

ERAD-SP 2018

Anais da 9ª Escola Regional de Alto Desempenho
de São Paulo
13 a 15 de Abril de 2018
São José dos Campos - SP

Realização



Patrocínio governamental



Apoio



Patrocinadores privados



ISSN 2178-1397

Mensagem dos Coordenadores

A Escola Regional de Alto Desempenho de São Paulo (ERAD-SP) é um evento realizado pela Sociedade Brasileira de Computação (SBC) desde 2010, sempre sob a autorização da Comissão Especial de Arquitetura de Computadores e Processamento de Alto Desempenho (CE-ACPAD) da SBC.

O evento justifica-se pela forte carência de conhecimento em Processamento de Alto Desempenho, Arquitetura de Computadores e Sistemas Distribuídos entre os desenvolvedores de aplicações no estado de São Paulo e no Brasil. A ERAD-SP vem disseminando tal conhecimento a esse público alvo de maneira ininterrupta há 9 anos (2010-2018), por meio de minicursos, palestras convidadas e apresentação de trabalhos de pesquisa desenvolvidos pela academia e pela indústria. Nesse cenário, a ERAD-SP permite a interação entre alunos, professores e profissionais do mercado de trabalho.

A ERAD-SP 2018 começou a ser planejada em Junho de 2017, pouco tempo depois do término da ERAD-SP 2017, tendo como meta manter seu caráter de *Escola Regional* e reunir a maior quantidade possível de participantes no evento. Cabe ainda lembrar que, a partir da ERAD-SP de 2017, o evento vem sendo realizado em meados do mês de abril, de forma a mantê-lo distante dos outros eventos relacionados à Área de Computação de Alto Desempenho com forte organização e participação de pesquisadores do Brasil, o WSCAD e o SBAC-PAD, eventos de caráter nacional e internacional, respectivamente, e que ocorrem normalmente no mês de outubro. Além disso, pela segunda vez consecutiva, a ERAD-SP ocorreu parcialmente em um final de semana (Sexta-feira, Sábado e Domingo), de forma a permitir uma maior participação de alunos, não afetando suas atividades discentes nas universidades, e permitindo também que os profissionais tenham maior liberdade de participação sem comprometer suas funções regulares nas instituições em dias úteis.

Por se tratar de uma Escola de Computação, a ERAD-SP 2018 focou majoritariamente nos alunos de graduação e também em pós-graduandos e profissionais interessados em construir sistemas computacionais de alto desempenho, envolvendo tanto o desenvolvimento de aplicações paralelas quanto a plataforma de execução.

A 9ª ERAD-SP, ou ERAD-SP 2018 (<http://eradsp2018.lsc.ic.unicamp.br/>), foi realizada de 13/04/2018 a 15/04/2018 no Faro Hotel em São José dos Campos - SP, local aprazível com infra-estrutura adequada e grande acessibilidade. A organização esteve sob responsabilidade do Instituto de Ciência e Tecnologia da Universidade Federal de São Paulo (ICT/UNIFESP) e do Instituto Nacional de Pesquisas Espaciais (INPE). Os números da programação do evento incluíram mais de 18 atividades diferentes, sendo que as principais foram: quatro minicursos, quatro tutoriais, duas palestras convidadas, três palestras patrocinadas, um desafio de programação paralela, um painel, duas sessões técnicas para apresentação oral, uma sessão técnica de apresentação de pôsteres, além da reunião ordinária da CRAD-SP.

Um total de 23 artigos foram submetidos e avaliados para apresentação nas sessões técnicas, sendo 19 artigos de autoria de alunos de pós-graduação e 4 artigos de alunos de graduação. Todos os 23 trabalhos submetidos foram avaliados por um mínimo de três revisores, sendo que 12 trabalhos foram avaliados por quatro revisores e 3 trabalhos foram avaliados por cinco revisores. Foram aceitos 16 artigos, sendo 9 para apresentação oral e 7 na forma de poster. Por decisão da Coordenação de Programa, todos os 16 artigos aceitos estão sendo aqui publicados. De acordo com a avaliação realizada pelos revisores, dois artigos receberam premiação: melhor artigo de pós-graduação (*Desempenho da comunicação MPI Shared Memory no Modelo Meteorológico BRAMS*, autoria de Carlos R. Souza, Jairo Panetta e Stephan Stephany) e melhor artigo de graduação (*Análise de Custo da Nuvem Computacional para a*

Execução de Algoritmos no Processamento Sísmico, autoria de Nikolas Okita, Tiago A. Coimbra e Edson Borin).

Entendemos que atingimos com sucesso os objetivos traçados para a ERAD-SP 2018. Esperamos que as contribuições deste evento apareçam ao longo dos anos, através de um maior número de alunos interessados na área. Por fim, faz-se necessário agradecer àqueles que contribuíram para o sucesso da ERAD-SP 2018. Agradecemos aos demais membros da Comissão Organizadora da ERAD-SP 2018 por terem aceitado o desafio desta empreitada, aos membros do Comitê de Programa pelas revisões feitas dentro do prazo, aos convidados que trouxeram um rico conhecimento às atividades do evento, aos alunos do grupo de apoio que ajudaram na secretaria do evento voluntariamente, à CE-ACPAD e à SBC pelo apoio fundamental ao evento, à CAPES, CNPq e FAPESP pelo suporte financeiro, ao INPE e ICT/UNIFESP pelo apoio com infraestrutura física, logística e de pessoal para a realização do evento, e, para finalizar, a todos os prestadores de serviço que nos atenderam com responsabilidade e profissionalismo.

Álvaro Luiz Fazenda / Celso Luiz Mendes - Coordenação Geral

Lucas Wanner / Jairo Panetta - Coordenação de Programa

Desafio de Programação Paralela

O Desafio de Programação Paralela tem como principal objetivo fomentar o conhecimento em programação paralela e distribuída em um torneio de alto nível. Na ERAD-SP 2018, essa competição foi um dos pontos altos do evento, sendo realizada na tarde do sábado (14/04/2018).

A competição foi dividida em dois períodos. No primeiro período, os times receberam a descrição de um dos problemas: resolução de um método numérico simplificado para a medição de calor. Esse período teve como principal objetivo acomodar todos os participantes do Desafio com os diversos ambientes da competição. O problema foi escolhido devido à sua característica de poder ser implementado com diferentes técnicas de paralelismo. Particularmente para esse problema, não existe dependência espacial, apenas temporal, ou seja, a cada iteração o processamento pode ser dividido facilmente, mas é necessário sincronizar a cada avanço no tempo.

No segundo período, os times receberam o caderno completo de questões, adicionando-se o último problema: soma especial de matrizes bidimensionais. Este enunciado foi criado na intenção de apresentar um problema simples a ser resolvido com qualquer técnica de paralelismo e sem dependência, mas com grande uso de memória. Essa característica permite soluções que utilizam mais nós de processamento para aproveitar ainda mais o uso da memória e não exigir tanto o uso da área de swap - o que geraria um gargalo de acesso nesse nível.

Além disso, os times receberam também uma solução sequencial (serial) de cada um destes problemas. Durante a competição, as resoluções não só deveriam ter as respostas iguais às das soluções originais, mas também apresentar desempenho (speedup) em suas versões paralelas (ou distribuídas), medidas de acordo com critérios clássicos de avaliação de desempenho.

Durante a competição, os times tiveram acesso a um cluster heterogêneo oferecido pela UNIFESP, sendo este cluster formado de processadores multicore, manycores e GPGPUs, com rede de alta velocidade (Infiniband), e diversos compiladores (GCC, Intel, NVidia).

Cada time foi composto por até 3 (três) estudantes e um treinador (opcional). Na ERAD-SP 2018, o Desafio recebeu 9 times, com um total de 21 participantes, dentre alunos de graduação e pós-graduação. Os competidores não podem fazer acesso a qualquer tipo de material online sobre os problemas do Desafio, mas podem consultar qualquer material impresso (livros, manuais, anotações, artigos). Também não é permitido consultar outras pessoas ou competidores, além dos membros da sua própria equipe.

Ao final, dois times se destacaram e foram declarados vencedores do Desafio de Programação Paralela da ERAD-SP 2018, sendo o time dos participantes Manoel Ferreira Neto e João Morais, ambos da UNIFESP, ganhador em primeiro lugar, e o time dos participantes Jeferson Brunetta e Nicholas Torres Okita, ambos da UNICAMP, ganhador em segundo lugar.

Os prêmios foram cedidos pelos patrocinadores da ERAD-SP 2018, os quais têm incentivado cada vez mais a realização de competições e aumentando a participação de alunos.

Calebe de Paula Bianchini / Daniel Merli Lamosa – Coordenação do Desafio

Comitês

Coordenação Geral

Prof. Dr. Álvaro Luiz Fazenda (ICT/UNIFESP)

Prof. Dr. Celso Luiz Mendes (LAC/INPE)

Coordenação de Programa

Prof. Dr. Lucas Wanner (IC/UNICAMP)

Prof. Dr. Jairo Panetta (ITA)

Coordenação de Minicursos e Tutoriais

Profa. Dra. Denise Stringhini (ICT/UNIFESP)

Coordenação Local

Prof. Dr. Tiago de Oliveira (ICT/UNIFESP)

Coordenação do Desafio de Programação Paralela

Prof. Dr. Calebe de Paula Bianchini (MACKENZIE)

Prof. Daniel Merli Lamosa (CPTEC/INPE)

Comitê de Programa

Aleardo Manacero Jr.	- UNESP/IBILCE - Universidade Estadual Paulista
Alexandre Rocha	- UNESP
Alexandro Baldassin	- UNESP-IGCE
Alfredo Goldman	- IME - USP
Alvaro Fazenda	- Universidade Federal de São Paulo
Arlindo da Conceição	- ICT-UNIFESP
Caetano Miranda	- Universidade de São Paulo
Calebe Bianchini	- Universidade Presbiteriana Mackenzie
Celso Hirata	- ITA
Celso Mendes	- INPE
Daniel Cordeiro	- Universidade de São Paulo
Daniel Pedronette	- UNESP
Denise Stringhini	- Unifesp
Edson Borin	- Universidade de Campinas
Eduardo Rodrigues	- IBM Research Brazil
Emilio Franceschini	- UNICAMP
Francisco Massetto	- Universidade Federal do ABC
Francisco Monaco	- ICMC - Universidade de São Paulo
Helio Guardia	- Universidade Federal de São Carlos
Hermes Senger	- Universidade Federal de São Carlos
Igor Freitas	- Intel
Jairo Panetta	- Instituto Tecnológico de Aeronáutica
Jó Ueyama	- Universidade de São Paulo
Júlio Estrella	- ICMC-USP
Liria Sato	- EPUSP-USP
Luciana Arantes	- Université de Paris VI
Luciano Silva	- Universidade Mackenzie
Luis Scott	- UFABC
Luiz Fernando Bittencourt	- UNICAMP
Marcio Castro	- Universidade Federal de Santa Catarina
Marco Netto	- IBM Research
Norian Marranghello	- UNESP
Paulo Lopes de Souza	- ICMC-USP
Philippe O. A. Navaux	- UFRGS
Raphael Camargo	- Universidade Federal do ABC
Renata Spolon Lobato	- UNESP - Universidade Estadual Paulista
Renato Ishii	- Universidade Federal de Mato Grosso do Sul
Ricardo Menotti	- Universidade Federal de São Carlos
Ricardo Santos	- Universidade Federal do Mato Grosso do Sul
Roberta Spolon	- UNESP
Rodolfo Azevedo	- UNICAMP
Sandro Rigo	- Universidade do Vale do Rio dos Sinos
Sarita Bruschi	- ICMC-USP
Siang Song	- Universidade de São Paulo
Vanderlei Bonato	- ICMC-USP

SBC

Sociedade Brasileira de Computação

Diretoria

Lisandro Zambenedetti Granville (UFRGS) - Presidente
Thais Vasconcelos Batista (UFRN) - Vice-Presidente
Renata de Matos Galante (UFRGS) - Administrativa
Carlos André Guimarães Ferraz (UFPE) - Finanças
Antônio Jorge Gomes Abelém (UFPA) - Eventos e Comissões Especiais
Renata Mendes de Araújo (UNIRIO) - Educação
José Viterbo Filho (UFF) - Publicações
Claudia Lage Rebello da Motta (UFRJ) - Planejamento e Programas Especiais
Marcelo Duduchi Feitosa (CEETEPS) - Secretarias Regionais
Eliana Silva de Almeida - Divulgação e Marketing
Ricardo de Oliveira Anido (UNICAMP) - Relações Profissionais
Esther Colombini (UNICAMP) - Competições Científicas
Raimundo José de Araújo Macêdo (UFBA) - Cooperação com Sociedades Científicas
Cláudia Cappelli (UNIRIO) - Articulação com Empresas
Leila Ribeiro (UFRGS) – Ensino de Computação na Educação Básica

Conselho

Altigran Soares da Silva (UFAM)
Fabio Kon (USP)
Paulo Roberto Freire Cunha (UFPE)
Ana Carolina Salgado (UFPE)
Rodolfo Azevedo (UNICAMP)
José Carlos Maldonado (USP)
Rosa Maria Vicari (UFRGS)
Itana Maria de Souza Gimenes (UEM)
Roberto da Silva Bigonha (UFMG)
Cristiano Maciel (UFMT)

Comissão Especial de Arquitetura de Computadores e Processamento de Alto Desempenho - CE-ACPAD

Alfredo Goldman (USP) - Coordenador

Secretaria Regional de São Paulo Leste

Marcia Ito (IBM) - Secretária

CRAD-SP
Comissão Regional de Alto Desempenho
do Estado de São Paulo

Região Capital I: **Daniel Cordeiro** (USP, São Paulo)

Região Capital II: **Rogério L. Iope** (NCC/UNESP)

Região Interior I: **Denise Stringhini** (UNIFESP, S.J. dos Campos)

Região Interior II: **Ricardo Menotti** (UFSCar)

Região Interior III: **Edson Borin** (UNICAMP) – **PRESIDENTE**

Coordenadores Gerais da ERAD-SP 2016 e 2017: **Aleardo Manacero** (UNESP, S.J. do Rio Preto) e **Paulo Sérgio Lopes de Souza** (ICMC/USP)

Representante da Indústria: **Pedro Mário Cruz e Silva** (NVIDIA)

Minicursos

Minicurso I: *OpenMP – Um Modelo de Programação Paralela com Threads*

Palestrante: Prof. Dr. Calebe P. Bianchini (U.Mackenzie)

Resumo: OpenMP é um modelo de programação paralela que oferece, basicamente, diretivas de compilação, bibliotecas e variáveis de ambiente. Mas, o que pode ser feito com tudo isso? Nesse minicurso vamos explorar alguns dos recursos existentes no OpenMP atual (versão 4.5) e resolver problemas clássicos da computação. Utilizaremos o modelo de paralelismo apresentado no OpenMP chamado de fork/join e, juntamente com outros recursos disponíveis, teremos um alto grau de abstração para compartilhamento, sincronização, divisão de tarefas, e portabilidade, desenvolvendo soluções independentes de arquiteturas e plataformas de execução.

Minicurso II: *Introdução à Programação Paralela em Clusters Heterogêneos*

Palestrante: Prof. Dr. Silvio Stanzani (UNESP)

Resumo: Atualmente, arquiteturas computacionais têm sido montadas de modo heterogêneo, compostas por recursos computacionais que possuem diversos processadores e coprocessadores ou aceleradores que podem ser usados em conjunto por uma mesma aplicação. Tais arquiteturas disponibilizam diversas unidades heterogêneas de processamento, escalares e vetoriais, e em geral dispõem de um sistema de memória heterogêneo e composto por múltiplos níveis. Mais recentemente, tem-se observado uma variação quanto ao uso de coprocessadores e/ou aceleradores como nós independentes, que podem ser agregados em um sistema computacional por meio de uma rede de alto desempenho. Dois exemplos de uso de coprocessadores como nós independentes são o DGX-SATURNV desenvolvido pela NVIDIA que utiliza nós do tipo NVIDIA DGX-1, e o projeto Aurora, que é um projeto de supercomputador que prevê o uso de nós do tipo Xeon Phi (KNL). Nesse sentido, as infraestruturas de clusters heterogêneas permitem explorar as diferentes vantagens que cada arquitetura pode oferecer. O objetivo deste minicurso é apresentar uma introdução a arquiteturas paralelas heterogêneas, ao modelo de programação por troca de mensagens e técnicas elementares de otimização para tais ambientes.

Minicurso III: *Fundamentos de Computação Acelerada com CUDA C/C++*

Palestrante: Dr. Pedro Mario Cruz e Silva (NVIDIA)

Resumo: Este curso dará uma introdução ao desenvolvimento de aplicações massivamente paralelas com CUDA em C/C++ para GPUs da NVIDIA. O público-alvo deste curso são desenvolvedores com alguma experiência com as linguagens C/C++ que estejam interessados em acelerar o desempenho de suas aplicações além dos limites da programação somente para CPU. Neste curso, você aprenderá como: estender seu código C/C++ com o modelo de programação CUDA; escrever e executar kernels que executam com paralelismo massivo em uma GPU NVIDIA; perfilar e otimizar seus programas acelerados. Após a conclusão, você poderá escrever programas massivamente paralelos para arquiteturas heterogêneas com poderosas GPUs NVIDIA, e otimizar seu desempenho utilizando o NVVP.

Minicurso IV: *Programação de Aceleradores Baseada em Diretivas de Compilação*

Palestrantes: Pedro Pais Lopes (Exaflop)

Resumo: Processadores com centenas a milhares de núcleos de processamento estão presentes no cotidiano da computação. Este número expressivo de núcleos estão em placas gráficas (denominadas GPUs) como também em aceleradores x86 (como os Xeon Phi) ou

mesmo em chips ARM, com arquiteturas complexas de acesso a memória e de operações de ponto flutuante. A programação para utilizar este grande poder de processamento pode envolver linguagens especializadas, o que demanda profunda modificação do programa e possivelmente redução da sua portabilidade. Padrões de programação baseados em diretivas, amplamente utilizados para expor paralelismo em arquitetura de memória central, surgiram ou foram expandidos para permitir expor o paralelismo destes processadores massivamente paralelos. Neste minicurso serão tratados os padrões existentes (OpenACC e OpenMP v4.5), suas principais diretivas, compiladores com suporte a estes padrões e desempenho obtido em aplicações simples em algumas arquiteturas (GPU, CPU multicore e Xeon Phi).

Tutoriais

Tutorial I: *TensorFlow – Framework Multi-Plataforma para Deep Learning*

Palestrante: João Paulo de Oliveira (NVIDIA)

Resumo: TensorFlow é um framework de código aberto para computação numérica que utiliza o conceito de fluxo em grafos. Os nós do grafo representam operações matemáticas, enquanto suas arestas representam matrizes de dados multidimensionais. Matrizes, ou tensores, são transformados e fluem pelo grafo de maneira estruturada. A flexibilidade do TensorFlow™ permite a implementação de soluções de alto nível de Machine Learning e Deep Learning em CPUs ou GPUs. Atualmente é o framework mais utilizado para treinamento e inferência de redes neurais. Neste tutorial serão apresentados tópicos importantes para o desenvolvimento de aplicações com o framework: Arquitetura, características e sintaxe; Treinamento, inferência e deploy de aplicações; Escalabilidade de treinamentos em múltiplas GPUs; Recomendações de uso e bibliotecas auxiliares - TFLearn e TF-Slim; Aplicações em Visão Computacional.

Tutorial II: *Meltdown/Spectre*

Palestrante: Prof. Dr. Rodolfo Azevedo (UNICAMP)

Resumo: Recentemente foram divulgados dois bugs de processadores que têm o potencial de afetar praticamente todos os equipamentos computacionais ativos atualmente. Nesta palestra, serão discutidos os desenvolvimentos arquiteturais que permitiram que estes bugs surgissem e também como eles podem ser explorados e mitigados.

Tutorial III: *HardCloud / Intel Harp - Automatic Offloading of Cluster Accelerators*

Palestrantes: Ciro Ceissler, Ramon Nepomuceno, Marcio Pereira e Prof. Dr. Guido Araujo (UNICAMP)

Resumo: The sheer amount of computing resources required to run modern cloud workloads has put a lot of pressure on the design of power efficient cluster nodes. To address this problem, Intel (HARP) and Microsoft (Catapult) have proposed CPU-FPGA integrated architectures that can deliver efficient power-performance executions. Unfortunately, the integration of FPGA acceleration modules to software is a challenging endeavor that does not have a seamless programming model. This tutorial presents HardCloud (www.hardcloud.org), an extension of the OpenMP 4.X standard that eases the task of offloading FPGA modules to cluster accelerators. Participants will be able do hands-on experiments that use HardCloud to program FPGA modules in order to accelerate code on the Intel Altera HARP 2 architecture. This work will also be presented at the 26th IEEE International Symposium on Field-Programmable Custom Computing Machines (FCCM 2018).

Tutorial IV: *Charm++/AMPI*

Palestrante: Dr. Eduardo Rocha Rodrigues (IBM Research)

Resumo: Neste tutorial, o ambiente Charm++ para programação paralela é apresentado. Esse ambiente permite programação paralela orientada a objetos. Os objetos executam em paralelo e se comunicam de forma assíncrona. Dessa forma, a programação paralela se tornam mais próxima da semântica de interação entre objetos do mundo "real". Uma série de benefícios desse ambiente é apresentada. Por exemplo, balanceamento de carga e sobreposição automática de comunicação e processamento. Por outro lado, essa forma de programação com objetos assíncronos exige uma mentalidade diferente da programação tradicional. Para que programadores de aplicações paralelas tradicionais possam se beneficiar do ambiente Charm++ e ainda manter seu conhecido estilo de programação, o AMPI é apresentado, que é uma implementação de Message Passing Interface (MPI) sobre Cham++.

Palestras

Palestra Convidada I: *Approximate Computing, from Programming Language to the Hardware*

Palestrante: Prof. Luis Ceze (Paul G. Allen School of Computer Science and Engineering, University of Washington, EUA)

Resumo: A significant proportion of computer system resources are devoted to applications that can inherently tolerate inaccuracies in their data, execution and communication. Hence, approximate computing is promising for performance and energy efficiency. However, taking advantage of approximate computing needs: language support to specify where and how to apply approximation; analysis and mechanisms that ensure good output quality; and hardware/system support that take advantage of approximation. In this talk I will describe our effort on co-designing language, hardware and system support to take advantage of approximate computing across the system stack (compute, storage and communication) in a safe and efficient way. I will end with some thoughts on how effective approximate computing techniques can not only improve computer systems in the short and medium term but can also enable the use of new substrates and applications.

Palestra Convidada II: *Modernas Arquiteturas de Computadores: Core e Cache*

Palestrante: Dr. Leonardo Fialho (Atos/Bull – ARM Software Lab)

Resumo: Arquiteturas de computadores modernos são bastante complexas e incluem múltiplos níveis de paralelismo. Existem dois níveis de paralelismo dentro do núcleo, pelo menos um nível dentro do nó e também existe paralelismo entre os nós. Fazer uso eficiente dessas máquinas é um desafio que precisa ser planejado a partir do algoritmo de solução para o problema, através da distribuição de dados entre diferentes processadores. Existem diferentes estratégias para gerar um código eficiente, entretanto, algumas dessas estratégias podem se sobrepor ou estar em conflito. Procurar uma solução ideal é um desafio constante para programadores e também para especialistas em diferentes áreas do conhecimento. O único consenso é que o resultado final depende do nível de conhecimento que se possui sobre a arquitetura em que se deseja programar e executar a aplicação paralela.

Palestra Empresarial I: *Inteligência Artificial na Arquitetura Intel*

Palestrante: Igor Freitas, Intel

Palestra Empresarial II: *Apresentação SDC/Seagate*

Palestrante: Guilherme Friol, SDC

Palestra Empresarial III: *Projeto HPC Coral - Detalhes de Arquitetura*

Palestrante: Fernando Toledo, IBM

Painel

Tema: *Produtividade versus Desempenho em Projetos de Software em Computação de Alto Desempenho*

Painelistas:

Prof. Dr. Edson Borin (UNICAMP) - Moderador

Prof. Dr. Rodolfo de Azevedo (UNICAMP)

Prof. Dr. Alfredo Goldman (IME/USP)

Dr. Leonardo Fialho (BULL)

Prof. Dr. Celso Luiz Mendes (INPE)

Trabalhos de Iniciação Científica

Análise de Custo da Nuvem Computacional para a Execução de Algoritmos no Processamento Sísmico	01
<i>Nicholas T. Okita, UNICAMP</i>	
<i>Tiago A. Coimbra, UNICAMP</i>	
<i>Edson Borin, UNICAMP</i>	
Desempenho dos algoritmos Blowfish e RC6 usando CUDA	05
<i>Luccas de Quadros, UNESP</i>	
<i>Roberta Spolon, UNESP</i>	
<i>Renata Spolon Lobato, UNESP</i>	
<i>Aleardo Manacero Jr., UNESP</i>	
Distribuidor de Tarefas para uma plataforma de Computação Voluntária utilizando WebAssembly	09
<i>Miguel Felipe Silva Vasconcelos, USP</i>	
<i>Daniel Cordeiro, USP</i>	

Trabalhos de Pós-Graduação

Estudo preliminar de métricas de produtividade e portabilidade para linguagens de programação paralela	13
<i>Daniela F. Daniel, ITA</i>	
<i>Jairo Panetta, ITA</i>	
Uma análise do uso de containers para portabilidade de código para GPU na nuvem computacional	17
<i>Jeferson Rech Brunetta, UNICAMP</i>	
<i>Charles Boulhosa Rodamilans, UNICAMP</i>	
<i>Caian Benedicto, UNICAMP</i>	
<i>Edson Borin, UNICAMP</i>	
Desempenho de Implementações MPI de Redes Reversíveis usando Comunicação Unilateral e Tipos de Dados Derivados	21
<i>Matheus da Silva, ITA</i>	
<i>Carlos Henrique Costa Ribeiro, ITA</i>	
<i>Jairo Panetta, ITA</i>	
Avaliando o Impacto de Mudanças na Arquitetura de Memória entre Gerações de GPGPUs no Desempenho das Otimizações de Computações de Estênceis	25
<i>Rodrigo L. Machado, ITA</i>	
<i>Thiago C. Nasciutti, ITA</i>	
<i>Jairo Panetta, ITA</i>	
Desempenho da comunicação MPI Shared Memory no Modelo Meteorológico BRAMS	29
<i>Carlos R. de Souza, INPE</i>	
<i>Jairo Panetta, ITA</i>	
<i>Stephan Stephany, INPE</i>	
Approximate Reciprocal Square Root with Single- and Half-Precision Floats.....	33
<i>Matheus Susin, UNICAMP</i>	
<i>Lucas Wanner, UNICAMP</i>	
Otimização Computacional do Modelo BRASIL-SR	37
<i>Jefferson Gonçalves Souza, INPE</i>	
<i>Celso Luiz Mendes, INPE</i>	
<i>Rodrigo Costa Santos, INPE</i>	
Paralelização do algoritmo DIANA em OpenMP	41
<i>Hethini Ribeiro, UNESP</i>	
<i>Aleardo Manacero Jr., UNESP</i>	
<i>Roberta Spolon, UNESP</i>	
<i>Renata Spolon Lobato, UNESP</i>	
Tolerância a falhas em ambientes de computação em nuvem	45
<i>Vinicius S. Andrade, UNESP</i>	
<i>Aleardo Manacero Jr., UNESP</i>	
<i>Roberta Spolon, UNESP</i>	
<i>Renata Spolon Lobato, UNESP</i>	

Parallelization of a Large-Scale Watersheds Hydrological Model using CPU and GPU	49
<i>Henrique R. A. Freitas, INPE</i>	
<i>Celso L. Mendes, INPE</i>	
A Context-Aware Library for Mathematical Approximations	53
<i>Roberto Alejandro Hidalgo Castro, UNICAMP</i>	
<i>Lucas Wanner, UNICAMP</i>	
Implementações paralelas para o algoritmo Online Sequential Extreme Learning Machine aplicadas a Previsão de concentração de Material Particulado no ar	57
<i>Luís Fernando L. Grim, UNICAMP</i>	
<i>Jorge Andres Bueno Barajas, UNICAMP</i>	
<i>Andre Leon S. Gradvohl, UNICAMP</i>	
Uma Análise da Facilidade de Emulação de Binários RISC-V	61
<i>Leandro Lupori, UNICAMP</i>	
<i>Vanderson Martins do Rosario, UNICAMP</i>	
<i>Edson Borin, UNICAMP</i>	