers over the three types is assumed. Considering all the three types of workers renders a combination of game-theoretic and classical distributed computing approaches to the design of mechanisms for reliable Internet-based computing. Indeed, in this work, such an algorithmic mechanism that makes use of realistic incentives to obtain the correct task result with a parametrized probability is designed. Only when necessary, the incentives are used to force the rational players to a certain equilibrium (which forces the workers to be truthful) that overcomes the attempts of the malicious workers to deceive the master. Finally, the mechanism is analyzed in two realistic Internet-based master-worker applications. This work is an example of how game theory can be used as a tool to formalize and solve a practical Distributed Computing problem such as Internet supercomputing.

Stabilizing Pipelines for Streaming Applications

Andrew Berns, Anurag Dasgupta and Sukumar Ghosh
Department of Computer Science
The University of Iowa
Iowa City, Iowa, USA
{adberns, adasgupt, ghosh}@cs.uiowa.edu

Abstract

In this paper, we study a compositional approach to designing a class of stabilizing distributed systems. We show that the linear pipelined composition of a number of stabilizing modules is inherently stabilizing, and is a useful method of constructing scalable stabilizing solutions for streaming applications that are on the rise in peer-to-peer and sensor networks. We present the correctness proof and complexity analysis of the composition for a linear pipeline. Subsequently, we generalize the pipelined composition to alternative, concurrent, and repetitive versions, investigate the stabilization properties of these versions, and present a set of conditions under which these extended constructions retain their stabilization properties.

A Dynamic Approach for Characterizing Collusion in Desktop Grids

Louis-Claude Canon
Nancy University, LORIA, LaBRI

Emmanuel Jeannot
INRIA, LORIA, LaBRI

Jon Weissman
Dept. of Computer Science and Engineering
University of Minnesota, Twin Cities

Abstract

By exploiting idle time on volunteer machines, desktop grids provide a way to execute large sets of tasks with negligible maintenance and low cost. Although desktop grids are attractive for cost-conscious projects, relying on external resources may compromise the correctness of application execution due to the well-known unreliability of nodes. In this paper, we consider the most challenging threat model: organized groups of cheaters that may collude to produce incorrect results. We propose two on-line algorithms for detecting collusion and characterizing the participant behaviors. Using several real-life traces, we show that our approach is accurate and efficient in identifying collusion and in estimating group behavior.

Session 25

Automatic Tuning and Automatic Parallelization

Offline Library Adaptation Using Automatically Generated Heuristic

Frédéric de Mesmay, Yevgen Voronenko and Markus Püschel
Department of Electrical and Computer Engineering
Carnegie Mellon University, Pittsburgh, USA
{fdemesma, yvoronen, pueschel}@ece.cmu.com

Abstract

Automatic tuning has emerged as a solution to provide high-performance libraries for fast changing, increasingly complex computer architectures. We distinguish *offline* adaptation (e.g., in ATLAS) that is performed during installation without the full problem description from *online* adaptation (e.g., in FFTW) that is performed at runtime. Offline adaptive libraries are simpler to use, but, unfortunately, writing the adaptation heuristics that power them is a daunting task. The overhead of online adaptive libraries, on the other hand, makes them unsuitable for a number of applications. In this paper, we propose to automatically generate heuristics in the form of decision trees using a statistical classifier, effectively converting an online adaptive library into an offline one. As testbed we use Spiral-generated adaptive transform libraries for current multicores with vector extensions. We show that replacing the online search with generated decision trees maintains a performance competitive with vendor libraries while allowing for a simpler interface and reduced computation overhead.

An Auto-Tuning Framework for Parallel Multicore Stencil Computations

Shoaib Kamil (LBNL/UC Berkeley), Cy Chan (CSAIL, MIT), Leonid Oliker (LBNL),
John Shalf (NERSC/LBNL) and Samuel William (LBNL)

Abstract

Although stencil auto-tuning has shown tremendous potential in effectively utilizing architectural resources, it has hitherto been limited to single kernel instantiations; in addition, the large variety of stencil kernels used in practice makes this computation pattern difficult to assemble into a library. This work presents a stencil auto-tuning framework that significantly advances programmer productivity by *automatically* converting a straightforward sequential Fortran 95 stencil expression into tuned parallel implementations in Fortran, C, or CUDA, thus allowing performance portability across diverse computer architectures, including the AMD Barcelona, Intel Nehalem, Sun Victoria Falls, and the latest NVIDIA GPUs. Results show that our generalized methodology delivers significant performance gains of up to 22× speedup over the reference serial implementation. Overall we demonstrate that such domain-specific auto-tuners hold enormous promise for architectural efficiency, programmer productivity, performance portability, and algorithmic adaptability on existing and emerging multicore systems.

DynTile: Parametric Tiled Loop Generation for Parallel Execution on Multicore Processors

Albert Hartono[1], Muthu Manikandan Baskaran[1], J. Ramanujam[2] and P. Sadayappan[1]
[1]Department of Computer Science and Engineering
Ohio State University, Columbus, Ohio 43210C1277
{hartonoa, baskaran, saday}@cse.ohio-state.edu
[2]Department of Electrical and Computer Engineering
Louisiana State University, Baton Rouge, Louisiana 70803C5901
jxr@ece.lsu.edu

Abstract

Loop tiling is an important compiler transformation used for enhancing data locality and exploiting coarse-grained parallelism. Tiled codes in which tile sizes are runtime parameters—called *parametrically-tiled* codes—are important for empirical tuning systems like ATLAS. Some recent work has addressed the problem of generating sequential parametric tiled code. In this paper we describe *DynTile*, a system for transforming untiled sequential input C code containing affine imperfectly nested loops to parametrically tiled code for parallel execution on multicore processors. The effectiveness of the system is demonstrated using a number of benchmarks on an eight-core system.

Using Focused Regression for Accurate Time-Constrained Scaling of Scientific Applications

Brad Barnes[1], Jeonifer Garren[2], David K. Lowenthal[3], Jaxk Reeves[2], Bronis R. de Supinski[4], Martin Schulz[4] and Barry Rountree[3]
[1]Department of Computer Science, The University of Georgia
[2]Department of Statistics, The University of Georgia
[3]Department of Computer Science, The University of Arizona
[4]Lawrence Livermore National Laboratory

Abstract

Many large-scale clusters now have hundreds of thousands of processors, and processor counts will be over one million within a few years. Computational scientists must scale their applications to exploit these new clusters. *Time-constrained scaling*, which is often used, tries to hold total execution time constant while increasing the problem size along with the processor count. However, complex interactions between parameters, the processor count, and execution time complicate determining the input parameters that achieve this goal.

In this paper we develop a novel gray-box, focused regression-based approach that assists the computational scientist with maintaining constant run time on increasing processor counts. Combining application-level information from a small set of training runs, our approach allows prediction of the input parameters that result in similar per-processor execution time at larger scales. Our experimental validation across seven applications showed that median prediction errors are less than 13%.

Session 26
Architectural Support for Runtime Systems

A Low Cost Split-Issue Technique to Improve Performance of SMT Clustered VLIW Processors

Manoj Gupta, Fermín Sánchez and Josep Llosa
Department of Computer Architecture
Universitat Politècnica de Catalunya
Barcelona, Spain
{mgupta,fermin,josepll}@ac.upc.edu

Abstract

Very Long Instruction Word (VLIW) processors are a popular choice in embedded domain due to their hardware simplicity, low cost and low power consumption. Simultaneous MultiThreading (SMT) is a popular technique for improving processor performance. To maintain execution semantics, a VLIW instruction needs to be issued in entirety, which restricts the opportunities in SMT. Split-issue at operation-level is a technique that allows issuing a VLIW instruction in parts without breaking execution semantics. Issuing an instruction in parts allows non-conflicting part of an instruction to be issued along with other instructions and improves SMT performance. However, implementing split-issue at operation-level requires complex structures and is not practical for an embedded VLIW processor. This paper proposes cluster-level split-issue, which implements split-issue at a cluster-level boundary for clustered VLIW processors. Cluster-level split-issue has a very low hardware overhead in contrast to split-issue at operation-level. Experimental results show that cluster-level split-issue, despite being more restrictive than split-issue at operation-level, achieves similar performance and improves SMT performance significantly.

Exploiting Inter-thread Temporal Locality for Chip Multithreading

Jiayuan Meng
Department of Computer Science
University of Virginia
Charlottesville, Virginia
jm6dg@virginia.edu

Jeremy W. Sheaffer
Department of Computer Science
University of Virginia
Charlottesville, Virginia
jws9c@cs.virginia.edu

Kevin Skadron
Department of Computer Science
University of Virginia
Charlottesville, Virginia
skadron@cs.virginia.edu

Abstract

Multi-core organizations increasingly support multiple threads per core. Threads on a core usually share a single first-level data cache, so thread schedulers must try to minimize cache contention among threads. While this has been studied for concurrent threads with *disjoint* working sets, the problem has not been addressed for multi-threaded data-parallel workloads in which threads can be scheduled or constructed to improve inter-thread cache sharing. This paper proposes the *symbiotic affinity scheduling (SAS)* algorithm in which work is first partitioned according to the number of cores (i.e., the number of caches), and these partitions are then subdivided and scheduled among each core's available thread contexts so that threads sharing a core operate on neighboring elements to maximize cache locality.

We demonstrate this concept with a series of data-parallel benchmarks. Simulations on M5 achieve an average speedup of 1.69× and 36% energy savings over conventional scheduling techniques that are oblivious to whether threads share a cache. Even compared to an approach that extends oblivious scheduling to ensure that the sum of the threads' working sets fits in the cache, symbiotic affinity scheduling is able to exploit greater temporal locality and provide 30% performance gains on average. Symbiosis also outperforms adaptive contention reduction techniques by 17%.

Profitability-Based Power Allocation for Speculative Multithreaded Systems

Polychronis Xekalakis
Intel Barcelona Research Center
Intel Labs Barcelona C UPC
polychronisX.xekalakis@intel.com

Nikolas Ioannou, Salman Khan and Marcelo Cintra
School of Informatics
University of Edinburgh
{nikolas.ioannou, salman.khan}@ed.ac.uk,
mc@staffmail.ed.ac.uk

Abstract

With the shrinking of transistors continuing to follow Moore's Law and the non-scalability of conventional out-of-order processors, multi-core systems are becoming the design choice for industry. Performance extraction is thus largely alleviated from the hardware and placed on the programmer/ compiler camp, who now have to expose Thread Level Parallelism (TLP) to the underlying system in the form of explicitly parallel applications.

Unfortunately, parallel programming is hard and error-prone. The programmer has to parallelize the work, perform the data placement, and deal with thread synchronization. Systems that support speculative multithreaded execution like Thread Level Speculation (TLS), offer an interesting alternative since they relieve the programmer from the burden of parallelizing applications and correctly synchronizing them.

Since systems that support speculative multithreading usually treat all threads equally, they are energy-inefficient. This inefficiency stems from the fact that speculation occasionally fails and, thus, power is spent on threads that will have to be discarded. In this paper we propose a power allocation scheme for TLS systems, based on Dynamic Voltage and Frequency Scaling (DVFS), that tries to remedy this inefficiency. More specifically, we propose a profitability-based power allocation scheme, where we "steal" power from non-profitable threads and use it to speed up more useful ones. We evaluate our techniques for a state-of-the-art TLS system and show that, with minimal hardware support, they lead to improvements in ED of up to 39.6% with an average of 21.2%, for a subset of the SPEC 2000 Integer benchmark suite.

Evaluating Standard-Based Self-Virtualizing Devices: A Performance Study on 10 GbE NICs with SR-IOV Support

Jiuxing Liu
IBM T. J. Watson Research Center
19 Skyline Drive, Hawthorne, NY 10532
jl@us.ibm.com

Abstract

Virtual machine (VM) technologies have made much progress in improving the efficiency of virtualizing CPU and memory. However, achieving high performance for I/O virtualization remains a challenge, especially for high speed networking devices such as 10 Gigabit Ethernet (10GbE) NICs, and commonly used software-based I/O virtualization approaches usually suffer significant performance degradation compared with native hardware.

One promising approach to address the performance issue of I/O virtualization is to use single root I/O virtualization (SR-IOV) devices which have been standardized by the PCI-SIG. With SRIOV, a PCI Express (PCIe) device can present itself as multiple virtual devices. By dedicating a virtual device to a single VM, it is possible for the VM to access the virtual device hardware directly, thus reducing overheads such as context/control switches and extra memory copies. However, SR-IOV comes with its limitations such as requiring special hardware support and increased complexity in achieving VM tasks such as checkpointing, migration, and record/reply. Therefore, it is very important for us to fully understand the performance benefit of SR-IOV before adopting it. Unfortunately, there exists little previous work which provides such information.

In this paper, we present a detailed performance evaluation of a 10 GbE SR-IOV PCIe device from Neterion in the KVM (Kernel-based Virtual Machine) virtualization environment. Our focus is not just performance metrics such as bandwidth and latency, but also other aspects of the system such as CPU utilization, memory access, VM exits, and host/guest interrupts. We have also studied several important factors that affect networking performance in both virtualized and native systems. These include issues such as the MTU size, the use of a single processor versus multiple processors, IRQ affinity, and IRQ distribution.

Our experiments show that the hardware-based SR-IOV approach provides superior performance to the software-based approach in KVM. SR-IOV can achieve close to line rate TCP communication (9.3 Gbps) for both transmitting (Tx) and receiving (Rx) with the standard 1500 byte Ethernet MTU, although it does consume more CPU cycles than the native (non-virtualized) case. Overall, our evaluation demonstrates that the SR-IOV approach has great potential to achieve high performance I/O in a virtualized environment.

Session 27

Client-Server System Management and Analysis

QoS Assessment of WS-BPEL Processes through non-Markovian Stochastic Petri Nets

Dario Bruneo, Salvatore Distefano, Francesco Longo, Marco Scarpa
Dipartimento di Matematica
Università di Messina
Messina, Italy
Email: {dbruneo,sdistefano,flongo,mscarpa}@unime.it

Abstract

Service Oriented Architecture (SOA) is the most important and effective software paradigm to design Internet-based services. Using the SOA technology, value-added services can be easily deployed as a combination of existing Web services. In this context, WS-BPEL language has become the SOA industrial standard. To allow services to be composed, business relationships between providers and consumers have to be adequately managed. This implies that a formal definition of Quality of Service (QoS) is agreed and that effective tools for its measurement have to be developed. However, the design of QoS guaranteed composed Web services still requires several efforts due to the highly distributed nature of such software applications. This work aims at proposing a methodology to evaluate Web service performance at the earliest design phase. We present a novel technique to translate WSBPEL processes into non-Markovian stochastic Petri nets with the final goal to evaluate parameters such as service time distribution and service reliability. The obtained model can be numerically solved through automatic tools, allowing to investigate the service behavior under different operating conditions and thus helping software engineers to develop QoS-guaranteed software solutions.

Power-aware Resource Provisioning in Cluster Computing

Kaiqi Xiong
Department of Computer Science
Texas A&M University
Commerce, TX USA

Abstract

The high power consumption of cluster computing infrastructures has become a major concern. It leads to the increased heat dissipation and decreased reliability of cluster servers. Power management becomes a critical issue in cluster computing. In this paper, we start with an analysis of the relationship between cluster performance and power consumption. We study both the problem of minimizing the average end-to-end delay with the constraint of average energy consumption and the problem of minimizing the average energy consumption of cluster service requests with the constraint of an average end-to-end delay for customer services. We propose novel approaches to solving these two problems. In an effort to maximize profits, a service provider only provides sufficient resources to ensure quality of services (QoS) but often avoid over provisioning to meet QoS defined in a service level agreement (SLA) which is a contract agreed between a customer and a service provider. We present an approach for optimizing SLA-based resource provisioning in cluster computing in that we minimize the total cost of cluster servers owned by a service provider while satisfying the requirements of both a percentile of the end-to-end delay and average energy consumption. Numerical experiments show that the proposed approach is efficient and accurate for the SLA-based resource provisioning problem in cluster computing.

Using the Middle Tier to Understand Cross-Tier Delay in a Multi-tier Application

Haichuan Wang[1], Qiming Teng[2], Xiao Zhong[2] and Peter F. Sweeney[3]
[1]IBM Research, Shanghai, China
Email: wanghaic@cn.ibm.com
[2]IBM Research, Beijing, China
Email: {tengqim,zhongx}@cn.ibm.com
[3]IBM Research, Hawthorne, New York, USA
Email: pfs@us.ibm.com

Abstract

Understanding the cause of poor performance in a multi-tier enterprise application is challenging, because a performance bottleneck on any tier may cause the whole system to be under utilized, and to fail its throughput or quality of service goals.

This paper presents an approach that focuses on the application server to identify bottlenecks in a multi-tier application that are caused by tiers other then the application server. The approach uses a performance tool, named SLICE, that selectively tracks method invocations that cross tier boundaries, and extracts contextual information associated with these invocations. SLICE also collects information from the operating systems scheduler to determine when a thread is blocked. Using the contextual information from method invocations and the information of when a thread is blocked from the operating system, SLICE computes cross tier delay. Experiments on DayTrader, a multi-tier application, show that performance bottlenecks caused by clients or database servers can be identified using cross tier delay.

Service and Resource Discovery in Cycle-Sharing Environments with a Utility Algebra

João Nuno Silva, Paulo Ferreira and Luís Veiga
INESC-ID / Instituto Superior Técnico - Technical University of Lisbon
Rua Alves Redol N^o 9, 1000 Lisboa, Portugal
Email: {joao.n.silva, paulo.ferreira, luis.veiga}@inesc-id.pt

Abstract

The Internet has witnessed a steady and widespread increase in available idle computing cycles and computing resources in general. Such available cycles simultaneously allow and foster the increase in development of existing and new computationally demanding applications, driven by algorithm complexity, intensive data processing, or both. Available cycles may be harvested from several scenarios, ranging from college or office LANs, cluster, grid and utility or cloud computing infrastructures, to peer-to-peer overlay networks.

Existing resource discovery protocols have a number of shortcomings for the existing variety of cycle sharing scenarios. They either i) were designed to return only a binary answer stating whether a remote computer fulfills the requirements, ii) rely on centralized schedulers (or coherently replicated) that are impractical in certain environments such as peer-to-peer computing, iii) they are not extensible as it is impossible to define new resources to be discovered and evaluated or new ways to evaluate them.

In this paper we present a novel, extensible, expressive, and flexible requirement specification algebra and resource discovery middleware. Besides standard resources (CPU, memory, network bandwidth,...), application developers may define new resource requirements and new ways to evaluate them. Application programmers can write complex requirements (that evaluate several resources) using fuzzy logic operators. Each resource evaluation (either standard or specially coded) returns a value between 0.0 and 1.0 stating the capacity to (partially) fulfill the requirement, considering client-specific utility depreciation (i.e., *partial-utility*, a downgraded measure of how the user assesses the available resources) and policies for combined utility evaluation. By comparing the values obtained from the various hosts, it is possible to precisely know which ones best fulfill each client's needs, regarding a set of required resources.

Session 28
Parallel Graph Algorithms II

Optimization of Linked List Prefix Computations on Multithreaded GPUs Using CUDA

Zheng Wei and Joseph JaJa
Department of Electrical and Computer Engineering
Institute for Advanced Computer Studies, University of Maryland
College Park, U. S. A
{zwei, joseph}@umiacs.umd.edu

Abstract

We present a number of optimization techniques to compute prefix sums on linked lists and implement them on multithreaded GPUs using CUDA. Prefix computations on linked structures involve in general highly irregular fine grain memory accesses that are typical of many computations on linked lists, trees, and graphs. While the current generation of GPUs provides substantial computational power and extremely high bandwidth memory accesses, they may appear at first to be primarily geared toward streamed, highly data parallel computations. In this paper, we introduce an optimized multithreaded GPU algorithm for prefix computations through a randomization process that reduces the problem to a large number of fine-grain computations. We map these fine-grain computations onto multithreaded GPUs in such a way that the processing cost per element is shown to be close to the best possible. Our experimental results show scalability for list sizes ranging from 1M nodes to 256M nodes, and significantly improve on the recently published parallel implementations of list ranking, including implementations on the Cell Processor, the MTA-8, and the NVIDIA GeForce 200 series. They also compare favorably to the performance of the best known CUDA algorithm for the scan operation on the Tesla C1060.

Parallel External Memory Graph Algorithms

Lars Arge
MADALGO
University of Aarhus
Aarhus, Denmark
large@madalgo.au.dk

Michael T. Goodrich
University of California – Irvine
Irvine, CA USA
goodrich@ics.uci.edu

Nodari Sitchinava
MADALGO
University of Aarhus
Aarhus, Denmark
nodari@madalgo.au.dk

Abstract

In this paper, we study parallel I/O efficient graph algorithms in the Parallel External Memory (PEM) model, one of the private-cache chip multiprocessor (CMP) models. We study the fundamental problem of list ranking which leads to efficient solutions to problems on trees, such as computing lowest common ancestors, tree contraction and expression tree evaluation. We also study the problems of computing the connected and biconnected components of a graph, minimum spanning tree of a connected graph and ear decomposition of a biconnected graph. All our solutions on a P-processor PEM model provide an optimal speedup of $\Theta(P)$ in parallel I/O complexity and parallel computation time, compared to the single-processor external memory counterparts.

Engineering a Scalable High Quality Graph Partitioner

Manuel Holtgrewe, Peter Sanders and Christian Schulz
Karlsruhe Institute of Technology, Karlsruhe, Germany
Email: manuel.holtgrewe@fu-berlin.de, sanders@kit.edu, c_schulz@ira.uka.de

Abstract

We describe an approach to parallel graph partitioning that scales to hundreds of processors and produces a high solution quality. For example, for many instances from Walshaw's benchmark collection we improve the best known partitioning. We use the well known framework of multi-level graph partitioning. All components are implemented by scalable parallel algorithms. Quality improvements compared to previous systems are due to better prioritization of edges to be contracted, better approximation algorithms for identifying matchings, better local search heuristics, and perhaps most notably, a parallelization of the FM local search algorithm that works more locally than previous approaches.

Session 29
Algorithms for Wireless Networks

Sparse Power-Efficient Topologies for Wireless Ad Hoc Sensor Networks

Amitabha Bagchi
Department of Computer Science and Engineering
Indian Institute of Technology
Hauz Khas, New Delhi 110016, India
Email: bagchi@cse.iitd.ernet.in

Abstract

We study the problem of power-efficient routing for multi-hop wireless ad hoc sensor networks. The guiding insight of our work is that unlike an ad hoc wireless network, a wireless ad hoc sensor network does not require full connectivity among the nodes. As long as the sensing region is well covered by connected nodes, the network can perform its task. We consider two kinds of geometric random graphs as base interconnection structures: unit disk graphs $UDG(2,\lambda)$ and k-nearest-neighbor graphs $NN(2,k)$ built on points generated by a Poisson point process of density λ in $\mathbb{R}^2$. We provide subgraph constructions for these two models–$UDG-SENS(2,\lambda)$ and $NN-SENS(2,k)$ respectively–and show that there are values of the parameters λ and k, λ_s and k_s respectively, above which these constructions have the following good properties: (i) they are sparse; (ii) they are power-efficient in the sense that the graph distance is no more than a constant times the Euclidean distance between any pair of points; (iii) they cover the space well; (iv) the subgraphs can be set up easily in a distributed fashion using local information at each node. We also describe a simple local algorithm for routing packets on these subgraphs.

Contention-based Georouting with Guaranteed Delivery, Minimal Communication Overhead, and Shorter Paths in Wireless Sensor Networks

Stefan Rührup
OFFIS Institute for Information Technology
Oldenburg, Germany
stefan.ruehrup@offis.de
Ivan Stojmenović
School of Information Technology and Engineering,
University of Ottawa, Canada
ivan@site.uottawa.ca

Abstract

Nodes in contention-based (beaconless) georouting forward packets towards a known destination position without the knowledge of the neighborhood. The only existing methods [17], [20] that guarantee delivery in unit disk graphs (UDG) require runtime planarization of the communication graph with either unbounded message overhead per hop while preserving the Gabriel graph property of the subgraph, or a constant overhead per hop with up to 13 control messages.

In this paper we show that the next hop can be selected directly by a contention mechanism and without prior planarization. Existing greedy routing methods select the next hop with 3 messages using a RTS-CTS-DATA scheme in a timerbased contention where only the next hop neighbor responds. We extend this to provide also recovery from local minima with 3 messages per hop by the Rotational Sweep algorithm. We prove that our algorithm guarantees delivery in UDGs, and also yields routes that are shorter than or equal to the combined greedy and face routing with Gabriel graph planarization. Simulation results show that especially the duration of the contention process can be significantly reduced.

Our algorithm can be also used for conventional beaconbased routing with guaranteed delivery without prior planarization, replacing the complicated implementation from [19] by a very simple method which evaluates angular distances to select the proper forwarding neighbor. It also provides a simple network boundary detection algorithm, with or without beacons.

Midpoint Routing Algorithms for Delaunay Triangulations

Weisheng Si and Albert Y. Zomaya
Centre for Distributed and High Performance Computing
School of Information Technologies, University of Sydney
Sydney, NSW, Australia
{weisheng, zomaya}@it.usyd.edu.au

Abstract

Memoryless online routing (MOR) algorithms are important for the applications with only local information available to make routing decisions. This paper gives two new MOR algorithms for a class of geometric graphs called Delaunay triangulations (DTs): the Midpoint Routing algorithm and the Compass Midpoint algorithm. More meaningfully, the former is generalized into a set of MOR algorithms that use the Euclidean distance as the reference and work for DTs, and the latter is generalized into a set of MOR algorithms that use the direction as the reference and work for DTs. Many other existing MOR algorithms can also be covered by these two sets. Finally, the two new algorithms are evaluated and compared with other existing MOR algorithms, and the experimental results give new findings on the performances of these algorithms in average and general cases.

A Local, Distributed Constant-Factor Approximation Algorithm for the Dynamic Facility Location Problem

Bastian Degener, Barbara Kempkes and Peter Pietrzyk
Heinz Nixdorf Institute
Computer Science Department, University of Paderborn
33095 Paderborn, Germany
degener@uni-paderborn.de, barbaras@uni-paderborn.de, toon@uni-paderborn.de

Abstract

We present a distributed, local solution to the dynamic facility location problem in general metrics, where each node is able to act as a facility or a client. To decide which role it should take, each node keeps up a simple invariant in its local neighborhood. This guarantees a global constant factor approximation when the invariant is satisfied at all nodes. Due to the changing distances between nodes, invariants can be violated. We show that restoring the invariants is bounded to a constant neighborhood, takes logarithmic (in the number of nodes) asynchronous rounds and affects each node at most twice per violation.

Session 30

Analysis of heterogeneity and future platforms

Toward Understanding Heterogeneity in Computing

Arnold L. Rosenberg and Ron C. Chiang
Dept. of Electrical & Computer Engineering
Colorado State University
Fort Collins, CO 80523, USA
{rsnbrg,ron.chiang}@colostate.edu

Abstract

Heterogeneity complicates the efficient use of multicomputer platforms, but does it enhance their performance? their cost effectiveness? How can one measure the power of a heterogeneous assemblage of computers ("cluster," for short), both in absolute terms (how powerful is this cluster) and relative terms (which cluster is more powerful)? What makes one cluster more powerful than another? Is one better off with a cluster that has one super-fast computer and the rest of "average" speed or with a cluster all of whose computers are "moderately" fast? If you could replace just one computer in your cluster with a faster one, which computer would you choose: the fastest? the slowest? How does one even ask questions such as these in a rigorous, yet tractable manner? A framework is proposed, and some answers are derived, a few rather surprising. Three highlights: (1) If one can replace only one computer in a cluster by a faster one, it is provably (almost) always most advantageous to replace the fastest one. (2) If the computers in two clusters have the same mean speed, then, empirically, the cluster with the larger variance in speed is (almost) always the faster one. (3) Heterogeneity can actually lend power to a cluster!

Balls into Non-uniform Bins

Petra Berenbrink
School of Computing Science
Simon Fraser University
Burnaby, B.C., V5A IS6, Canada
Email: petra@cs.sfu.ca

André Brinkmann
Paderborn Center for Parallel Computing
University of Paderborn
33102 Paderborn, Germany
Email: brinkman@uni-paderborn.de

Tom Friedetzky
School of Engineering and Computing Sciences
Durham University
Durham DH1 3LE, United Kingdom
Email: tom.friedetzky@dur.ac.uk

Lars Nagel
School of Engineering and Computing Sciences
Durham University
Durham DH1 3LE, United Kingdom
Email: lars.nagel@dur.ac.uk

Abstract

Balls-into-bins games for uniform bins are widely used to model randomized load balancing strategies. Recently, balls-into-bins games have been analysed under the assumption that the selection probabilities for bins are not uniformly distributed. These new models are motivated by properties of many peer-to-peer (P2P) networks, which are not able to perfectly balance the load over the bins. While previous evaluations try to find strategies for uniform bins under non-uniform bin selection probabilities, this paper investigates heterogeneous bins, where the "capacities" of the bins might differ significantly. We show that heterogeneous environments can even help to distribute the load more evenly, and that the load difference between bins can be bounded by O(log log n) if each ball has two random choices, where n is the number of bins. Our analysis and simulation results show, for the first time, that the maximum load in heterogeneous balls-into-bins games is independent from the overall system capacity C and that bigger bins therefore can help to achieve good load balancing properties.

An Introductory Exascale Feasibility Study for FFTs and Multigrid

Hormozd Gahvari and William Gropp
Computer Science Department
University of Illinois at Urbana-Champaign
{gahvari,wgropp}@illinois.edu

Abstract

The coming decade is going to see a push towards exascale computing. Assuming gigahertz cores, this means exascale systems will have between 100 million and 1 billion of them to achieve this level of performance. At this scale, some important questions need to be answered on the applications end. What applications are feasible at this scale? What needs to be done to make them scalable? How does the hardware have to adapt to meet application needs? In this paper, we introduce a new feasibility-based approach to answering these questions. Our approach involves finding upper and lower bounds on problem size and machine parameters to determine a feasibility region for the application in question. As the underlying architecture of a future exascale machine is currently unknown, we use LogP-based performance models and vary machine parameters to give architecture-indepenent hardware constraints. We consider both strong-scaling and weak-scaling scenarios, and present results for two applications, the Fast Fourier Transform and basic geometric multigrid. The results show substantial constraints that need to be satisfied to enable exascale performance.

Session 31
Data Management

A Cost-Effective Strategy for Intermediate Data Storage in Scientific Cloud Workflow Systems

Dong Yuan, Yun Yang, Xiao Liu and Jinjun Chen
Faculty of Information and Communication Technologies,
Swinburne University of Technology
Hawthorn, Melbourne, Australia 3122
{dyuan, yyang, xliu, jchen}@swin.edu.au

Abstract

Many scientific workflows are data intensive where a large volume of intermediate data is generated during their execution. Some valuable intermediate data need to be stored for sharing or reuse. Traditionally, they are selectively stored according to the system storage capacity, determined manually. As doing science on cloud has become popular nowadays, more intermediate data can be stored in scientific cloud workflows based on a pay-foruse model. In this paper, we build an Intermediate data Dependency Graph (IDG) from the data provenances in scientific workflows. Based on the IDG, we develop a novel intermediate data storage strategy that can reduce the cost of the scientific cloud workflow system by automatically storing the most appropriate intermediate datasets in the cloud storage. We utilise Amazon's cost model and apply the strategy to an astrophysics pulsar searching scientific workflow for evaluation. The results show that our strategy can reduce the overall cost of scientific cloud workflow execution significantly.

BlobSeer: Bringing High Throughput under Heavy Concurrency to Hadoop Map-Reduce Applications

Bogdan Nicolae
University of Rennes 1
IRISA
Rennes, France
bogdan.nicolae@inria.fr

Diana Moise and Gabriel Antoniu
INRIA
IRISA
Rennes, France
{diana.moise,gabriel.antoniu}@inria.fr

Luc Bougé and Matthieu Dorier
ENS Cachan, Brittany
IRISA
Rennes, France
{luc.bouge,matthieu.dorier}@bretagne.ens-cachan.fr

Abstract

Hadoop is a software framework supporting the Map-Reduce programming model. It relies on the Hadoop Distributed File System (HDFS) as its primary storage system. The efficiency of HDFS is crucial for the performance of Map- Reduce applications. We substitute the original HDFS layer of Hadoop with a new, concurrency-optimized data storage layer based on the BlobSeer data management service. Thereby, the efficiency of Hadoop is significantly improved for data-intensive Map-Reduce applications, which naturally exhibit a high degree of data access concurrency. Moreover, BlobSeer's features (builtin versioning, its support for concurrent append operations) open the possibility for Hadoop to further extend its functionalities. We report on extensive experiments conducted on the Grid5000 testbed. The results illustrate the benefits of our approach over the original HDFS-based implementation of Hadoop.

PreDatA – Preparatory Data Analytics on Peta-Scale Machines

Fang Zheng[1], Hasan Abbasi[1], Ciprian Docan[2], Jay Lofstead[1], Qing Liu[3], Scott Klasky[3]
Manish Parashar[2], Norbert Podhorszki[3], Karsten Schwan[1] and Matthew Wolf[1,3]
[1]College of Computing, Georgia Institute of Technology, Atlanta, GA 30332
[2]Center for Autonomic Computing, Rutgers University, Piscataway, NJ 08854
[3]Oak Ridge National Laboratory, Oak Ridge, TN 37831

Abstract

Peta-scale scientific applications running on High End Computing (HEC) platforms can generate large volumes of data. For high performance storage and in order to be useful to science end users, such data must be organized in its layout, indexed, sorted, and otherwise manipulated for subsequent data presentation, visualization, and detailed analysis. In addition, scientists desire to gain insights into selected data characteristics 'hidden' or 'latent' in these massive datasets while data is being produced by simulations. PreDatA, short for Preparatory Data Analytics, is an approach to preparing and characterizing data while it is being produced by the large scale simulations running on peta-scale machines. By dedicating additional compute nodes on the machine as 'staging' nodes and by staging simulations' output data through these nodes, PreDatA can exploit their computational power to perform select data manipulations with lower latency than attainable by first moving data into file systems and storage. Such in-transit manipulations are supported by the PreDatA middleware through asynchronous data movement to reduce write latency, application-specific operations on streaming data that are able to discover latent data characteristics, and appropriate data reorganization and metadata annotation to speed up subsequent data access. PreDatA enhances the scalability and flexibility of the current I/O stack on HEC platforms and is useful for data pre-processing, runtime data analysis and inspection, as well as for data exchange between concurrently running simulations.

Reconciling Scratch Space Consumption, Exposure, and Volatility to Achieve Timely Staging of Job Input Data

Henry M. Monti and Ali R. Butt
Dept. of Computer Science
Virginia Tech.
Blacksburg, Virginia, USA
Email: {hmonti, butta}@cs.vt.edu

Sudharshan S. Vazhkudai
Computer Science and Mathematics Division
Oak Ridge National Laboratory
Oak Ridge, Tennessee, USA
vazhkudaiss@ornl.gov

Abstract

Innovative scientific applications and emerging dense data sources are creating a data deluge for high-end computing systems. Processing such large input data typically involves copying (or *staging*) onto the supercomputer's specialized high-speed storage, *scratch* space, for sustained high I/O throughput. The current practice of conservatively staging data as early as possible makes the data vulnerable to storage failures, which may entail re-staging and consequently reduced job throughput. To address this, we present a timely staging framework that uses a combination of job start-up time predictions, user-specified intermediate nodes, and decentralized data delivery to coincide input data staging with job start-up. By delaying staging to when it is necessary, the exposure to failures and its effects can be reduced.

Evaluation using both PlanetLab and simulations based on three years of Jaguar (No. 1 in Top500) job logs show as much as 85.9% reduction in staging times compared to direct transfers, 75.2% reduction in wait time on scratch, and 2.4% reduction in usage/hour.

Session 32
Synchronization

Hierarchical Phasers for Scalable Synchronization and Reductions in Dynamic Parallelism

Jun Shirako and Vivek Sarkar
Department of Computer Science
Rice University
Houston, TX, USA
{shirako, vsarkar}@rice.edu

Abstract

The phaser construct is a unification of collective and point-to-point synchronization with dynamic parallelism. This construct gives each task the option of synchronizing on a phaser in signal-only/wait-only mode for producer/consumer synchronization or signal-wait mode for barrier synchronization. A phaser accumulator is a reduction construct that works with phasers in a phased setting. Phasers and accumulators support dynamic parallelism i.e., they allow dynamic addition and removal of tasks from the synchronizations and reductions that they support.

Past implementations of phasers and phaser accumulators have used a single master task to advance a phaser to the next phase and to perform computations for lazy reductions, while also supporting dynamic parallelism. Though the single master approach provides an effective solution for modest levels of parallelism, it quickly becomes a scalability bottleneck as the number of threads increases. To address this limitation, we propose an approach based on hierarchical phasers for scalable synchronization and hierarchical accumulators for scalable reduction. Our approach also includes tunable initialization parameters that specify the degree and number of tiers for the phaser hierarchy, thereby allowing different values to be chosen for different platforms. Our performance results show significant scalability benefits from our approach. To the best of our knowledge, this is the first approach to support hierarchical synchronization and reductions in the presence of dynamic parallelism.

Clustering JVMs with Software Transactional Memory Support

Christos Kotselidis[1], Mikel Luján[1], Mohammad Ansari[2], Konstantinos Malakasis[1],
Behram Kahn[1], Chris Kirkham[1] and Ian Watson[1]
[1]School of Computer Science, The University of Manchester, Manchester UK
{kotselidis, mikel.lujan, malakasis, kahn, kirkham, watson}@cs.manchester.ac.uk
[2]Department of Computer Science,
Umm Al-Qura University, Makkah, Saudi Arabia
mmansari@uqu.edu.sa

Abstract

Affordable transparent clustering solutions to scale non-HPC applications on commodity clusters (such as Terracotta) are emerging for Java Virtual Machines (JVMs). Working in this direction, we propose the Anaconda framework as a research platform to investigate the role Transactional Memory (TM) can play in this domain. Anaconda is a software transactional memory framework that supports clustering of multiple off-the-shelf JVMs on commodity clusters.

The main focus of Anaconda is to investigate the implementation of Java synchronization primitives on clusters by relying on Transactional Memory. The traditional lock based Java primitives are replaced by memory transactions and the framework is responsible for ensuring transactional coherence.

The contribution of this paper is to investigate which kind of TM coherency protocol can be used in this domain and compare the Anaconda framework against the state-of-the-art Terracotta clustering technology. Furthermore, Anaconda tracks TM conflicts at object granularity and provides distributed object replication and caching mechanisms. It supports existing TM coherence protocols while adding a novel decentralized protocol.

The performance evaluation compares Anaconda against three existing TM protocols. Two of these are centralized, while the other is decentralized. In addition, we compare Anaconda against lock-based (coarse, medium grain) implementations of the benchmarks running on Terracotta. Anaconda's performance varies amongst benchmarks, outperforming by 40 to 70% existing TM protocols. Compared to Terracotta, Anaconda exhibits from 19x speedup to 10x slowdown depending on the benchmark's characteristics.

Inter-Block GPU Communication via Fast Barrier Synchronization

Shucai Xiao[1] and Wu-chun Feng[1,2]
[1]Department of Electrical and Computer Engineering
[2]Department of Computer Science
Virginia Tech
Blacksburg, Virginia 24061
Email: {shucai, wfeng}@vt.edu

Abstract

While GPGPU stands for general-purpose computation on graphics processing units, the lack of explicit support for inter-block communication on the GPU arguably hampers its broader adoption as a general-purpose computing device. Interblock communication on the GPU occurs via global memory and then requires barrier synchronization across the blocks, i.e., inter-block GPU communication via barrier synchronization. Currently, such synchronization is only available via the CPU, which in turn, can incur significant overhead.

We propose two approaches for inter-block GPU communication via barrier synchronization: GPU lock-based synchronization and GPU lock-free synchronization. We then evaluate the efficacy of each approach via a micro-benchmark as well as three well-known algorithms — Fast Fourier Transform (FFT), dynamic programming, and bitonic sort. For the microbenchmark, the experimental results show that our GPU lockfree synchronization performs 8.4 times faster than CPU explicit synchronization and 4.0 times faster than CPU implicit synchronization. When integrated with the FFT, dynamic programming, and bitonic sort algorithms, our GPU lock-free synchronization further improves performance by 10%, 26%, and 40%, respectively, and ultimately delivers an overall speed-up of 70x, 13x, and 24x, respectively.

A Lock-Free, Cache-Efficient Multi-Core Synchronization Mechanism for Line-Rate Network Traffic Monitoring

Patrick P. C. Lee
Dept of Computer Science and Engineering
The Chinese University of Hong Kong
Hong Kong
pclee@cse.cuhk.edu.hk

Tian Bu and Girish Chandranmenon
Bell Laboratories
Alcatel-Lucent
Murray Hill, NJ, USA
{tbu, girishc}@alcatel-lucent.com

Abstract

Line-rate data traffic monitoring in high-speed networks is essential for network management. To satisfy the line-rate requirement, one can leverage multi-core architectures to parallelize traffic monitoring so as to improve information processing capabilities over traditional uni-processor architectures. Nevertheless, realizing the full potential of multi-core architectures still needs substantial work, especially in the face of the ever-increasing volume and complexity of network traffic. This paper addresses the issue through the design of a lock-free, cache-efficient synchronization mechanism that serves as a basic building block for a general class of multi-threaded, multi-core traffic monitoring applications. We embed the synchronization mechanism into *MCRingBuffer*, a multi-core shared ring buffer that provides fast data accesses among threads running in different cores. MCRingBuffer allows concurrent lock-free data accesses and improves the cache locality of accessing the control variables that are used for thread synchronization. Through extensive evaluation on an Intel Xeon multi-core machine, we show that MCRingBuffer achieves a throughput gain of up to 5× over existing lock-free ring buffers. Finally, we present a parallel traffic monitoring prototype that is built upon MCRingBuffer, and demonstrate via trace-driven simulation how MCRingBuffer facilitates packet processing at line rate.