

# 21st International Symposium on Computer Architecture and High Performance Computing

# SBAC-PAD 2009

October, 28-31, 2009 São Paulo, SP - Brazil

Program

Promoted By



TCCA TCSC



Organized By



Sponsored By





**SBAC-PAD 2009**

**PROGRAM**



## Conference Overall Schedule

### October 28<sup>th</sup> - Wednesday

	João Calvino Building	Building 13
	João Calvino Auditorium	Room 402
08:30	WEAC	
09:00		
10:30	Coffee break	
10:45	WEAC	LAGRID
12:15	Lunch	

	Building T			
	Room 506	Room 507	Room 508	Room 509
13:15			WIC	
13:30	WEAC	SBAC Tutorial 1		LAGRID
15:20	Coffee break			
15:40	WEAC	CTD	WIC	
18:00	End			

### October 29<sup>th</sup> - Thursday

	Ruy Barbosa Auditorium
09:00	Opening
09:25	Keynote 1
10:25	Coffee break
10:40	Keynote 2
11:40	Industrial Talk 1
12:30	Lunch

	Building T		
	Room 506	Room 507	Room 508
14:00	SBAC Session I	WSCAD Session I	SINAPAD
15:50	Coffee break		
16:05	SBAC Session II	WSCAD Session II	SINAPAD
18:00	End		

	Building 14		
	Room 4	Room 5	Room 6
13:00	Warmup - Marathon of Parallel Programming		
18:00	End		

18:00	Special Comitee Meeting on Benedito Novaes Garcez Auditorium
-------	--

October 30<sup>th</sup> – Friday

	Ruy Barbosa Auditorium
09:00	Keynote 3
10:00	<b>Coffee break</b>
10:10	Panel
11:40	Industrial Talk 2
12:30	<b>Lunch</b>

	Building T		
	Room 506	Room 507	Room 508
14:00	SBAC Session III	WSCAD Session III	SINAPAD
15:50	<b>Coffee break</b>		
16:10	SBAC Session IV	WSCAD Session IV	SINAPAD
18:00	<b>End</b>		

	Building 14		
	Room 4	Room 5	Room 6
13:00	Marathon of Parallel Programming - The real contest		
18:00	<b>End</b>		

20:00	<b>Conference Dinner – Angélica Grill</b>
-------	---

October 31<sup>th</sup> – Saturday

	Building 40	
	Room 102	Room 103
09:00	SBAC Session V	WSCAD Session V
10:50	<b>Coffee break</b>	
11:10	SBAC Session VI	WSCAD Session VI
12:30	<b>Lunch</b>	

	Building 40		
	Room 102	Room 103	Room 104
14:00	Industrial Talk 3		
14:50	Minicurso 1	Minicurso 2	SBAC Tutorial 2
17:20	<b>Closing</b>		

## SBAC – Keynotes

### SBAC KEYNOTE 1

**Day: Thursday, 29<sup>th</sup> – 9:25 – Ruy Barbosa Auditorium**

- **HPC Clouds: Raining Multicore Services**

Dan Reed (Vice President of Extreme Computing Group – Microsoft)

**Abstract:** Scientific and technical computing are at an inflection point, where system scale, complexity and operating costs, together with the explosive growth of data and the availability of high-speed networks, are changing the ways we will deliver technical computing services. Scientific discovery, business practice and social interactions are moving rapidly from a world of homogeneous and local systems to a world of distributed software, virtual organizations and cloud computing infrastructure, supported by ever larger computing infrastructure. Where is the technology going and what are the research implications? What architectures and policies are appropriate for different user bases? How do we build scalable infrastructure and what is the role of government, academia and industry? How do we develop and support software? What is the ecosystem of components in which they will operate? How do we optimize performance, power and reliability?

### SBAC KEYNOTE 2

**Day: Thursday, 29<sup>th</sup> – 10:40 – Ruy Barbosa Auditorium**

- **The 1000 core microprocessor: Will we be ready for it?**

Yale Patt (The University of Texas at Austin)

**Abstract:** Process technology predicts 50 billion transistors on a chip in the next six or eight years, and there are plenty of existence proofs of single-core microprocessors with around 50 million transistors. Simple mindless arithmetic yields a 1000 processor chip. But because we will be able to produce a chip with 1000 cores does not mean we should produce one. And, if we do, will be able to harness it effectively? I would like to do two things in this talk today: first, review the history that brought us to our current state of multi-core chips, and ask if there isn't a better way, moving forward. A comment like "Today, Moore's Law is about doubling the number of cores on the chip" is one example, in my view, of continuing the current nonsense. Second, I would like to focus on some fundamental problems we will face when the number of transistors permits a 1000 core chip: algorithm development in a domain where the costs of communication and processing are totally different than they are today, off-chip (and on-chip) bandwidth, power considerations, which can involve much more than simply running the clock more slowly. And, as is usually the case, at the heart of whether or not we will be ready is education.

**SBAC KEYNOTE 3****Day: Friday, 30<sup>th</sup> – 09:00 – Ruy Barbosa Auditorium**

- **GPUs in High-Performance Computing – architectures, software stack, education, and applications**

Wen-mei W. Whu (University of Illinois at Urbana Champaign)

**Abstract:** Modern GPUs such as the NVIDIA GTX280 series are massively parallel, many-core compute engines. According to the semiconductor industry scaling roadmap, these compute engines could scale up to 10,000x the peak performance of a circa-2007 microprocessor core by the end of the year 2016. Such a dramatic increase in computation power will likely trigger multiple major science discoveries as well as revolutions in consumer applications. We are experiencing a once-in-a-life-time opportunity in our profession. However, the programming and code optimization models of GPU computing are quite different from those of parallel systems based on traditional CPUS. In this presentation, I will describe the recent progress by the HPC community in building GPU-based clusters, enhanced software stacks, educational materials ([www.courses.ece.uiuc.edu/ece498/al](http://www.courses.ece.uiuc.edu/ece498/al)), and porting applications to fully exploit the current and future GPU computing platforms. I will then discuss the coming challenges and some promising work to address these challenges.



## SBAC – Industrial Talks

### SBAC INDUSTRIAL TALK 1

**Day: Thursday, 29<sup>th</sup> – 11h:40 – Ruy Barbosa Auditorium**

- **High Performance Computing on Windows Platform**

Markus Christen

Microsoft

### SBAC INDUSTRIAL TALK 2

**Day: Friday, 30<sup>th</sup> – 11h:40 – Ruy Barbosa Auditorium**

- **Simplifying the Deployment and Verification of HPC Clusters**

Ricardo Mendel, PhD

Intel

**Abstract:** Clusters are the fastest growing segment in HPC. However, clusters remain difficult to specify, deploy, and manage, especially for new HPC users. The Intel Cluster Ready Program goal is to lower the entry barriers to HPC by defining a reference architecture for ensuring hardware/software integration and providing tools that allow end users to get their work done with minimum knowledge about clusters. In this talk we will present the different components of the Intel Cluster Ready Program, with special emphasis on the Intel Cluster Checker tool, a key component of the program.

### SBAC INDUSTRIAL TALK 3

**Day: Saturday, 29<sup>th</sup> – 14h:00 – Building 40, Room 102**

- **Putting all together: the future is fusion.**

Roberto Brandão, PhD

AMD

**Abstract:** Computer processing has reached a crossroads where the relationship between hardware and software must change to support the increased processing needs of modern computing workloads, including virtualized environments and media-rich applications. Accelerated computing uses specialized hardware to increase the speed of certain processing tasks, offering commercial and consumer users simultaneous energy efficiency, high performance, and low cost. From a hardware perspective, accelerated computing provides a general platform that supports the addition of specialized processing units, or accelerators, to a multicore computer. These accelerators may take the form of off-core chips such as media or network packet accelerators, or they may be additional cores designed for specific types of processing tasks. From a software perspective, the accelerated computing framework will simplify writing parallel applications: developers will have high-level tools that support parallel programming,

and applications will not have to specify which hardware should be used for a given processing task.

AMD has been building expertise and capability in general- and specific-purpose computing for the last several years, including how to effectively couple general- and specific-purpose processors. This activity has built a solid foundation for further research and development in accelerated computing. We are actively collaborating with large research institutions, independent software vendors, the open source community, and other organizations to develop an open ecosystem around accelerated computing. Our commitment to accelerated computing includes designing hardware accelerators, building developer tools, and driving thought leadership by participating in standards development bodies. We believe that accelerated computing embodies a rapidly approaching paradigm shift, and we intend to offer commercial and consumer users flexible computing solutions that meet their needs based on the components available in the growing ecosystem.

This presentation explores the accelerated computing framework. It provides examples of accelerated computing in action today, describes potential uses of accelerators in the near- to mid-term, and makes a clear case that accelerated computing is the natural outcome of the computer industry's current integration trend. It further illustrates that, in full bloom, accelerated computing will create many opportunities to customize computer systems by providing targeted, precise solutions for different kinds of workloads to satisfy the need for application performance, energy efficiency, and low cost.

## SBAC Panel

**SBAC Panel****Day: Friday, 30<sup>th</sup> – 10:10 – Ruy Barbosa Auditorium**

- **Research Challenges and Opportunities in Cloud Computing**

Rajkumar Buyya (The University of Melbourne - Australia),

Dilma da Silva (IBM – USA)

Orran Krieger (vMWare – USA)

Dan Reed (Microsoft – USA)

**Abstract:** Cloud computing has the potential to fundamentally change the Information Technology sector by transferring most computing to third-party remote providers. Cloud computing leverages high wide-area network bandwidth, virtualization, statistical multiplexing of resources, and large-scale data centers to deliver computing resources and applications at very low cost. However, the promise of widespread cloud computing faces many research challenges, including improving performance and security isolation between applications and services, improving storage scalability, and enabling performance monitoring of applications and services. The main purpose of this panel is to discuss these and other challenges, while highlighting the research opportunities that they prompt.

# SBAC Tutorials

**SBAC Tutorial 1****Day: Wednesday, 28<sup>th</sup> – 13:30 – Building T, Room 507**

- **Cloud Computing: Vision, Tools, and Technologies for Delivering Computing as the 5th Utility**

Rajkumar Buyya

**Abstract:** Computing is being transformed to a model consisting of services that are commoditised and delivered in a manner similar to utilities such as water, electricity, gas, and telephony. In such a model, users access services based on their requirements without regard to where the services are hosted. Several computing paradigms have promised to deliver this utility computing vision and they include Grid computing, P2P computing, and more recently Cloud computing. The latter term denotes the infrastructure as a “Cloud” in which businesses and users are able to access applications from anywhere in the world on demand. Cloud computing delivers infrastructure, platform, and software (application) as services, which are made available as subscription-based services in a pay-as-you-go model to consumers. These services in industry are respectively referred to as Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). To realize Cloud computing, vendors such as Amazon, HP, IBM, and Sun are starting to create and deploy Clouds in various locations around the world. In addition, companies with global operations require faster response time, and thus save time by distributing workload requests to multiple Clouds in various locations at the same time. This creates the need for establishing a computing atmosphere for dynamically interconnecting and provisioning Clouds from multiple domains within and across enterprises. There are many challenges involved in creating such Clouds and Cloud interconnections.

**SBAC Tutorial 2****Day: Saturday, 31<sup>th</sup> – 14:50 – Building 40, Room 104**

- **Visualization for Performance Debugging of Large-Scale Parallel Applications**

Lucas Mello Schnorr

Benhur de Oliveira Stein

Guillaume Huard

Jean-Marc Vincent

**Abstract:** Performance analysis of parallel and distributed applications has been used for several years to optimize the code and improve resources utilization. Despite the strong research effort spent during these years to develop new mechanisms to enhance the analysis task, new challenges for the analysis of these applications are still present. These challenges are mainly due to the appearance of new concepts, such as Grid Computing and more recently Cloud Computing, but also the availability of new infrastructures, such as Large-Scale Clusters, composed by thousands of multi-core processing units. Within this tutorial, we will present the main aspects related to the performance debugging of parallel applications. We give special attention to the impact of large-scale aspects on the techniques that are commonly used, from the tracing mechanisms to the analysis through visualization. Since the tutorial is addressing one of the most interesting tasks of high performance computing and the challenges of large-scale application analysis, it demonstrates an extremely high degree of relevance and addresses a broad spectrum of potential attendees, from computer architecture to high-performance parallel areas.

The tutorial is divided in four parts: the definition of common problems found by parallel application developers; the common methodology applied to understand these problems; the implementation of this methodology in grid and cluster environments; and a demonstration of real-case analysis, from trace to the visualization.

The first part characterizes the main problems that parallel application developers must face in order to optimize their code. Problems like computing and network bottleneck identification, low resource utilization leading to load imbalances among processes and low performance in large-scale situations are defined. The second part presents the methodology and theoretical approaches to detect these problems, through execution tracing and visualization analysis. The third part presents implemented tools and techniques that are used to analyze performance problems, showing examples of visualization tools. The fourth part presents a demonstration of a real-world large-scale parallel application executed on a Grid and also the analysis of synthetic traces that illustrate common problems found when doing a performance debugging

## SBAC-PAD – Technical Sessions

### SBAC-PAD – Technical Session I – Accelerators and GPU

Day: Thursday, 29<sup>th</sup> – 14:00 – Building T, Room 506

- **LALP: A Novel Language to Program Custom FPGA-Based Architectures**, by Ricardo Menotti, João M.P. Cardoso, Marcio M. Fernandes, and Eduardo Marques  
*Field-Programmable Gate Arrays (FPGAs) are becoming increasingly important in embedded and highperformance computing systems. They allow performance levels close to the ones obtained from Application-Specific Integrated Circuits (ASICs), while still keeping design and implementation flexibility. However, to efficiently program FPGAs, one needs the expertise of hardware developers and to master hardware description languages (HDLs) such as VHDL or Verilog. The attempts to furnish a high-level compilation flow (e.g., from C programs) still have open issues before efficient and consistent results can be obtained. Bearing in mind the FPGA resources, we have developed LALP, a novel language to program FPGAs. A compilation framework including mapping capabilities supports the language. The main ideas behind LALP is to provide a higher abstraction level than HDLs, to exploit the intrinsic parallelism of hardware resources, and to permit the programmer to control execution stages whenever the compiler techniques are unable to generate efficient implementations. In his paper we describe LALP, and show how it can be used to achieve high-performance computing solutions.*
- **Profiling General Purpose GPU Applications**, by Bruno Rocha Coutinho, George Luiz Medeiros Teodoro, Rafael Sachetto Oliveira, Dorgival Olavo Guedes Neto, and Renato Antonio Celso Ferreira  
*We are witnessing an increasing adoption of GPUs for performing general purpose computation, which is usually known as GPGPU. The main challenge in developing such applications is that they often do not fit in the model required by the graphics processing devices, limiting the scope of applications that may benefit from the computing power provided by GPUs. With CUDA, the graphics community has provided a general purpose framework for developing applications which can be effectively executed on the underlying graphics hardware, however, the model mismatch is still there, and it is perceived mainly by a limited performance gain. In this work we propose a profiling strategy for GPGPU applications. This strategy is based on performance predicates and is able to quantify the major sources of performance degradation while providing hints on how to improve the applications. We applied our strategy to CUDA-based programs and were able to understand and improve their performance.*
- **Massively Parallel Nearest Neighbor Queries for Dynamic Point Clouds on the GPU**, by Pedro José Silva Leite, João Marcelo Xavier Natário Teixeira, Thiago Souto Maior Cordeiro de Farias, Veronica Teichrieb, and Judith Kelner  
*We introduce a parallel algorithm to solve approximate and exact nearest neighbor queries on the GPU, exploiting its massively parallel processing power. Both data structure construction and nearest neighbor queries are performed on the GPU,*

*avoiding memory copies from system memory to device memory. This algorithm achieves real-time performance, enabling its usage in dynamic scenarios, by minimizing the sorting comparisons needed for a large  $K$  value. The underlying data structure for spatial subdivision handles 3D points and is based on grid spatial hashing. Users can specify the grid size interactively. Comparisons were done with other nearest neighbor algorithms implemented on both CPU and GPU. Our approach clearly surpasses CPU implementations regarding processing time, while it presents a competitive solution to GPU ones. Real-time results were obtained with ANN searches ( $K = 10$ ) for data sets up to 163K points and the potential of our algorithm is demonstrated through a point-based rendering application.*

- **Accelerating Kirchhoff Migration by CPU and GPU Cooperation**, by Jairo Panetta, Thiago Teixeira, Paulo R.P. de Souza Filho, Carlos A. da Cunha Finho, David Sotelo, Fernando M. Roxo da Motta, Silvio Sinedino Pinheiro, Ivan Pedrosa Junior, Andre L. Romanelli Rosa, Luiz R. Monnerat, Leandro T. Carneiro, and Carlos H.B. de Albrecht

*We discuss the performance of Petrobras production Kirchhoff prestack seismic migration on a cluster of 64 GPUs and 256 CPU cores. Porting and optimization of the application hot spot (98.2% of a single CPU core execution time) to a single GPU reduces total execution time by a factor of 36 on a control run. We then argue against the usual practice of porting the next hot spot (1.5% of single CPU core execution time) to the GPU. Instead, we show that cooperation of CPU and GPU reduces total execution time by a factor of 59 on the same control run. Remaining GPU idle cycles are eliminated by overloading the GPU with multiple requests originated from distinct CPU cores. However, increasing the number of CPU cores in the computation reduces the gain due to the combination of enhanced parallelism in the runs without GPUs and GPU saturation on runs with GPUs. We proceed by obtaining close to perfect speed-up on the full cluster over homogeneous load obtained by replicating control run data. To cope with the heterogeneous load of real world data we show a dynamic load balancing scheme that reduces total execution time by a factor of 20 on runs that use all GPUs and half of the cluster CPU cores with respect to runs that use all CPU cores but no GPU.*

<b>SBAC-PAD – Technical Session II - Multiprocessors</b> <b>Day: Thursday, 29<sup>th</sup> – 16:05 – Building T, Room 506</b>
--

- **Parallel LDPC Decoding on a Network-on-Chip Based Multiprocessor Platform**, by Wen-Hsiang Hu, Jun Ho Bahn, and Nader Bagherzadeh  
*Low Density Parity Check (LDPC) code is an error correction code that can achieve performance close to Shannon limit and inherently suitable for parallel implementation. It has been widely adopted in various communication standards such as DVB-S2, WiMAX, and Wi-Fi. However, the irregular message exchange pattern is a major challenge in LDPC decoder implementation. In addition, faced with an era that diverse applications are integrated in a single system, a flexible, scalable, efficient and cost-effective implementation of LDPC decoder is highly preferable. In this paper, we proposed a multi-processor platform based on network-on-chip (NoC) interconnect as a solution to these problems. By using a distributed and cooperative way for LDPC decoding, the memory bottleneck commonly seen in LDPC decoder design is eliminated. Simulation results from long LDPC codes with various code rates show good scalability and speedups are obtained by our approach.*
- **kD-Tree Traversal Implementations for Ray Tracing on Massive Multiprocessors: A Comparative Study**, by Artur L. dos Santos, João Marcelo X.N. Teixeira, Thiago S.M.C. de Farias, Veronica Teichrieb, and Judith Kelner  
*Current GPU computational power enables the execution of complex and parallel algorithms, such as Ray Tracing techniques supported by kD-trees for 3D scene rendering in real time. This work describes in detail the study and implementation of five different kD-Tree traversal algorithms using the parallel framework NVIDIA Compute Unified Device Architecture (CUDA), in order to point their pros and cons regarding adaptation capability to the chosen architecture. In addition, a new algorithm is proposed by the authors based on this analysis, aiming performance improvement. A performance analysis of the implemented techniques demonstrates that two of these algorithms, once adequately adapted to CUDA architecture, are capable of reaching speedup gains up to 15x when compared to former CPU implementations and up to 4x in comparison to existing and optimized parallel ones. As a consequence, interactive frame rates are possible for scenes with 1376 x 768 pixels of resolution and 1 million primitives.*
- **Analysis of Performance Dependencies in NUCA-Based CMP Systems**, by Pierfrancesco Foglia, Francesco Panicucci, Cosimo Antonio Prete, and Marco Solinas  
*Improvements in semiconductor nanotechnology have continuously provided a crescent number of faster and smaller per-chip transistors. Consequent classical techniques for boosting performance, such as the increase of clock frequency and the amount of work performed at each clock cycle, can no longer deliver to significant improvement due to energy constraints and wire delay effects. As a consequence, designers interests have shifted toward the implementation of systems with multiple cores per chip (Chip Multiprocessors, CMP). CMP systems typically*



*adopt a large last-level-cache (LLC) shared among all cores, and private L1 caches. As the miss resolution time for private caches depends on the response time of the LLC, which is wire-delay dominated, performance are affected by wire delay. NUCA caches have been proposed for single and multi core systems as a mechanism for such tolerating wire-delay effects on the overall performance.*

*In this paper, we introduce our design for S-NUCA and D-NUCA cache memory systems, and we present an analysis of an 8-cpu CMP system with two levels of cache, in which the L1s are private, while the L2 is a NUCA shared among all cores. We considered two different system topologies (the first with the eight cpus connected to the NUCA at the same side -8p-, the second with half of the cpus on one side and the others at the opposite side -4+4p), and for all the configurations we evaluate the effectiveness of both the static and dynamic policies that have been proposed. Our results show that adopting a D-NUCA scheme with the 8p configuration is the best performing solution among all the considered configurations, and that for the 4+4p configuration the D-NUCA outperforms the S-NUCA in most of the cases. We highlight that performance are tied to both mapping strategy variations (Static and Dynamic) and topology changes. We also observe that bandwidth occupancy depends on both the NUCA policy and topology.*

<b>SBAC-PAD – Technical Session III – Scheduling and Data Allocation</b> <b>Day: Friday, 30<sup>th</sup> – 14:00 – Building T, Room 506</b>
--

- **Memory Affinity for Hierarchical Shared Memory Multiprocessors**, by Christiane Pousa Ribeiro, Jean-François Méhaut, Alexandre Carissimi, Márcio Castro, and Luiz Gustavo Fernandes  
*Currently, parallel platforms based on large scale hierarchical shared memory multiprocessors with Non-Uniform Memory Access (NUMA) are becoming a trend in scientific High Performance Computing (HPC). Due to their memory access constraints, these platforms require a very careful data distribution. Many solutions were proposed to resolve this issue. However, most of these solutions did not include optimizations for numerical scientific data (array data structures) and portability issues. Besides, these solutions provide a restrict set of memory policies to deal with data placement. In this paper, we describe an user-level interface named Memory Affinity interface (MAi)I, which allows memory affinity control on Linux based cache-coherent NUMA (ccNUMA) platforms. Its main goals are, fine data control, flexibility and portability. The performance of MAi is evaluated on three ccNUMA platforms using numerical scientific HPC applications, the NAS Parallel Benchmarks and a Geophysics application. The results show important gains (up to 31%) when compared to Linux default solution.*
- **Thread to Core Assignment in SMT On-Chip Multiprocessors**, by Carmelo Acosta, Francisco J. Cazorla, Alex Ramirez, and Mateo Valero  
*State-of-the-art high-performance processors like the IBM POWER5 and Intel i7 show a trend in industry towards on-chip Multiprocessors (CMP) involving Simultaneous Multithreading (SMT) in each core. In these processors, the way in which applications are assigned to cores plays a key role in the performance of each application and the overall system performance.*  
*In this paper we show that the system throughput highly depends on the Thread to Core Assignment (TCA), regardless the SMT Instruction Fetch (IFetch) Policy implemented in the cores. Our results indicate that a good TCA can improve the results of any underlying IFetch Policy, yielding speedups of up to 28%.*  
*Given the relevance of TCA, we propose an algorithm to manage it in CMP + SMT processors. The proposed throughput-oriented TCA Algorithm takes into account the workload characteristics and the underlying SMT IFetch Policy. Our results show that the TCA Algorithm obtains thread-to-core assignments 3% close to the optimal assignation for each case, yielding system throughput improvements up to 21%.*
- **An Adaptive Mechanism for Fair Sharing of Storage Resources**, by Chao Jin and Rajkumar Buyya  
*To ensure Quality of Service (QoS) for data centers, it is critical to enforce a fair share of storage resources between competing users. Interposed schedulers are one of the most practical methods for performance isolation. Most fair queuing-based proportional sharing algorithms for existing interposed scheduler are variants of counterparts designed for network routers and may result in breaking the fairness*

*of proportional sharing required by Service Level Agreements for storage systems. This paper presents a novel algorithm to address this problem. As an extension of the fair queuingbased algorithm, it can dynamically adapt to the performance variation of storage systems and guarantee a fair sharing of resources as well as satisfying the minimal performance requirements for different clients. The design and performance evaluation are presented.*

- **Exploiting Computational Resources in Distributed Heterogeneous Platforms**, by George Teodoro, Rafael Sachetto, Daniel Fireman, Dorgival Guedes, and Renato Ferreira

*We have been witnessing a continuous growth of both heterogeneous computational platforms (e.g., Cell blades, or the joint use of traditional CPUs and GPUs) and multi-core processor architecture; and it is still an open question how applications can fully exploit such computational potential efficiently. In this paper we introduce a run-time environment and programming framework which supports the implementation of scalable and efficient parallel applications in such heterogeneous, distributed environments. We assess these issues through well-known kernels and actual applications that behave regularly and irregularly, which are not only relevant but also demanding in terms of computation and I/O. Moreover, the irregularity of these, as well as many other applications poses a challenge to the design and implementation of efficient parallel algorithms. Our experimental environment includes dual and octa-core machines augmented with GPUs and we evaluate our framework performance for standalone and distributed executions. The evaluation on a distributed environment has shown near to linear scale-ups for two data mining applications, while the applications performance, when using CPU and GPU, has been improved into around 25%, compared to the GPU-only versions.*

<b>SBAC-PAD – Technical Session IV – Complex Applications</b> <b>Day: Friday, 30<sup>th</sup> – 16:10 – Building T, Room 506</b>
---

- **Irregular Grid Raycasting Implementation on the Cell Broadband Engine**, by Guilherme Cox, André Máximo, Cristiana Bentes, and Ricardo Farias  
*Direct volume rendering has become a popular technique for visualizing volumetric data from sources such as scientific simulations, analytic functions, medical scanners, among others. Volume rendering algorithms, such as raycasting, can produce high-quality images, however, the use of raycasting has been limited due to its high demands on computational power and memory bandwidth. In this paper, we propose a new implementation of the raycasting algorithm that takes advantage of the highly parallel architecture of the Cell Broadband Engine processor, with 9 heterogeneous cores, in order to allow efficient raycasting of irregular datasets. All the computational power of the Cell BE processor, though, comes at the cost of a different programming model. Applications need to be rewritten, which requires using multithreading and vectorized code. In our approach, we tackle this problem by distributing ray computations using the visible faces, and vectorizing the lighting integral operations inside each core. Our experimental results show that we can obtain good speedups reducing the overall rendering time significantly.*
- **Design and Implementation of Brain Fiber Tracking for GPUs and PC Clusters**, by Adiel Mittmann, Mario A.R. Dantas, and Aldo von Wangenheim  
*Fiber tracking is a technique that determines the location of fiber bundles in a human brain. Sequential implementations of fiber tracking usually require a large execution time, which hinders both the interactivity currently experienced by physicians and its applications to large collections of data sets. In this paper we present the design and implementation of fiber tracking applications in two different parallel environments, GPUs and PC clusters. The performance results obtained in our experiments show that both HPC configurations are suitable for fiber tracking: GPUs for interactive applications targeted at physicians and PC clusters for batch processing by fiber tracking researchers.*
- **Finite Element Nodal Approach for Parallel Processing of Non Linear Shell Analysis**, by Célia Leiko Ogawa Kawabata, Masaki Kawabata Neto, Humberto Breves Coda, and Wilson Sergio Venturini  
*This study presents a new parallel Finite Element Method (FEM) strategy designed for coarse grain distributed memory systems. The adopted communication protocol is Message Passing Interface (MPI) and tests are carried out in a cluster of PCs. Compressed data structure is used to store the Hessian matrix in order to optimize memory usage and to use the parallel direct solver MUMPS. The new partitioning paradigm is based on structural finite element nodes, not elements (as usually done in references), resulting in an overlapping algorithm, where a reduced amount of information should be allocated and manipulated to integrate finite elements. The main advantage of the nodal partitioning is the performance improvement of the Hessian matrix assembly and the natural ordering to improve the system solution.*

*Numerical examples are shown in order to demonstrate the efficiency and scalability of the proposed algorithm.*

**SBAC-PAD – Technical Session V – Performance Analysis Tools and Tool-less Systems**

**Day: Saturday, 31<sup>st</sup> – 09:00 – Building 40, Room 109**

- **A Paradigm Change: From Performance Monitoring to Performance Analysis**, by Luiz DeRose and Heidi Poxon  
*The purpose of an application performance tool is to help users identify whether or not their application are running efficiently on the computing resources available. However, the scale of current and future high end systems, as well as increasing system software and architecture complexity, brings a new set of challenges for performance tuning. To achieve high performance on peta-scale systems, users need a new infrastructure for performance measurement that can automate the process of performance analysis. In order to address these issues, we designed an automatic performance analysis framework, which contains innovative techniques that use a knowledge base and set of performance models. In this paper we outline the key aspects of this design and present the features from this design that have been implemented in the Cray Performance Analysis Toolset.*
- **Visual Mapping of Program Components to Resources Representation: A 3D Analysis of Grid Parallel Applications**, by Lucas Mello Schnorr, Philippe Olivier Alexandre Navaux, and Guillaume Huard  
*Highly distributed systems such as Grids are usually interconnected by a hierarchical organization of different types of network. This strong network hierarchy influences directly the behavior of parallel applications. In order to obtain a good understanding of parallel application's behavior, the performance analysis must take into account a correspondence between application and network characteristics. This paper presents a novel way to analyze parallel applications, by using a three dimensional visualization and a technique to visually map the application's components to the used resources. This mapping technique is able to handle different types of resources description: different system logical organizations and the description of the network or system interconnection in different levels. The technique, implemented in our prototype Triva, is evaluated through a series of visual representations of the monitoring data obtained through real executions of parallel applications in a grid. The resulting visualizations enable an alternative and powerful way to view and understand various aspects of application behavior together with the network topology.*
- **Performance and Energy Consumption Evaluation of Embedded Applications: A Method Based on Platform's Behavioral Model**, by Bruno Costa e Silva Nogueira, Paulo Romero Martins Maciel, Eduardo Antônio Guimarães Tavares, Ermeson Carneiro, Gustavo Callou, Ricardo Massa Oliveira Lima, Rodolfo Ferraz, and Bruno Montenegro  
*This paper presents a performance and energy consumption modeling technique for embedded systems. The proposed method adopts a formal model based on Coloured Petri Nets for modeling the functional behavior of processors and memory architectures at a high-level of abstraction. The applicability of the proposed*

*method is illustrated by evaluating a set of applications and a general-purpose microcontroller. Experimental results demonstrate an average accuracy of 96% in comparison with the respective measures acquired from the real hardware platform. Moreover, the high-level behavioral representation of platforms allows the rapid analysis of performance and energy consumption of complex systems.*

- **The Performance of a Bare Machine Email Server**, by George H. Ford Jr., Ramesh K. Karne, Alexander L. Wijesinha, and Patrick Appiah-Kubi  
*Bare machine applications run directly over the hardware without using an operating system or a hard disk. This paper studies the performance of a bare machine email server whose design and implementation is based on several novel architectural features with a view towards optimizing performance. The results are compared with those for the AxiGen and ShareMailPro email servers, and a lean Javabased email server prototype running on Windows whose application-level operation closely matches that of the bare machine email server. For 80,000 emails in a LAN environment, the bare Machine server processing time is approximately 2 times faster than a Java-based server, and 2.4 times faster than the AxiGen server. For 5,500 emails in a WAN environment, the bare machine server performed at least 1.8 times faster than the Java-based and ShareMailPro servers. The results indicate that the bare machine email server outperforms the conventional email servers in LAN and WAN environments, and demonstrate the capability of using bare machines to build high-performance email servers.*

**SBAC-PAD – Technical Session VI – Processor Architecture****Day: Saturday, 31<sup>st</sup> – 11:10 – Building 40, Room 109**

- **TMT - A TLB Tag Management Framework for Virtualized Platforms**, by Girish Venkatasubramanian, Renato J. Figueiredo, Ramesh Illikkal, and Donald Newell

*The rise in multi-core architectures has led to the abundance of computing resources on a chip. Virtualization has emerged as a way to efficiently partition and share these resources. Thus, the emphasis in micro-architecture design, especially in x86, has shifted towards providing hardware support for better performance of VMs on bare metal. One of the areas of focus for these efforts is the Translation Lookaside Buffer (TLB). Recent modifications in the TLB include the addition of tags as a part of the TLB entry and the incorporation of hardware primitives to perform tag comparison during TLB lookup.*

*In this paper we present the Tag Manager Table (TMT), a low-latency management architecture for tagging the TLB entries using process-specific identifiers (based on the CR3 register in x86), and thereby reducing the number of flushes and the miss rate in the TLB. Using a full system simulation approach, we investigate the performance benefit of these tags and explore how it varies with the size of the TMT, the TLB architecture and the workload characteristics. We also perform a sensitivity analysis and quantify the relative importance of all these factors in determining the benefit from CR3 tagging. While our focus is on virtualized platforms, this approach is equally applicable for non virtualized environments.*

- **Composite Confidence Estimators for Enhanced Speculation Control**, by Daniel A. Jiménez

*This paper proposes a way to allow more effective use of speculation control techniques by combining multiple confidence estimators into a composite confidence estimator. This new class of confidence estimators provides improved performance and finer speculation control. This paper makes three contributions. First, we describe techniques for building efficient composite confidence estimators. Second, we present an improved statistical methodology for evaluating confidence estimators. Finally, we use a detailed microarchitectural simulator to evaluate the ability of our estimator to support an energy reduction technique called pipeline gating. Using previous confidence estimators, pipeline gating reduces the amount of extra work due to mis-speculated instructions by 22%, with a reduction in IPC of 5%. With the same impact on IPC, our confidence estimators reduce extra work by 31%.*

- **SPARC16: A New Compression Approach for the SPARC Architecture**, by Leonardo Luiz Ecco, Bruno Cardoso Lopes, Eduardo Candido Xavier, Ricardo Pannain, Paulo Centoducatte, and Rodolfo Jardim de Azevedo

*RISC processors can be used to face the ever increasing demand for performance required by embedded systems. Nevertheless, this solution comes with the cost of poor code density. Alternative encodings for instruction sets, such as MIPS16 and*



*Thumb, represent an effective approach to deal with this drawback. This article proposes to apply a new encoding to the SPARCv8 architecture. Through extensive analysis of a program mix from the Mibench and Mediabench benchmark suites, we suggest a new 16-bit instruction set, easily translated to its 32-bit counterpart during execution time. Using the aforementioned program mix to infer how code could be represented in the proposed 16-bit ISA, compression ratios as low as 56% can be obtained. We also evaluated the cache behavior and showed reductions of 42% on cache misses that can increase performance up to 28% (for patricia program with 2KB cache).*

## WSCAD – Sessões Técnicas

**WSCAD – Sessão Técnica I – Escalonamento e Balanceamento de Carga**

**Dia: Quinta-feira, 29 – 14:00 – Prédio T, Sala 507**

- **Escalonamento Realimentado para Diferenciação de Serviços e Garantia de Desempenho em ambientes SOA com requisitos Soft-RT**, por Priscila Tiemi Maeda Saito, USP, Pedro Northon Nobile, USP, Francisco Monaco, ICMC/USP.  
*Trabalhos abordando provisão de QoS em nível de aplicação têm recebido crescente atenção. Diversas técnicas de escalonamento têm sido propostas objetivando garantias relativas ou absolutas de responsividade. No primeiro caso, investigam-se algoritmos para diferenciação de serviços baseados em atendimento preferencial à classes de serviço com distintas prioridades; no segundo, pretende-se oferecer garantias de desempenho especificadas para cada classe independentemente uma das outras. A integração de QoS relativa e absoluta não tem sido explorada da mesma forma. Este artigo apresenta uma estratégia de escalonamento realimentado capaz de atender a requisitos de QoS formulados em termos de limites superiores para o tempo médio de resposta das requisições, além de especificar que determinadas classes de usuários terão prioridade sobre outras.*
- **Melhorando o Desempenho de Algoritmos do Tipo Branch & Bound em MPI via Escalonador com Roubo Aleatório de Tarefas**, por Stéfano Mor, UFRGS, Nicolas Maillard, II-UFRGS.  
*Nossa principal contribuição é a integração de um modelo de escalonamento distribuído por roubo de tarefas para computação em MPI capaz de otimizar o desempenho de programas do tipo Branch & Bound. Esse escalondor é introduzido em tempo de compilação e é independente da distribuição MPI usada. Resultados experimentais mostram que se pode obter um ganho de até 80% no desempenho, mantendo o speedup próximo ao linear e sem a perda do consumo linear de memória. Esses ganhos se confirmam mesmo em um ambiente de processadores homogêneos, que tendem a produzir um menor desbalanceamento da carga de trabalho.*
- **Avaliação de Estratégias de Balanceamento de Carga do Tipo Mestre-Escravo para Aplicações SPMD em Clusters e Grids Computacionais**, por André de Oliveira, UFRJ, Alexandre Plastino, Simone Martins, UFF, Ines Dutra, Universidade do Porto, Portugal.  
*O desempenho de programas paralelos é fortemente influenciado por diferentes fatores dinâmicos de desequilíbrio de carga. A utilização de um algoritmo de balanceamento de carga adequado é essencial para a redução dos efeitos causados por esses fatores de desequilíbrio. Neste trabalho, avalia-se o desempenho de diversas estratégias de balanceamento quando executadas em aplicações SPMD com diferentes características. Os experimentos foram realizados em um cluster e em um grid computacional e foram considerados diversos fatores que podem ocasionar desequilíbrios em um ambiente computacional, tais como: capacidade de*

*processamento, presença de carga externa à aplicação e velocidade de comunicação.*

- **Escalabilidade de aplicação operacional em ambiente massivamente paralelo**, por Alvaro Fazenda, CPTEC/INPE, Jairo Panetta, CPTEC/INPE, Philippe Navaux, UFRGS, Luiz Flavio Rodrigues, Daniel Massaru Katsurayama, Luis Filipe Motta, CPTEC/INPE.

*A demanda constante para melhorar a qualidade de previsões numéricas do tempo obriga o uso de computadores progressivamente mais potentes. Com a popularização de processadores “multicore”, o uso de sistemas com muitas centenas de processadores tornou-se economicamente viável. Este trabalho investiga como escalar o paralelismo de uma aplicação operacional para a previsão do tempo que executa eficientemente em muitas dezenas de processadores para máquinas com muitas centenas de processadores. A investigação determinou as limitações da aplicação, suas surpreendentes causas e permitiu desenvolver uma solução eficiente que atinge a escala de paralelismo desejado.*

<b>WSCAD – Sessão Técnica II – Avaliação de Desempenho e Software Básico</b> <b>Dia: Quinta-feira, 29– 16:05 – Prédio T, Sala 507</b>
--

- **MPI Broadcast com Compressão de Números de Ponto Flutuante**, por Jose Camata, UFRJ, Renato Elias, UFRRJ, Alvaro Coutinho, COPPE/UFRJ.  
*Aplicações científicas desenvolvidas para sistemas paralelos de memória distribuída consomem parte do seu tempo total de execução trocando dados entre processos. Portanto, aprimorar o desempenho das rotinas responsáveis pela comunicação vem ganhando cada vez mais importância. Neste contexto, este trabalho investiga a utilização de um algoritmo de compressão de ponto-flutuante na transmissão de mensagens longas. Este algoritmo foi implementado na primitiva broadcast do MPI e foram efetuadas medições de desempenho para diferentes tipos de mensagens em até 512 núcleos de processamento. Os resultados obtidos demonstram que a compressão pode acelerar significativamente o broadcast padrão do MPI.*
- **Experimentos com Gerenciamento de Contenção em uma Memória Transacional com Suporte em Software**, por Fernando Kronbauer, UNICAMP, Sandro Rigo, UNICAMP.  
*Devido à grande disseminação recente de arquiteturas paralelas, mais e mais programadores são expostos aos problemas relacionados ao uso dos mecanismos tradicionais de controle de concorrência. Memórias transacionais têm sido propostas como um meio de aliviar as dificuldades encontradas ao escreverem-se programas paralelos. Neste trabalho exploramos um sistema de memória transacional em software (STM), apresentando uma abordagem nova para gerenciar a contenção entre transações, que leva em consideração os padrões de acesso aos diferentes dados de um programa ao escolher o gerenciador de contenção usado para o acesso a estes dados. Elaboramos uma modificação da plataforma de STM que nos permite realizar esta associação entre dados e gerenciamento de contenção, e realizamos uma caracterização baseada nos padrões de acesso aos dados de um programa executando em diferentes sistemas de computação.*
- **Arquitetura e Avaliação do Cluster de Alto Desempenho Netuno**, por Vinicius Silva, UFRJ, Sergio Guedes, UFRJ, Cristiana Bentes, UERJ, Gabriel P. Silva, UFRJ.  
*Este artigo apresenta a arquitetura e os resultados da avaliação de desempenho do supercomputador Netuno, um cluster de alto desempenho recentemente instalado na UFRJ. São apresentados detalhes tanto de sua arquitetura como dos softwares básicos e de middleware utilizados na sua construção. Os resultados de avaliação obtidos registram um desempenho de 16,2 TFlops sustentados para o benchmark HPL (High Performance Linpack), o que colocou o supercomputador Netuno na 138ª posição na lista Top500 de junho de 2008. Atualmente, o supercomputador Netuno atende diversas instituições de ensino e pesquisa no Brasil, participantes*

*das redes temáticas de pesquisa de Geofísica Aplicada e de Oceanografia (REMO), patrocinadas pela Petrobras.*

- **Trebuchet: Explorando TLP com Virtualização DataFlow**, por Tiago Alves, COPPE/UFRJ, Leandro Marzulo, COPPE/UFRJ, Felipe França, COPPE/UFRJ, Vítor Santos Costa, Universidade do Porto, Portugal.

*No modelo DataFlow as instruções são executadas tão logo seus operandos de entrada estejam disponíveis, expondo, de forma natural, o paralelismo em nível de instrução (ILP). Por outro lado, a exploração de paralelismo em nível de thread (TLP) passa a ser também um fator de grande importância para o aumento de desempenho na execução de uma aplicação em máquinas multicore. Este trabalho propõe um modelo de execução de programas, baseado nas arquiteturas DataFlow, que transforma ILP em TLP. Esse modelo é demonstrado através da implementação de uma máquina virtual multi-threaded, a Trebuchet. A aplicação é compilada para o modelo DataFlow e suas instruções independentes (segundo o fluxo de dados) são executadas em Elementos de Processamento (EPs) distintos da Trebuchet. Cada EP é mapeado em uma thread na máquina hospedeira. O modelo permite a definição de blocos de instruções de diferentes granularidades, que terão disparo guiado pelo fluxo de dados e execução direta na máquina hospedeira, para diminuir os custos de interpretação. Como a sincronização é obtida pelo modelo DataFlow, não é necessária a introdução de locks ou barreiras nos programas a serem paralelizados. Um conjunto de três benchmarks reduzidos, compilados em oito threads e executados por um processador quadcore Intel Core™ i7 920, permitiu avaliar: (i) o funcionamento do modelo; (ii) a versatilidade na definição de instruções com diferentes granularidades (blocos); (iii) uma comparação com o OpenMP. Acelerações de 4,81, 2,4 e 4,03 foram atingidas em relação à versão sequencial, enquanto que acelerações de 1,11, 1,3 e 1,0 foram obtidas em relação ao OpenMP.*

**WSCAD – Sessão Técnica III – Aplicações de Alto Desempenho I**  
**Dia: Sexta-Feira, 30 – 14:00 – Prédio T, Sala 507**

- **Automação de Refatorações para Programas Fortran de Alto Desempenho**, por Bruno Boniati, UFSM, Andrea Charao, Benhur Stein, UFSM.

*Refatoração é uma técnica de engenharia de software que objetiva aplicar melhorias internas no código-fonte de aplicações, sem que isso influencie no seu comportamento. é uma técnica amplamente empregada em código orientado a objetos e utilizada por algumas metodologias de desenvolvimento. Na computação de alto desempenho, a refatoração é uma técnica pouco explorada, sendo que grande parte código legado de programas de alto desempenho está escrita em linguagens não orientadas a objetos. Neste trabalho, explora-se a automatização de técnicas de refatoração e sua aplicação em códigos de alto desempenho escritos em linguagem imperativa Fortran. São identificadas oportunidades de melhorar a legibilidade e o design do código sem afetar o desempenho, além de reestruturações de código quem podem representar ganho de desempenho. As técnicas estudadas e automatizadas são utilizadas em aplicações reais, como meio de validar a abordagem. A automatização das técnicas é feita estendendo as funcionalidades do IDE Photran, um plugin para Eclipse, cujo funcionamento também será abortado neste texto.*
- **Uma abordagem de alto desempenho para multiplicação de matrizes densas em sistemas Reconfiguráveis**, por Viviane Souza, Victor Medeiros, Derci Lima, UFPE, Manoel Lima, UFPE.

*A demanda por máquinas de alto desempenho e por novas estratégias que buscam melhorar o processamento de dados em aplicações de computação científica tem crescido muito nos últimos anos. Algumas novas arquiteturas baseadas em GPU, processadores Cell e FPGA ou ainda plataformas híbridas aparecem como soluções para esses problemas.*

*Neste trabalho nós apresentamos uma arquitetura de alto desempenho para implementação de multiplicação de matrizes densas em uma plataforma comercial híbrida, o RASC (Reconfigurable Application-Specific Computing). O RASC foi desenvolvido pela Silicon Graphics e consiste em uma plataforma composta por um processador de propósito geral acoplado a co-processadores baseados em FPGA.*

*A arquitetura proposta investiga como a solução do problema de multiplicação de matrizes pode tirar proveito das características de uma plataforma com alto grau de paralelismo. Nós também investigamos a escalabilidade do algoritmo e os mecanismos de reuso de dados. Baseado nessas investigações um estudo de caso é sugerido e discutido me detalhes.*
- **Análise do processamento paralelo em Clusters multicore na simulação de escoamento miscível implementado pelo método dos elementos finitos**, por Adriana Barbosa, Lucia Catabriga, Alberto De Souza, Andrea Valli, UFES.

*Neste trabalho analisamos o desempenho paralelo de um código de elementos finitos em Clusters multi-core considerando duas alternativas de particionamento do job paralelo MPI entre os cores: single-core, na qual apenas um processo é*

*enviado para cada máquina multi-core; e fill-up, na qual todos os cores de cada máquina do Cluster recebem processos. A aplicação envolve o escoamento miscível de fluido incompressível em meio poroso, mais especificamente, a simulação de traçadores no contexto da Engenharia de Petróleo. Nós medimos o tempo de processamento e o speedup obtidos em dois Clusters de computadores multi-core: um composto por 31 máquinas, cada uma com uma Unidade Central de Processamento (Central Processing Unit - CPU) dual-core (Enterprise 2); e outro composto por 29 máquinas, cada uma com uma CPU quad-core (Enterprise 3). Além disso, analisamos a eficiência dos algoritmos de escalonamento no Enterprise 3 em termos da comunicação externa entre as CPUs e a concorrência dos cores pela hierarquia de memória das CPUs. Nossos resultados mostraram que, muito embora máquinas mais modernas como o Enterprise 3 possuam um número maior de cores e estes sejam mais rápidos, a competição destes cores pela hierarquia de memória tem grande impacto no desempenho final das aplicações.*

- **Estudo Quantitativo do Modelo WRF de Previsão do Tempo em um Ambiente de Cluster Multi-core**, por Luiz Carlos Pinto, UFSC, Mario Dantas, UFSC.

*A computação científica demanda poder computacional de alto desempenho principalmente para resolver em tempo hábil problemas conhecidos como “grandes desafios”. Devido à limitação de clock, cada vez mais arquiteturas não-convencionais construídas com processadores conhecidos como commodity estão sendo utilizadas nesta tarefa, como por exemplo os ambientes multi-cluster. Atualmente, a inserção de processadores com múltiplos núcleos nas configurações de cluster cria um cenário diferenciado no que diz respeito à comunicação entre processos paralelos nestes ambientes. Nesse contexto, o presente artigo vem para ampliar a discussão e apontar possibilidades de ganho em desempenho e eficiência. Os resultados empíricos obtidos com a execução do modelo numérico de previsão do tempo WRF (Weather Research and Forecasting Model) revelaram speedup de 1.39 resultante da adequação do subsistema de comunicação entre processos às especificidades da aplicação e do cluster multi-core em foco, o que reforça a importância da análise e a pertinência deste trabalho.*

**WSCAD – Sessão Técnica IV – Arquitetura de Processadores e Hierarquia de Memória**

**Dia: Sexta-feira, 30 – 16:10 – Prédio T, Sala 507**

- **MPSoC Minimalista com Caches Coerentes Implementado num FPGA**, por Roberto Hexsel, Universidade Federal do Paraná, Jorge Junior, UFPR.  
*Este artigo descreve o projeto e a implementação de um MPSoC com caches coerentes num FPGA. O sistema pode ser compilado para conter de 1 a 8 processadores MIPS-I, caches de dados coerentes (L1), unidades de gerenciamento de memória, controladores de memória e um barramento multiplexado. O artigo contém uma descrição detalhada da implementação em VHDL, enfocando o sistema de memória. A inicialização do sistema e a sincronização com semáforos é discutida brevemente. Um programa de testes simples é usado para aferir, preliminarmente, o desempenho do sistema.*
- **Avaliação de Desempenho, Área e Energia de Caches com Controle de Poluição**, por Richard Souza, UFPR, Giancarlo Heck, UFPR, Renato Carmo, UFPR, Roberto Hexsel, UFPR.  
*Este artigo compara o desempenho, a área e o dispêndio de energia de quatro projetos do primeiro nível da hierarquia de memória para sistemas embarcados. Os quatro projetos são: <i>i</i> cache primária com mapeamento direto (MD); <ii> cache primária com associatividade binária; <iii> cache primária com MD associada a uma cache com controle de poluição; e <vi> cache primária com MD associada a uma cache de vítimas e controle de poluição. Os projetos foram simulados com os programas da suíte CommBench, e para estes programas, caches com 4-8 Kbytes e os projetos <iii> e <vi> tem bom desempenho; os sistemas com o projeto <vi> tem a melhor relação desempenho/área e energia.*
- **Estimativa de Consumo de Energia em Nível de Instrução para Processadores Modelados em ArchC**, por Josué Ma, UNICAMP, Rodolfo Azevedo, UNICAMP.  
*A constante redução do tamanho e o conseqüente aumento do número de transistores em um mesmo chip faz com que a potência dissipada pelos circuitos digitais aumente exponencialmente. Esse fato, combinado com a crescente demanda por dispositivos portáteis, têm levado à uma crescente preocupação quanto ao consumo de energia. Quanto mais potência é dissipada mais calor é gerado e mais energia é gasta com o seu resfriamento. Como resultado, projetistas estão considerando cada vez mais o impacto do consumo em suas decisões. Atualmente, Linguagens de Descrição de Arquiteturas (ADLs) têm sido utilizadas para projetar novos processadores. Essas linguagens descrevem o comportamento da arquitetura para cada ação ou instrução. ADLs, além de diminuírem o tempo de projeto, são úteis para descobrir problemas arquiteturais em um nível mais elevado. Nesse trabalho, foi desenvolvida uma ferramenta de estimativa de consumo de energia em nível de instrução utilizando-se como base a ADL ArchC e, como estudo de caso, um processador (ISA) SPARCv8. Como resultado do uso da ferramenta desenvolvida, uma simulação de um programa com estimativa de consumo de*



*energia pode ser realizada 100 vezes mais rápida, na média, em relação ao fluxo tradicional.*

- **Análise de Desempenho de Topologias para Redes em Chip**, por Nelson Antonio Gonçalves Junior, Ronaldo A. L. Goncalves , João Angelo Martini, Universidade Estadual de Maringá.

*Os avanços nos processos de fabricação de chips têm permitido um constante aumento na quantidade de transistores integrados em uma mesma pastilha de silício, possibilitando a associação de todos os componentes de um computador em um único chip. São os chamados Sistemas em Chip (SoCs – Systems on Chip), cuja complexidade vem aumentando frequentemente com a integração de diversos componentes e exigindo formas eficientes de realizar a comunicação entre esses componentes. Uma abordagem muito discutida atualmente para garantir essa comunicação é a utilização de redes em chip (NoCs – Networks on Chip), que mantêm chaves roteadoras para direcionar os pacotes de dados para seus respectivos destinos e são interligadas de acordo com determinada topologia. Nesse contexto, o presente trabalho busca investigar tais redes, analisando e comparando o desempenho de NoCs com topologias diferentes, para mostrar o impacto que a topologia pode ter no desempenho final de uma rede intra-chip.*

**WSCAD – Sessão Técnica V – Aplicações de Alto Desempenho II**  
**Dia: Sábado, 31 – 09:00 – Prédio 40, Sala 103**

- **Aplicando Model-Driven Development à Plataforma GPGPU**, por Ademir Carvalho, Thiago Farias, UFPE, João Marcelo Teixeira, Centro de Informática - UFPE, Veronica Teichrieb, UFPE, Judith Kelner, UFPE.  
*GPUs (Graphics Processing Units) são dispositivos gráficos que vêm ganhando destaque nos últimos anos pela sua eficiência em processamento paralelo. Neste contexto, o termo GPGPU (General-Purpose computation on GPU) é um novo conceito que visa explorar as vantagens das GPUs em áreas não necessariamente relacionadas a processamento gráfico. Este trabalho aplica princípios do MDD (Model-Driven Development) ao desenvolvimento de aplicações para GPU, visando produzir um ambiente mais adequado para a construção desse tipo de software. O resultado do trabalho foi o desenvolvimento de uma ferramenta que enxerga uma aplicação como um modelo e gera automaticamente parte significativa do código desta aplicação. O código gerado é expresso na linguagem definida por CUDA (Compute Unified Device Architecture), uma plataforma de programação para GPGPU.*
- **Implementação Paralela em C+CUDA de um Categorizador Multi-Rótulo de Texto Baseado no Algoritmo k-NN**, por Lucas Veronese, Alberto De Souza, UFES, Claudine Badue, UFES, Elias Oliveira, UFES.  
*Em problemas de categorização automática de texto com um grande número de rótulos, as bases de dados de treinamento são grandes, o que pode tornar proibitivo o tempo de categorização para sistemas on-line. Neste trabalho avaliamos a implementação paralela em C+CUDA (Compute Unified Device Architecture) de um categorizador multi-rótulo de texto baseado no algoritmo k-NN (k-Nearest Neighbors). Nós implementamos este algoritmo de duas formas: seqüencial em C e paralela em C+CUDA. Nossos resultados experimentais mostram que, com o uso de GPUs (Graphics Processing Units) e C+CUDA, são possíveis ganhos de desempenho da ordem de 65 vezes o desempenho seqüencial alcançado com CPUs.*
- **Simulação de Neurônios Biologicamente Realistas em GPUs**, por Raphael Camargo, Universidade Federal do ABC.  
*GPUs modernas são compostas por centenas de processadores capazes de executar milhares de threads simultaneamente e estão sendo utilizadas com sucesso em diversas aplicações na área de computação de alto-desempenho. Neste trabalho apresentamos um simulador de modelos neuronais realistas capaz de simular centenas de milhares de neurônios por GPU. Resultados experimentais mostram que as simulações são executadas 5 vezes mais rápidas em GPUs, quando comparadas a CPUs modernas.*
- **Técnica de Proteção de Bytecodes para Processador Java em Tecnologia CMOS**, por Ricardo Jardel Silveira, UFC, David Viana, LESC/UFC, Helano Castro, Alexandre Coelho, UFC, Jarbas Silveira, Instituto Atlântico.

*O soft core JOP (Java Optimized Processor) para FPGAs (Field Programmable Gate Array) é uma implementação otimizada da máquina virtual Java em hardware, para aplicações de tempo real. No entanto, este processador não contempla em sua arquitetura técnicas de tolerância a falhas. O trabalho descrito neste artigo é parte de um esforço maior para tornar o processador JOP um processador tolerante a falhas. Neste artigo, apresentamos os resultados da aplicação de uma técnica de tolerância a falhas, proteção de memória através de ECC (Error Correction Code), no soft core JOP, que detecta e corrige erros na área destinada ao código da memória SRAM (Static Random Access Memory). A ocorrência da falha é percebida no nível sistêmico através de uma exceção, característica esta disponível na linguagem Java. Este artigo apresenta resultados inovadores na medida em que não existem registrados na literatura outro processador Java de tempo real e tolerante a falhas.*

**WSCAD – Sessão Técnica VI – Sistemas de Arquivos Distribuídos e Computação Móvel**  
**Dia: Sábado, 31 – 11:10 – Prédio 40, Sala 103**

- **Análise de desempenho do sistema de arquivos Lustre sob padrões de acesso a dados exclusivos**, por Francieli Boito, Rodrigo Kassick, UFRGS, Philippe Navaux, UFRGS.

*Aplicações que executam em ambientes de cluster frequentemente geram grandes quantidades de dados, que podem precisar ser acessados de forma eficiente por todos os nós envolvidos na computação. Sistemas de arquivos distribuídos (SADs) constituem uma solução natural nesse caso. Dependendo das suas opções de projeto, esses sistemas podem apresentar variados comportamentos sob diferentes padrões de acesso. Assim, estudar como se comporta o desempenho de um SAD sob padrões de acesso verificados na prática é uma tarefa importante, pois fornece ferramentas para que as aplicações possam adequar as suas operações de E/S para tirar melhor proveito do sistema utilizado. Um desses padrões comuns em aplicações científicas é o acesso pelos nós a dados exclusivos. Duas formas de fazê-lo são empregar um arquivo por processo, ou segmentos de um arquivo compartilhado. Esse artigo apresenta um estudo sobre o comportamento do sistema de arquivos Lustre nessas situações, a fim de determinar a que deve ser preferida em cada caso. Para realizar o estudo, foram realizados testes que simulam os padrões de acesso. Os resultados permitiram concluir que, para operações de escrita com números grandes de clientes, a abordagem de arquivos múltiplos é a melhor. No entanto, para outras situações, ela se mostra equivalente e até pior que a de arquivo compartilhado.*

- **Um modelo de armazenamento de metadados tolerante a falhas para o DDGfs**, por Alexandro Soares, UFCG.

*O Desktop Data Grid file system (DDGfs) é um sistema de arquivos distribuído desenvolvido para atender a requisitos de escalabilidade e manutenibilidade não oferecidos por sistemas de arquivos distribuídos amplamente utilizados na prática, como NFS e Coda. No DDGfs, dados e metadados são armazenados em componentes separados. Os dados são armazenados em servidores de dados, enquanto os metadados são mantidos em um único servidor de metadados. Essa arquitetura facilita o projeto do sistema, mas torna o servidor de metadados um ponto único de falha. Apesar de ser considerado um componente confiável, falhas são inevitáveis. Se esse componente falhar, os metadados podem ser perdidos e todos os dados armazenados se tornam inacessíveis. Este artigo apresenta um modelo de armazenamento de metadados que permite que o servidor de metadados seja restaurado após uma falha catastrófica que corrompa seu estado.*

- **Um Modelo para Gerenciamento de Trilhas em Ambientes de Computação Móvel**, por Jader Silva, UNISINOS, João da Rosa, UNISINOS, Jorge Barbosa, Laerte Franco, UNISINOS, Débora Barbosa, Centro Universitário La Salle, Luiz Palazzo, Universidade Católica de Pelotas.

*Nos ambientes de computação móvel, o acompanhamento da mobilidade permite a adaptação das aplicações aos contextos percorridos pelo usuário (ciência de contexto). Nos últimos anos, o uso conjunto de contextos e perfis de usuários vem sendo considerado uma oportunidade para a distribuição de conteúdo. Além disso, o aprimoramento e a ampla adoção dos sistemas de localização vêm estimulando ainda mais o acompanhamento da mobilidade, viabilizando o uso de Trilhas. Uma trilha consiste no histórico dos contextos visitados por um usuário durante um período. Este artigo propõe um modelo para o gerenciamento de trilhas. Considera-se que a Ciência de Trilhas é uma evolução da proposta de uso conjunto de contextos e perfis. O texto apresenta um protótipo e sua aplicação em um ambiente educacional ubíquo para a distribuição de conteúdo guiada pelas trilhas de aprendizes.*

## LAGRID – Technical Sessions

### LAGrid – Position Papers I

Day: Wednesday, 28<sup>th</sup> – 10:45 – Building 13, Room 402

- **GoStorm: A New Platform Driven to Scale-out Environments**, by André Eberle, Gustavo Frainer, Ricardo Rios, Lourenço Pereira Jr, Mauro Storch, Micael Geiss, GoDigital Ltda, Brazil, Rodrigo Mello, University of São Paulo, Brazil.
- **A Proposal of Prediction and Diagnosis in Grid Computing Self-Healing Problem**, by Mariza Ferro, National Laboratory of Scientific Computing, Brazil, Antonio Mury, Bruno Schulze, LNCC, Brazil.
- **Middleware integration with SAEMC\_GRID portal**, by Eugenio Almeida, Instituto Nacional de Pesquisas Espaciais (INPE), Brazil, Cláudio Baeza Retamal, Rodrigo Delgado Urzúa, Centro de Modelamiento Matemático (CMM) Universidad de Chile, Chile Luiz Henrique Ribeiro Coura da Silva, INPE/CPTEC, Brazil.

### LAGrid – Position Papers II

Day: Wednesday, 28<sup>th</sup> – 13:30 – Building T, Room 509

- **A Grid-based Infrastructure for Interoperability of Distributed and Heterogeneous PACS**, by Ramon Costa, Universidade Federal do Espírito Santo - UFES, Brazil, Alvaro Barbosa, Saulo Bortolon, UFES, Brazil, Bruno Schulze, LNCC, Brazil.
- **UnaGrid: A Virtual Opportunistic Grid Infrastructure for Custom Application Environments**, by Mario Villamizar, Eduardo Rosales, Harold Castro, Artur Jiménez, Universidad de los Andes, Colombia.

## WEAC

### WEAC – Abertura

**Dia: Quarta-feira, 28 – 08:30 – Auditório João Calvino**

### WEAC – Sessão de Artigos Completos

**Dia: Quarta-feira, 28 – 08:50 – Auditório João Calvino**

- **Introdução à Arquitetura de Computadores: Uma Abordagem para Ponto Flutuante e Cache**, por Ricardo dos Santos Ferreira (Universidade Federal de Viçosa)
- **R2DSim: Simulador Didático do RISC Reconfigurável**, por Angélica Aparecida Moreira, Carlos Augusto Paiva da Silva Martins (Pontifícia Universidade Católica de Minas Gerais)
- **SCC: Um compilador C como ferramenta de ensino de Compiladores**, por Juliano Henrique Foleiss, Guilherme Puglia Assunção, Eduardo Henrique Molina Cruz, Ronaldo Augusto de Lara Gonçalves, Valéria Delisandra Feltrim (Universidade Estadual de Maringá)
- **Ultimate Pipeline Simulator: Simulador do Pipeline do MIPS**, por César da Cunha Barcellos, Henrique Magalhães Simões (Pontifícia Universidade Católica de Minas Gerais)

### WEAC – Sessão de Artigos Curtos 1

**Dia: Quarta-feira, 28 – 10:45 – Auditório João Calvino**

- **Abordagem da educação integrando arquitetura de computadores com programação utilizando o LEGO MINDSTORMS**, por Bruno Augusto Teixeira, Diego Tavares da Silva, Gabriel de Godoy Correa e Castro (Pontifícia Universidade Católica de Minas Gerais)
- **Arquiteturas não convencionais incorporadas aos cursos de computação**, por Luciana dos S. Belo, Mauro S. de Castro, Tiago R. Lopes dos Santos (Pontifícia Universidade Católica de Minas Gerais)
- **Computação Paralela e Educação: um Elo Promissor**, por Breno S. Araújo, Igor Alves Perdigão, Jorge Moisés Assis (Pontifícia Universidade Católica de Minas Gerais)
- **Impactos, Oportunidades e Desafios no contexto da Educação em Arquitetura de Computadores causados pelos Processadores com Múltiplos Núcleos**, por

Lucas Goulart Silva, Márcio Oliveira Soares de Souza (Pontifícia Universidade Católica de Minas Gerais)

- **O Desafio de Ensinar Eletrônica e Arquitetura de Computadores nos Cursos de Engenharia Elétrica**, por Stefano Valente Menicucci (Pontifícia Universidade Católica de Minas Gerais)
- **Proposta de abordagem para a educação em arquitetura de computadores em diferentes cursos de graduação**, por Rogério Fernandes Carvalho, Gilson Antônio Fernandes (Pontifícia Universidade Católica de Minas Gerais)

**WEAC – Sessão de Artigos Curtos 2**

**Dia: Quarta-feira, 28 – 13:30 – Prédio T – Sala 506**

- **Proposta para o Processo de Ensino e Aprendizagem de Arquitetura de Computadores nos Cursos de Engenharia Eletrônica**, por Dayse Nascimento Anselmo, Roberto de Oliveira Campos Junior (Pontifícia Universidade Católica de Minas Gerais)
- **PS - CAS MIPS: Um Simulador De Pipeline Do Processador MIPS 32 Bits Para Estudo de Arquitetura de Computadores**, por Davidson W. N. Maia, Marcelo M. Vieira, Ramon F. Pessoa (Pontifícia Universidade Católica de Minas Gerais)
- **Simulação Baseada em Atores para no Ensino de Arquitetura de Computadores**, por Alisson V. Brito (Universidade Federal da Paraíba)
- **SIPPIS MIPS: Um Novo Simulador de Pipeline do MIPS 32 bits para Auxílio à Educação em Arquitetura de Computadores**, por Michael Martins Cosendey, Rafael Henrique Brasil, Carlos A. Paiva da Silva Martins (Pontifícia Universidade Católica de Minas Gerais)
- **Arquitetura de Computadores - Um Conjunto de Exercícios de Motivação**, por Rodolfo Azevedo (UNICAMP)

**WEAC – Painel de Discussão “Laboratórios de Arquitetura de Computadores”**

**Dia: Quarta-feira, 28 – 14:45 – Prédio T – Sala 506**

**WEAC – Sessão Especial “Grandes Desafios da Educação em Arquiteturas de Computadores”**

**Dia: Quarta-feira, 28 – 15:45 – Prédio T – Sala 506**

- **Reformulação do Ensino e Aprendizado de Arquitetura de Computadores no Contexto do novo Paradigma do Processamento Paralelo**, por Luiz Guilherme



Hilel Drumond Silveira, Carlos Augusto Paiva da Silva Martins (Centro Universitário UNA) (Pontifícia Universidade Católica de Minas Gerais)

- **Ensino e aprendizado do Pensamento Paralelo às Próximas Gerações de Cientistas da Computação**, por Angélica Aparecida Moreira, Daniel Armando Ribeiro Campos, Carlos Augusto Paiva da Silva Martins (Pontifícia Universidade Católica de Minas Gerais)
- **Formação de Docentes em Arquitetura de Computadores**, por Rodolfo Azevedo (UNICAMP)
- **Novas Perspectivas no Contexto da Arquitetura de Computadores - Softwares Educacionais para o Paradigma da Computação Paralela**, por Diego Marinho de Oliveira, Fernando Augusto Fernandes Braz, Tiago Figueiredo de Carvalho (Pontifícia Universidade Católica de Minas Gerais)
- **Os Impactos de um Modelo de Abstração em Camadas para a Educação em Arquitetura de Computadores**, por Wesley Oliveira Maia (Pontifícia Universidade Católica de Minas Gerais)
- **Pesquisa Científica no 1º Período como Motivação para o Estudo de Arquitetura de Computadores**, por Henrique Cota de Freitas (Pontifícia Universidade Católica de Minas Gerais)
- **Redefinição do Termo Arquitetura de Computador**, por Carlos Augusto Paiva da Silva Martins (Pontifícia Universidade Católica de Minas Gerais)
- **Revalorização da Arquitetura de Computadores no contexto da Engenharia Eletrônica**, por Paulo Júnio Vilaça Pousa (Pontifícia Universidade Católica de Minas Gerais)
- **Revitalização da educação em arquitetura de computadores, através do foco na construção de aplicações paralelas com qualidade**, por Roberto de Oliveira Campos Junior, Dayse Nascimento Anselmo (Pontifícia Universidade Católica de Minas Gerais)
- **Valorização acadêmica no processo de formação de um verdadeiro cientista da Computação**, por Izabela Éveli Martins Viana, Thiago Melo Machado Coelho, Harison Herman Silva, Luiz Guilherme Hilel Drumond Silveira, Carlos Augusto Paiva da Silva Martins (Pontifícia Universidade Católica de Minas Gerais)

**WEAC – Sessão de discussão e definição dos grandes desafios**

**Dia: Quarta-feira, 28 – 16:40 – Prédio T – Sala 506**

**WEAC – Sessão de Fechamento - Assembléia Geral do Workshop**

**Dia: Quarta-feira, 28 – 17:40 – Prédio T – Sala 506**

## WCTD

### **WCTD – Trabalhos de Mestrado**

**Dia: Quarta-feira, 28 – 15:45 – Prédio T, Sala 507**

- **Controle de Granularidade com threads em Programas MPI Dinâmicos**, por João Vicente Lima - UFRGS
- **Estudo de Casos com Aplicações Científicas de Alto Desempenho em Agregados de Computadores Multi-core**, por Luiz Carlos Pinto - UFSC
- **Investigação de Técnicas de Projeto de Cache de Dados para Sistemas Embarcados**, por Giancarlo Heck - UFPR

### **WCTD – Trabalhos de Doutorado**

**Dia: Quarta-feira, 28 – 17:10 – Prédio T, Sala 507**

- **Um Modelo Alternativo para Execução Eficiente de Aplicações Paralelas MPI nas Grades Computacionais**, por Alexandre da Costa Sena - UFF
- **Ambiente para Programação Orientada a Objetos Distribuídos e Paralelos em Grades Computacionais**, por Calebe Bianchini – USP

## SINAPAD

**SINAPAD – Abertura****Dia: Quinta-feira, 29 – 14:00 – Prédio T – Sala 508****SINAPAD – Apresentação dos CENAPADs****Dia: Quinta-feira, 29 – 14:30 – Prédio T – Sala 508****SINAPAD – Palestras****Dia: Quinta-feira, 29 – 16:05 – Prédio T – Sala 508**

- **O sistema HeMoLab: simulação numérica de larga escala na modelagem do sistema cardiovascular humano**, por Paulo Ziemer / Daniel Golbert (LNCC/MCT)
- **Modelagem numérica de eventos severos de tempo: Microbursts**, por Vagner Anabor (DF/UFSM)
- **Prove niência em workflows científicos e o SINAPAD**, por Marta Mattoso (COPPE/UFRJ)

**SINAPAD – Palestras****Dia: Sexta-feira, 30 – 14:00 – Prédio T – Sala 508**

- **Gerência de execução de aplicações computacionalmente complexas**, por Renato Cerqueira (DI/PUC-Rio)
- **Gerência de clusters através de Serviços Web e interfaces Web dinâmicas**, por Albino Avelada (COPPE/UFRJ)
- **Democratizando o acesso ao processamento de alto desempenho**, por Francisco Brasileiro (DSC/UFCG)

**SINAPAD – Palestras****Dia: Sexta-feira, 30 – 16:10 – Prédio T – Sala 508**

- **Gerência de dados para e-Science**, por Fábio Porto (LNCC/MCT)
- **Transporte de grandes massas de dados em redes com alto produto banda-latência**, por José Ferreira de Rezende (COPPE/UFRJ)

**SINAPAD – Encerramento****Dia: Sexta-feira, 30 – 17:30 – Prédio T – Sala 508**

## WIC

### WIC – Sessão I

**Dia: Quarta-feira, 28 – 13:15 – Prédio T – Sala 508**

- **Geração automática de "mutantes" baseada na especificação para construção de casos de testes em sistemas embarcados**, por César Rocha (UFAM), Raimundo Barreto (UFAM)
- **Geração automática de componentes de comunicação a partir de um modelo UML do sistema para plataformas baseadas em barramento**, por Josiane Bezerra (UFPE), Ciro Ceissler (UFPE), Denys Farias (UFPE)
- **Detecção de Marcadores para Realidade Aumentada em FPGA**, por Bernardo Reis (UFPE), João Marcelo Teixeira (UFPE), Veronica Teichrieb (UFPE), Judith Kelner (UFPE)
- **Simulação Quântica em VHDL: Um Estudo de Caso Baseado no Algoritmo Quântico de Grover**, por Eduarda Monteiro (UCPel), Renata Reiser (UCPel), Diego Porto (UCPel), Eduardo Costa (UCPel), Mauricio Pilla (UFPEL)
- **Aplicações do Protótipo VPE-qGM: Simulação via Software do Algoritmo de Grover**, por Adriano Maron (UCPel), Renata Reiser (UCPel), Adenauer Yamin (UCPEL and UFPEL), Mauricio Pilla (UFPEL)
- **Suporte para a descrição da arquitetura ARM utilizando a ADL ArchC**, por Rafael Auler (UNICAMP), Paulo Centoducatte (UNICAMP)
- **Metaheurísticas paralelas Algoritmos Genéticos e Resfriamento Simulado para solução de topologia de redes ópticas**, por Michael André Gonçalves (Instituto Federal do Espírito Santo), Sérgio Simões (Instituto Federal do Espírito Santo)
- **Análise de Consumo de Energia para Hierarquias de Memória com 2 Níveis**, por Gabriel Carvalho (UFPE), Antônio Araújo (UFPE), Filipe Cordeiro (UFPE), Abel Silva-Filho (UFPE)

### WIC – Sessão II

**Dia: Quarta-feira, 28 – 15:40 – Prédio T – Sala 508**

- **Implementação Paralela em C+CUDA de uma Rede Neural Probabilística**, por Lucas Veronese (UFES), Alberto De Souza (UFES), Claudine Badue (UFES), Elias Oliveira (UFES)

- **Avaliação de Desempenho de uma Aplicação do tipo Bag-of-Tasks utilizando o Middleware OurGrid**, por Tiago Machado (UFJF), Marcelo Lobosco (UFJF)
- **Desempenho SPEC2000 Preliminar da Arquitetura Dynamically Trace Scheduling Dataflow**, por Jorcy Neto (UFES), Alberto De Souza (UFES)
- **Estratégias Sequenciais e Paralelas de GRASP com Reconexão por Caminhos para o Problema de Síntese de Redes a k-Caminhos**, por Bruno Paes (UFF), Isabel Rosseti (PUC-Rio)
- **Metodologia de Pesquisa em Arquitetura de Computadores Baseado em Camadas para Robótica**, por Vicente Cruz (UFRGS), Henrique Freitas (PUC-MG), Philippe Navaux (UFRGS)
- **Avaliação do Sistema de Arquivos Paralelo do Cluster Netuno**, por Juliana Correa (UFRJ), Gabriel P. Silva (UFRJ)
- **Uma FFT Híbrida e Paralela para o Supercomputador Cray XD1**, por Vitor Conrado Faria Gomes (UFSM), Andrea Charao (UFSM), Haroldo Campos Velho (INPE)
- **Um Ambiente para Gerenciamento de Experimentos de Simulação com Redes de Petri**, por Leticia Gindri (UFS), Daniela Cruz (UFS), Ricardo Salgueiro (UFS), Edilayne Salgueiro (UFPE)
- **Algoritmo Genético Paralelo com uma Ilha Semi-Aleatória**, por Izabela Éveli Viana (PUC-MG), Thiago Machado Coelho (PUC-MG), Harison Silva (PUC-MG), Luiz Drumond (PUC-MG), Carlos Augusto Martins (PUC-MG)

## Minicursos

**Minicurso 1: Networks-on-Chip: Conceitos, arquiteturas e tendências****Dia: Sábado, 31 – 14:50 – Prédio 40 – Sala 102**

**Palestrantes:** Henrique Freitas, Pontifícia Universidade Católica de Minas Gerais, Brazil  
Philippe Navaux, UFRGS, Brazil.

**Resumo:**

Com a proposta deste mini-curso pretendemos atingir quatro objetivos principais: - Apresentar e difundir os conhecimentos relacionados às Networks-on-Chip (NoCs) para os futuros processadores many-core. - Produzir e disponibilizar literatura em português sobre Networks-on-Chip. - Apresentar arquiteturas, análises, aplicações e comparações entre Networks-on-Chip. - Mostrar e analisar a viabilidade das NoCs usando casos e problemas encontrados em pesquisas do estado da arte. Tratamento dado ao tema: Enfoques teóricos, mostrando a história e evolução até o conceito das NoCs, arquiteturas principais e propostas do estado da arte, arquiteturas de processadores multi/many-core que demandam NoCs, comparação entre as alternativas de NoCs pesquisadas, e métodos para avaliações de NoCs. Em breve mais informações.

**Minicurso 2: Computação Heterogênea de Alto Desempenho com FPGAs****Dia: Sábado, 31 – 14:50 – Prédio 40 – Sala 103**

**Palestrantes:** Andrea Charao, UFSM, Brazil Haroldo Campos Velho, Instituto Nacional de Pesquisas Espaciais, Brazil Vitor Conrado Faria Gomes, UFSM, Brazil.

**Resumo:**

A computação heterogênea tem sido apontada como uma das mais importantes tendências para processamento de alto desempenho na atualidade. Entende-se por computação heterogênea todo processamento que utiliza outros dispositivos além da CPU, como por exemplo Field-Programmable Gate Arrays (FPGAs) e Graphics Processing Units (GPUs). Embora não se trate propriamente de um novo paradigma, a computação heterogênea tem ganhado força com o surgimento de novos sistemas cuja arquitetura baseia-se nesta abordagem. Com relação ao desenvolvimento de código, o cenário da computação heterogênea também vem passando por transformações, com a ampliação das opções antes limitadas a APIs proprietárias e com longa curva de aprendizado. O principal objetivo deste minicurso é apresentar um panorama dos avanços recentes em computação heterogênea, com foco no uso de FPGAs como co-processadores paralelos de alto desempenho. Estes dispositivos, tradicionalmente usados no desenvolvimento de hardware, vem sendo incorporados em novos sistemas heterogêneos, como por exemplo o Cray XT5h. O desenvolvimento de aplicações nesse tipo de plataforma tira proveito de conceitos clássicos de programação paralela, mas também de novas ferramentas que apóiam o ciclo de desenvolvimento. Para alcançar o objetivo proposto, será feita primeiramente uma contextualização da computação heterogênea, para depois apresentar alguns casos de sistemas que utilizam esta abordagem. Na sequência será discutido o desenvolvimento de

software heterogêneo para CPU-FPGA e, por fim, será feita uma demonstração de uso, à distância, de um sistema deste tipo, disponível no Instituto Nacional de Pesquisas Espaciais.