RUTGERS CAC

The NSF Center for Autonomic Computing at Rutgers

http://nsfcac.rutgers.edu/

CoRE TeCH

Volume 1, Issue 2 Spring 2009

Sponsored by the National Science Foundation Office of Industrial Innovation and Partnerships Industry/University Cooperative Research Center (I/UCRC) Program under award 0758596.

Partner sites:
University of Florida
(www.nsfcac.org)

University of Arizona
(nsfcac.arizona.edu)

Inside this issue:

Address:
NSF Center for Autonomic Computing,
Rutgers University,
94 Brett Road ,
Piscataway ,
NJ 08854-8058

The Center for Autonomic Computing (CAC), an NSF Research Center funded by the I/UCRC program, combines resources from universities, private companies, and the federal government to conduct fundamental research on making all kinds of computer systems and applications more reliable, more secure, and more efficient.

Autonomic computing (AC) denotes a broad area of scientific and engineering research on methods, architectures and technologies for the design, implementation, integration and evaluation of special and general-purpose computing systems, components and applications that are capable of autonomously achieving desired behaviors. AC systems aim to be self-managed in order to enable independent operation, minimize cost and risk, accommodate complexity and uncertainty and enable systems of systems with large numbers of components. These might include, to varying degrees, self-organization, self-healing, self-optimization (e.g. for power or speed), self-protection and other similar behaviors. CAC research activities will advance several disciplines that impact the specification, design, engineering, and integration of autonomic computing and information processing systems. They include design and evaluation methods, algorithms, architectures, information processing, software, mathematical foundations and benchmarks for autonomic systems. Collectively, the participating universities have research and education programs whose strengths cover the technical areas of the center. Within this broad scope, the specific research activities will vary over time as a reflection of center member needs and the evolution of the field of autonomic computing.

What's new ?

CAC@Rutgers part of multiple prestigious grants from NSF and DOE:

"Actively Managing Data Movement with Models - Taming High Performance Data Communications in Exascale Machines," NSF HECURA, 08/08 - 07/11.
" Cross-layer Research on management of Virtualized Datacenters," NSF I/UCRC Supplement, 08/08 - 12/09.
"Development of Next Generation Collaborative Underwater Robotic Instrument," NSF MRI, 07/08 - 06/11.
"Center for Plasma Edge Simulation," DoE SciDAC FSP, 01/09 - 12/11.

CAC@Rutgers director appointed visiting research e-Science Institute, Edinburgh, UK to investigate the role of autonomics in e-science and engineering.

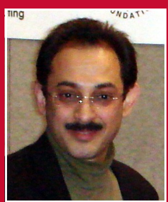
CAC@Rutgers director presented a seminar on the role of autonomics "Enabling Science on Emerging Computational Ecosystems," at the Office of Cyberinfrastructure of the National Science Foundation in Arlington, VA.

Founding members

intel, ISCA Technologies (Pest Management Tools & Solutions), Northrop Grumman, EWA Electronic Warfare Associates, Inc., Air Force Research Laboratory, BAE Systems, imaginestics, Merrill Lynch, Microsoft, NEC, Raytheon, IBM, xerox

From the Director's desk

by
Manish Parashar,
Director,
NSFCAC, Rutgers



It is my pleasure to introduce to you the National Science Foundation Center for Autonomic Computing (NSF CAC) - a center funded through the NSF Industry/University Cooperative Research Centers program, industry, government agencies and matching funds from member universities, which currently include the University of Florida (Lead), the University of Arizona and Rutgers, The State University of New Jersey. The mission of the center is to advance the knowledge of designing Information Technology (IT) systems and services to make them self-governed and self-managed. This will be achieved through developing innovative designs, programming paradigms and capabilities for computing systems.

In this newsletter, we highlight our ongoing autonomic research activities and how to apply them to wide range of applications/domains with profound impact on economy and industry. Research at the NSF-CAC will not only advance the science of autonomic computing, but also accelerate the transfer of technology to industry and contribute to the education of a workforce capable of designing and deploying autonomic computing systems to many sectors---from homeland security and defense, to business agility to the science of global change. Furthermore, the center will leave its mark on the education and training of a new generation of scientists that have the skills and know-how to create knowledge in new transformative ways. As we move forward, we would like to invite you to join the center so together we can develop innovative autonomic technologies and services that will revolutionize how to design and deploy next generation computation, information and communications services.

To become a member, contact:

Director:
Dr Manish Parashar
Office:
628 CoRE,
ECE Dept,
Rutgers University
Address:
94 Brett Road,
Piscataway ,
NJ 08854-8058
Voice:
(732) 445-5388
Email:
parashar
@rutgers.edu
Fax:
(732) 445-0593
Website:
http://
nsfcac.rutgers.edu/

Funding

Per NSF guidelines, industry and government contributions in the form of annual CAC memberships ($35K/year per regular membership), coupled with baseline funds from NSF and university matching funds, directly support the Center's expenses for personnel, equipment, travel, and supplies. Annual Memberships provide funds to support the Center's graduate students on a one-to-one basis, while NSF and university funds support various other costs of operation. Multiple annual memberships may be contributed by any organization wishing to support multiple students and/or projects. The initial operating budget for CAC is projected to be approximately $1.5M/year, including NSF and universities contributions, in an academic environment that is very cost effective. The single membership represents less than 3% of the projected annual budget of the Center yet reaps the full benefit of Center activities, a research program that could be significantly more expensive in an industry or government facility.

Benefits of Membership

CAC members are afforded access to leading-edge developments in autonomic computing and to knowledge accumulated by academic researchers and other industry partners. New members will join a growing list of our prestigious founding members. Benefits of membership include:

* Collaboration with faculty, graduate students, post-doctoral researchers and other center partners
* Choice of project topics to be funded by members' own contributions
* Formal periodic project reviews along with continuous informal interaction and timely access to reports, papers and intellectual property generated by the center
* Access to unique world-class equipment, facilities, and other CAC infrastructure
* Leveraging of investments, projects and activities by all CAC members
* Spin-off initiatives leading to new partnerships, customers or teaming for competitive proposals to funded programs

CAC Personnel at Rutgers

Faculty
Dr. Manish Parashar
Dr. Dario Pompili
Dr. Michael Bushnell
Dr. David Foran
Dr. Hui Xiong

Post-docs
Hyunjoo Kim

System Administrator
Jim Housell

UG Students
Pramod Kulkarni
Vamsi Kodamasimhan

Graduate Students
Andres Quiroz
Baozhi Chen
Ciprian Docan
Eun Kyung Lee
Indraneel Kulkarni
John Paul Varkey
Mingliang Wang
Moustafa AbdelBaky
Richa Saraf
Shivangi Chaudhari
Siddharth Wagh
Yogesh Kakodkar

Selected Ongoing Projects

OOSP: Combining Stream Programming with Object Oriented Programming

by
Mingliang Wang

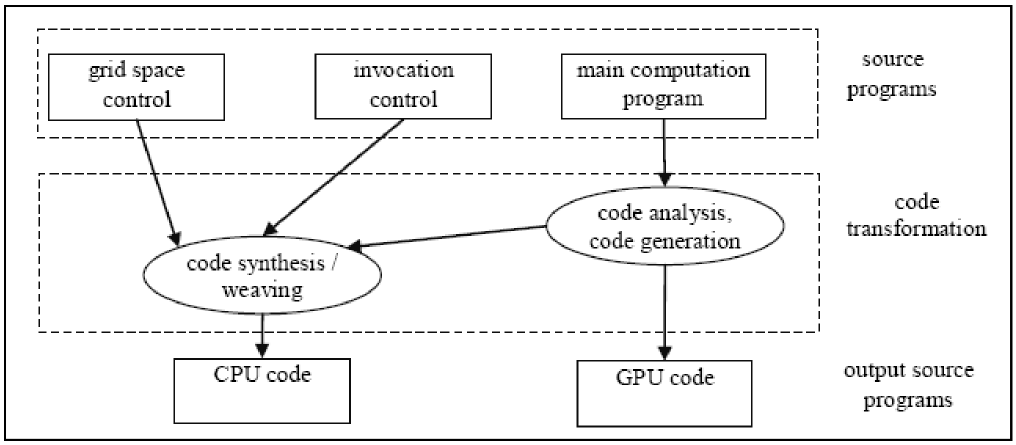
Principal Researchers:
Mingliang Wang
Manish Parashar

References:

M. McCool, “Scalable Programming Models for Massively Multicore Processors,” *Proceedings of the IEEE*, vol. 96, 2008, pp. 816-831.

E.A. Lee, “The Problem with Threads,” *Computer*, vol. 39, 2006, pp. 33-42.

As multicore processors are taking root, in a few years we will see mainstream computers with a few hundred to thousands of processing cores. This inevitable change towards massively parallel hardware requires a paradigm shift in software programs because existing sequential programs will not be able to perform better unless they are parallelized. Stream processing is a model of data parallel processing that decomposes data collections and relies on parallel execution of operations organized as kernels on the elements of a data collection. This model scales well into the multicore future as data collection is usually very large and can be simply divided into an appropriate number of chunks to match the number of cores.



Compiling OOSP programs for static binding to C++ with a CPU+GPU target

Stream Programming is conceptually simple and practically easy to implement. It can also be made deterministic because applications can entirely delegate synchronization and coordination to the runtime system, yielding a sequential view of a series of kernel invocations, which is familiar to programmers who write sequential programs. This helps avoid a lot of errors in parallel programs by construction.

In view of the promising model of stream processing and the need to make it accessible to mainstream programmers, who currently are predominately practitioners using C++, Java, or other object-oriented technologies, the authors propose a programming system, called OOSP, to provide native support for stream programming in these OO environments. OOSP exploits tagging interfaces to define stream processing semantics for member methods in object-oriented languages and relies on a source-source translator to generate conformant codes for further standard processing. Additionally, in the spirit of aspect-oriented programming, it factors out the aspects that depend on hardware architecture to allow for variations in vendors and/or evolution of hardware architecture, preparing for scaling into the future, so that investment in today’s development effort continue to benefit through simple scaling of hardware cores. Compared to other programming systems, such as Brook for GPU, RapidMind, CUDA, OpenCL, and Ct, the proposed the OOSP system enjoys the advantage of a high-level interface and abstraction, of native integration with OO paradigm, and of reliance on existing programming language constructs to express stream processing semantics.

The current language binding is C++ and we are implementing a source-source translation program that transforms C++ source codes into programs suitable for CUDA-enabled GPU accelerator boards. Dynamic language bindings would include Java and Python and will exploit the reflection mechanisms in these languages to realize stream processing semantics at runtime.

Autonomic Workload Provisioning for Enterprise Data Centers and Clouds

by
Andres Quiroz

Principal Researchers:
Manish Parashar
Andres Quiroz
Hyunjoo Kim

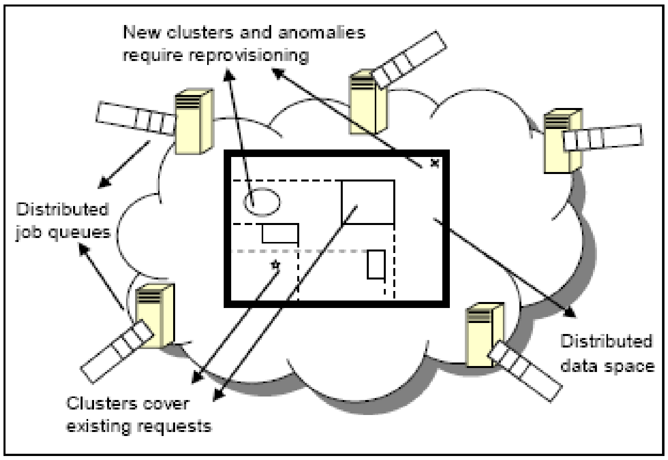
Current Collaborators:
Nathan Gnanasambandam

References:

"A Middleware Infrastructure for Content-based Decoupled Interactions in Pervasive Grid Environments," Concurrency and Computation: Practice and Experience, John Wiley and Sons, accepted 2007.

"Robust Clustering Analysis for the Management of Self-Monitoring Distributed Systems," A. Quiroz, N. Gnanasambandam, M. Parashar, and N. Sharma. In Journal of Cluster Computing, No. 12, pp. 73-85, Springer 2009, DOI: 10.1007/s10586-008-0068-5.

Consolidated and virtualized data centers provide virtual machine (VM) containers that host a wide range of user applications and provide users with an abstraction of unlimited computing resources - users can essentially "rent" VMs with required configurations to service their requests. As scales, operating costs, and energy requirements of such data centers increase, maximizing efficiency, cost-effectiveness, and utilization of these systems becomes paramount. Furthermore, enterprise data centers can have multiple geographically distributed entry points -- in a typical enterprise-wide data center, or a public cloud, there may exist multiple queues with a multitude of users submitting applications and requests to their individual local application queues, which are then scheduled along with other enterprise applications onto data center resources.



Procedure Overview

To this end, we explore an autonomic workload provisioning technique that is based on a decentralized, robust online clustering approach that specifically addresses the distributed nature of enterprise data centers and clouds. The approach builds on a decentralized messaging and data analysis infrastructure [1] that provides monitoring and density-based clustering capabilities [2]. By clustering workload requests across data center job queues to characterize different resource classes, it is possible to provide autonomic workload provisioning with the overall aim to improve data center utilization and efficiency. This approach has several advantages, including: (1) the capability of analyzing jobs across a dynamic set of distributed queues; (2) no requirement of a-priori knowledge of the number of classes, for example, as in *k*-means clustering; and (3) the amenity for online application and timely adaptation to changing workloads and resources.

Provisioning is meant to efficiently allocate resources to application jobs, as they are received by the job queue servers, through the creation of appropriately configured VM instances. Clustering is meant to balance over-provisioning and wait time costs by provisioning a pool of VMs in advance with respect to discovered resource clusters and assigning jobs to existing VMs. This can be done online and in a decentralized manner because the resource space that contains job descriptions is divided among a set of processing nodes, which can analyze the points within their region independently, only communicating with neighboring nodes to deal with boundary conditions. The exchange of job requests and the partitioning of the information space are supported by the content-based messaging substrate.

To realize the mechanism for dynamic and decentralized VM provisioning, we divide the flow of arriving jobs into *analysis windows*. In each window, an instance of the clustering algorithm is run on the jobs that arrive, producing a number of clusters or VM classes. At the same time, each job is assigned to an available VM as it arrives, if one has been provisioned with sufficient resources to meet its requirements. Each processing node locally triggers the creation of new VMs based on discovered clusters. In order to match jobs to provisioned VMs, cluster descriptions are distributed in the node network using the range of the clusters in the information space. Thus, when a new job arrives, it will be routed to a node that holds descriptors for VMs that have close resource requirements.

Note that the advantage of using online clustering is that analysis can be performed in each window, since clustering takes place as the jobs arrive and are being distributed in the network. In contrast, besides having limited scalability, centralized approaches require the data to be aggregated before the analysis can take place, which incurs greater wait time cost at the end of each window (or at the start of the next window).

by
Dario Pompili

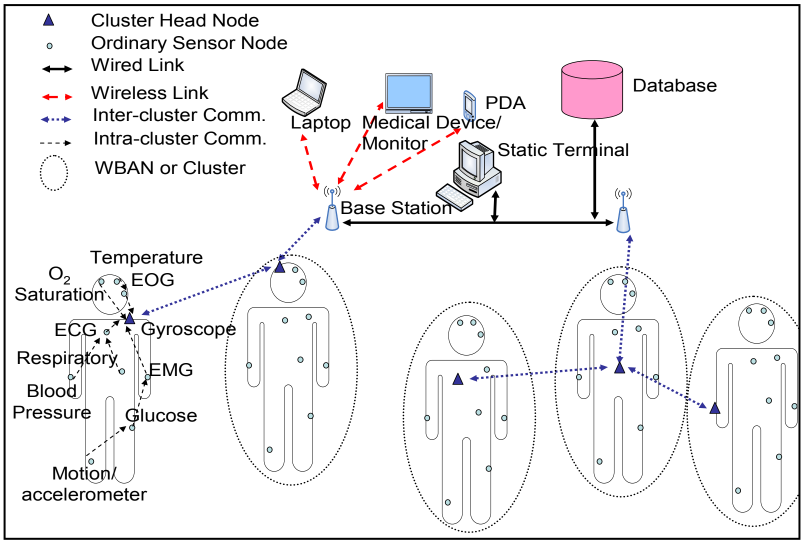
Principal Researchers:
Dario Pompili
Ivan Marsic
J.K-J Lee
R.S. Burd (CNMC)

References:

"Continuous Vital Sign Monitoring via Wireless Sensor Network," B. Chen, D. Pompili, I. Marsic, Malignant Spaghetti: Wireless Technologies in Hospital Health Care Workshop, NYU-Poly, Brooklyn, NY, November, 2008

Continuous Patient Vital Sign Monitoring Using Wireless Body Area Networks

Devices for monitoring patient vital signs are mostly wired, often depend on direct user interaction, have only partial analytic capability, require manual archiving even of digital data sources, and have limited capability to propagate data to the next destination on the patient's path. These issues are particularly critical in the prehospital setting, where accurate and reliable monitoring of patient's vital signs is important for making efficient and error-free triage decisions (i.e., classifying injured patients based on their need for or likely benefit from trauma center care). Emergency service (EMS) providers need to rapidly assess the injured patient and determine the need for trauma center care. Using existing technology, the EMS lacks effective methods for prioritizing information streams, evaluating time-dependent trends, managing incomplete data, and providing effective alerts. Current limitations of patient monitoring represent an important barrier for developing improved trauma triage strategies.



Wireless Information Patient System (WIPS)

To address these challenges, a new Wireless Information Patient System (WIPS) is proposed for monitoring in prehospital and hospital settings. WIPS will rely on wireless sensors that self-organize in a wireless body area network. Once deployed, the system will automatically and continuously 1) record vital signs (pulse, blood pressure, respiratory rate, etc.), 2) capture and disambiguate patient data that is now obtained by observation alone, and 3) issue alerts when critical deviations in vital-sign values are observed or predicted. The data will be transmitted reliably and securely via wireless communications to mobile portable devices, including PDAs, smart phones, and handheld computers. This project brings together technologies from different fields to solve real-world problems for injured patients that are now not being addressed with current technologies. To scale beyond a research project, we will facilitate interaction between engineers and clinicians and collect feedback about technology prototypes through on-site demonstrations. The current preclinical proposal will be the initial phase in a larger project that will lead to deployment and testing of the WIPS system in a clinical setting.

The research team is comprised of researchers at the Electrical and Computer Engineering and Biomedical Engineering Departments at Rutgers University and trauma physicians and nurses at Children's National Medical Center, and is uniquely positioned to address the technical and medical challenges of this project. The project aims at advancing knowledge by solving complex real-world problems that cross discipline boundaries. This research is potentially transformative because instead of incrementally improving individual technologies, it exploits the synergies of an interdisciplinary research group to improve the quality and accuracy of prehospital patient monitoring systems. This project will yield new methods for integrating and disambiguating multiplex data obtained in a dynamic and mobile setting. The specific application for trauma triage will have impact on the outcome of patients managed in a prehospital setting. In addition to advancing discovery and understanding, this project will promote teaching, training, and learning.

Accelerating Online Risk Analytics

by
Mukul Majmudar

Principal Researchers:
Mukul Majmudar
Ciprian Docan
Manish Parashar

References:

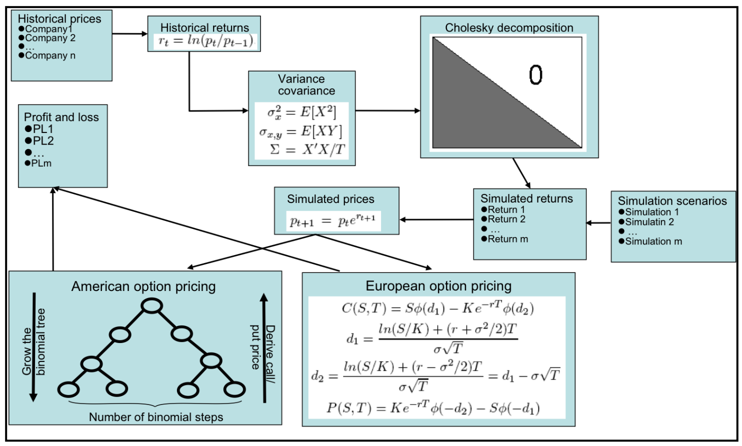
"Advanced Risk Analytics on the Cell Broadband Engine", C. Docan, C. Marty, M. Parashar, To appear in Proceedings of International Parallel and Distributed Processing Symposium (IPDPS'09), Rome, Italy, May 25-29, 2009.

Financial companies constantly strive to better understand and quantify risk. As the complexity and number of instruments firms trade increase, the ability to generate a comprehensive and consistent set of risk measures becomes computationally expensive. The tradeoff between increased cost and complexity of hardware versus careful and accurate risk measures has driven financial firms to look for innovative ways to decrease computing costs while, at the same time, increasing quality of risk measurements.

Value-at-Risk (VaR) is a market standard risk measure used by senior management and regulators to quantify the riskiness of a firms holdings. The VaR measure looks at the entirety of firms holdings at a confidence interval and time horizon, and reports an expected loss number. For example, a 1 Day 99% VaR number of $1 Million means that with 99% confidence, the firm holdings won't decrease in value by more than $1 Million over the next day. The non-linear nature of instrument pricing models and the requirement to preserve correlations of price movements make it difficult to develop a closed form solution to this problem. As such, large complex VaR calculations are often done using Monte-Carlo simulations. While these methods can effectively be parallelized, the capital cost of hardware plus the operational cost for data center space, power, cooling, and maintenance make their large scale usage cost prohibitive to all but the largest firms. However, the emergence of specialized hardware accelerators with more attractive cost/performance ratios can potentially lower this barrier and make high accuracy online risk analytics more accessible.

The overarching objective of this work is to autonomically outsource appropriate VaR kernels to appropriate accelerators to achieve more attractive cost/performance ratios. Specifically, it explores the online use of the IBM Cell Broadband Engine (CBE) processor and the NVIDIA GPU for accelerating option pricing algorithms for European and American options.

The parallel implementation of VaR for the Cell BE consists of defining separate routines for the PPU and the SPUs, and using the PPU as an orchestrator for the SPEs. The SPU code implements the two pricing algorithms, i.e., the Black-Scholes method for pricing European options and the binomial tree method for pricing American options (see Figure). The parallel implementations of these algorithms are used to process work units obtained from the PPU. The overall implementation used two levels of parallelism, (1) data parallelism across the work units and (2) instruction level parallelism within the SPU. Since SPU has limited memory and can only operate on blocks of data at a time, the data movement from the main memory to the SPU local memory must be optimized to maximize SPU efficiency. Our implementation uses a novel *triple buffering* technique that eliminates these stalls.



The option-pricing algorithms

The implementation on the NVIDIA GPU platform uses CUDA (Compute Unified Device Architecture). The key issue in implementing the tree-based Binomial option pricing model using CUDA is trading off memory access with redundant computations due to the limited shared memory available. The implemented solution does this by selectively storing intermediate results. In case of the Black-Scholes model, the parallelization of multiple scenarios on the GPU is straightforward, and a combination of CUDA streams and asynchronous *memcpy* is used to mask communication overheads,

The experimental evaluation used a portfolio with 96 stocks. The accelerated parallel implementations were compared with reference sequential implementations of the two algorithms. The Cell BE-based VaR application was evaluated on the IBM Q22 blade server with two PPU and 16 SPU cores. These experiments demonstrated that the parallel versions of the VaR pricing algorithms can achieve significant speedup using the SPU accelerators and can effectively utilize the Cell BE platform. On an NVIDIA 8400GS, peak speedups of 50 were achieved when using the Black-Scholes model (European options), and 22 when using the Binomial model (American options).

Note that the two platforms represent different price/parallelism/performance tradeoffs. The NVIDIA platform is less expensive (~$50), and has a larger number of processing units that are less powerful and have smaller amounts of memory per unit. On the other hand, the Cell BE is more expensive (~$300), has fewer processing units that are relatively more powerful and have more memory per unit. The Cell BE is also a full computing platform, while the NVIDIA platform requires a host platform.

Journal Publications

"AS Grid: Autonomic Management of Hybrid Sensor Grid Systems and Applications," X. Li, X. Liu, H. Zhao, H. Zhao, N. Jiang, M. Parashar, International Journal of Sensor Networks (IJSNet), InderScience Publishers, 2009.

"Robust Clustering Analysis for the Management of Self-Monitoring Distributed Systems," A. Quiroz*, N. Gnanasambandam, M. Parashar, and N. Sharma, Cluster Computing: The Journal of Networks, Software Tools, and Applications, Special Issue on Autonomic Computing, Kluwer Academic Publishers, (DOI:10.1007/s10586-008-0068-5), Vol. 12, No. 1, pp. 73-85, 2009.

Books published/edited

"Autonomic Communication," A. V. Vasilakos, M. Parashar, S. Karnouskos, W. Pedrycz, Spring Science-Business Media, estimated publication, August 2009.

"Advanced Computational Infrastructures for Parallel and Distributed Applications," M. Parashar, S. Chandra, X. Li, John Wiley and Sons, estimated publication in May 2009.

Conference/Workshop Publications

"Investigating Autonomic Behaviors in Grid-Based Computational Science Applications," S. Jha, M. Parashar and O. Rana, Grid Meets Autonomic Computing Workshop (GMAC '09), Proceedings of the 6th International Conference on Autonomic Computing and Communication, Barcelona, Spain, June 2009.

"Solving Sparse Linear Systems on NVIDIA Tesla GPUs," M. Wang, H. Klie, M. Parashar and H. Sudan, Proceeding of the Workshop on Using Emerging Parallel Architectures for Computational Science, in conjunction with the International Conference on Computational Science (ICCS 2009), Springer Verlag, Baton Rouge, Louisiana, May, 2009.

"Enabling End-to-end Data-driven Sensor-based Scientific and Engineering Applications," N. Jiang, M. Parashar, Proceeding of the Workshop on Dynamic Data-Driven Application Systems (DDDAS '09), in conjunction with the International Conference on Computational Science (ICCS 2009), Springer Verlag, Baton Rouge, Louisiana, May, 2009.

"Advanced Risk Analytics on the Cell Broadband Engine," C. Docan, M. Parashar and C. Marty, 2nd International Workshop on Parallel and Distributed Computing in Finance (PDCoF), in conjunction with the 23rd IEEE International Parallel and Distributed Processing Symposium (IPDPS 2009), Rome, Italy, IEEE Computer Society Press, May 2009.

"Enabling Autonomic Power-Aware Management of Instrumented Data Centers," N. Jiang and M. Parashar, 5th Workshop on High-Performance, Power-Aware Computing (HPPAC 2009), in conjunction with the 23rd IEEE International Parallel and Distributed Processing Symposium (IPDPS 2009), Rome, Italy, IEEE Computer Society Press, May 2009.

"Online Risk Analytics on the Cloud," H. Kim, S. Chaudhari, M. Parashar and C. Marty, Proceedings of the International Workshop on Cloud Computing (Cloud 2009), in conjunction with 9th IEEE/ACM International Symposium on Cluster Computing (CCGrid 2009), Shanghai, China, IEEE Computer Society Press, May 2009.

"In-network Data Estimation Mechanisms for Sensor-driven Scientific Applications," N. Jiang* and M. Parashar, Proceedings of the 15th IEEE International Conference on High Performance Computing (HiPC 2008), Bangalore, India, Editors: P. Sadayappan, M. Parashar, R. Badrinath, and V.K. Prasanna, LNCS, Vol. 5374, Springer-Verlag, December 2008. (Acceptance <15%).

Books Chapters

"High Throughput Data Movement," S. Klasky et al, "Scientific Data Management: Challenges, Existing Technologies, and Deployment," Editors: A. Shoshani and D. Rotem, Chapman and Hall, ISBN 9781420069808, 2009.

"Shared Memory Multiprocessors," L. Zhang* and M. Parashar, in "Wiley Encyclopedia of Computer Science and Engineering," Editor: B. Wah, John Wiley and Sons, Inc., ISBN: 978-0-471-38393-2, Volume 4, pp. 2522-2534, January 2009.

"Pervasive Grids: Challenges and Opportunities," M. Parashar and J-M Pierson, "Handbook of Research on Scalable Computing Technologies," Editors: K Li, C Hsu, Laurence T Yang, J. Dongarra and H Zima, Information Science Reference, IGI Global, ISBN: 978-1-60566-661-7, 2009.

Feathers in the 'CAC'

Viraj Bhat received the prestigious 2009 Grid Award at Yahoo!. The Grid Award is an yearly award that is presented within Yahoo! for evangelizing Grid (Hadoop, Pig, Zookeeper) technologies, profiling and optimizing Map Reduce, Pig applications within Yahoo!, migrating various mission critical applications which have very strict SLA requirements to the Grid and development of various Grid tools to help users migrate to the Grid. Viraj completed his Ph.D. under the guidance of Prof. M. Parashar in Spring 2008 and joined Yahoo! in May 2008.

Senior graduate student **Andres Quiroz** is co-chair of the Ph.D. Forum at the 6th IEEE International Conference on Autonomic Computing (ICAC 2009) (Barcelona, Spain, June 2009).

Nanyan Jiang recently completed her PhD under the guidance of Dr. Manish Parashar in Fall 2008 and joined Microsoft in January 2009.

CAC's Days Out