Workshop Program

Main Topics	Schedule	Speakers	Types of presentation	Titles (tentative)	Download
Diner	Sunday Nov. 21st 19:00	Radio Maria		http://www.radiomariarestaurant.com/	
Workshop Day 1 (Auditorium)	**Monday Nov. 22cd**				
Welcome and Introduction	08:30	Franck Cappello, INRIA & UIUC, France and Thom dunning, NCSA, USA	Background	Workshop details	
Post PetaScale and Exascale Systems, chair: Franck Cappello	08:45	Mitsuhisa Sato, U. Tsukuba, Japan	Trends in HPC	Challenges on Programming Models and Languages for Post-Petascale Computing -- from Japanese NGS project "The K computer" to Exascale computing --	INRIA-UIUC-WS4-msato.pdf
	09:15	Marc Snir, UIUC, USA	Trends in HPC	Toward Exascale	INRIA-UIUC-WS4-msnir.pdf
	09:45	Wen Mei Wu, UIUC, USA	Trends in HPC	Extreme-Scale Heterogeneous Computing	INRIA-UIUC-WS4-Hwu.pdf
	10:15	Arun Rodrigues, Sandia, USA	Trends in HPC	The UHPC X-Caliber Project	INRIA-UIUC-WS4-arodrigues.pdf
	10:45	Break			
Post Petascale Applications and System Software chair: Marc Snir	11:15	Pete Beckman, ANL, USA	Trends in HPC	Exascale Sofware Center	INRIA-UIUC-WS4-pbeckman.pdf
	11:45	Michael Norman, SDSC, USA	Trends in HPC	Extreme Scale AMR for Hydrodynamic Cosmology	INRIA-UIUC-WS4-mnorman.pptx
	12:15	Eric Bohm, UIUC, USA	Trends in HPC	Scaling NAMD into the Petascale and Beyond	INRIA-NCSA_WS4_ebohm.pdf
	12:45	Lunch			
BLUE WATERS , chair Bill Gropp	14:00	Bill Kramer, NCSA, USA	Overview	Blue Waters: A Super-System to Explore the Expanse and Depth of 21st Century Science	INRIA-UIUC-WS4-bkramer2.pdf
Collaborations on System Software	14:30	Ana Gainaru, NCSA, USA	Early Results	Framework for Event Log Analysis in HPC	INRIA-UIUC-WS4-againaru.pdf
Steve Gottlieb	15:00	Esteban Menese, UIUC, USA	Early Results	Clustering Message Passing Applications to Enhance Fault Tolerance Protocols	INRIA-UIUC-WS4-emenese.pdf
	15:30	Thomas Ropars, INRIA, France	Results	Latest Progresses on Rollback-Recovery Protocols for Send-Deterministic Applications	INRIA-UIUC-WS4-tropars.pdf
	16:00	Break			
Collaborations on System Software, chair: Bill Kramer	16:30	Leonardo Bautista, Titech, Japan	Results/International collaboration with Japan	Transparent low-overhead checkpoint for GPU-accelerated clusters	INRIA-UIUC-WS4-lbautista.pdf
	17:00	Gabriel Antoniu, INRIA/IRISA, France	Results	Concurrency-optimized I/O for visualizing HPC simulations: An Approach Using Dedicated I/O cores	INRIA-UIUC-WS4-gantoniu.pdf
	17:30	Mathias Jacquelin, INRIA/ENS Lyon	Results	Comparing archival policies for BlueWaters	INRIA-UIUC-WS4-mjacquelin.pdf
	18:00	Olivier Richard, Joseph Emeras, INRIA/U. Grenoble, France	Early Results	Studying the RJMS, applications and File System triptych: a first step toward experimental approach	INRIA-NCSA-WS4-jemeras.pdf
Diner	19:30	Gould's		http://www.jimgoulddining.com/	
Workshop Day 2 (Auditorium)	**Tuesday Nov. 23rd**				
Collaborations on System Software, chair: Raymond Namyst	08:30	Torsten Hoefler, NCSA, USA	Potential collaboration	Application Performance Modeling on Petascale and Beyond	INRIA-UIUC-WS4-thoefler.pdf
	09:00	Frederic Viven, INRIA/ENS Lyon, France	Potential collaboration	On Scheduling Checkpoints of Exascale Application	INRIA-UIUC-WS4-fvivien.pdf
Collaborations on Programming models,	09:30	Thierry Gautier	Potential collaboration	On the cost of managing data flow dependencies for parallel programming	INRIA-UIUC-WS4-tgautier.pdf
	10:00	Laercio Pilla, INRIA/U. Grenoble, France	Early Results	Charm++ on NUMA Platforms: the impact of SMP Optimizations and a NUMA-aware Load Balancing	INRIA-UIUC-WS4-llpilla.pdf
	10:30	Break			
chair: Sanjay Kale	11:00	Raymon Namyst, INRIA/U. Bordeaux, France	Potential collaboration	Bridging the gap between runtime systems and programming languages on heterogeneous GPU clusters	INRIA-UIUC-WS4-rnamyst.pdf
	11:30	Brian Amedo, INRIA/U. Nice, France	Potential collaboration	Improving asynchrony in an Active Object model	INRIA-UIUC-WS4-bamedro.pdf
	12:00	Christian Perez, INRIA/ENS Lyon, France	Early Results	High Performance Component with Charm++ and OpenAtom	INRIA-UIUC-W54-cperez.pdf
	12:30	Lunch			

Collaborations on Numerical Algorithms and Libraries, chair Mitsuhisa Sato	14:00	Luke Olson, UIUC, USA	Early Results	On the status of algebraic (multigrid) preconditioners	INRIA-UIUC-WS4-lolson.pdf
	14:30	Simplice Donfac, INRIA/U. Paris Sud, France	Early Results	Improving data locality in communication avoiding LU and QR factorizations	INRIA-UIUC-SW-sdonfack.pdf
	15:00	Desiré Nuentsa, INRIA/IRISA, France	Early Results	Parallel Implementation of deflated GMRES in the PETSc package	INRIA-UIUC-WS4-dnuentsa.pdf
	15:30	Sebastien Fourestier, INRIA/U. Bordeaux, France	Early Results	Graph repartitioning with Scotch and other on going work	INRIA-UIUC_WS4-fourestier.pdf
	16:00	Break			
chair: Luke Olson	16:15	Marc Baboulin, INRIA, U. Paris Sud, France	Early Results	Accelerating linear algebra computations with hybrid GPU-multicore systems	INRIA-UIUC-WS4-mbaboulin.pdf
	16:45	Daisuke Takahashi, U. Tsukuba, Japan	Results/International collaboration with Japan	Optimization of a Parallel 3-D FFT with 2-D Decomposition	INRIA-NCSA-WS4-dtakahashi.pdf
	17:15	Alex Yee, UIUC, USA	Early Results	A Single-Transpose implementation of the Distributed out-of-order 3D-FFT	INRIA-UIUC-WS4-ayee.pdf
	17:35	Jeongnim Kim, NCSA, USA	Early Results	Toward petaflop 3D FFT on clusters of SMP	INRIA-NCSA-WS4jkim.pdf
Diner	19:30	Escobar's		http://www.escobarsrestaurant.com/	
Workshop Day 3 (Auditorium)	**Wednesday Nov 24th**				
Break out sessions introduction	8:30	Cappello, Snir	Overview	Objectives of Break-out, expected results Collaborations mechanisms (internship, visits, etc.)	
Topics		Participants	Other NCSA participants		
Break out session 1	9:00-10:15				
Routing, topology mapping, scheduling, perf. modeling		**Snir**, Hoefler, Vivien, Gautier, Kale, Namyst, Méhaut, Bohm, Pilla, Amedo, Perez, Baboulin		Room 1030	Break-out-report-snir.pdf
Resilience		**Kramer**, Cappello, Gainaru, Ropars, Menese, Bautista, Antoniu, Richard, Fourestier, Jacquelin		Room 1040	Break-out-report-kramer.pdf
Libraries		**Olson**, Désiré, Simplice,		Room 1104	
	10:15	Break			
Break out session 2	10:30-11:45				
Programing models / GPU		**Kale**, Méhaut, Namyst, Wu, Amedo, Perez, Bohm, Pilla, Baboulin, Fourestier, Gautier		Room 1030	
I/O		**Snir**, Viven, Jaquelin, Antoniu, Richard, Kramer, Gainaru, Ropars		Room 1040	Break-out-report-snir.pdf
3D-FFT		**Cappello**, Takahashi, Yee, Jeongnim, Hoefler		Room 1104	Break-out-3D-FFT-cappello.pdf
Break out session report	12:00	Speakers: Snir, Cappello, Kramer, Kale, Olson		Auditorium	
Closing	12:30	Cappello, Snir		Auditorium	
	13:00	Lunch			
Diner	19:00	Buttitta's		http://buttittascu.com/	

Abstracts

Mitsuhisa Sato, University of Tsukuba

Challenges on Programming Models and Languages for Post-Petascale Computing
-- from Japanese NGS project "The K computer" to Exascale computing --

Post-petascale systems and future exascale computers are expected to have an ultra large-scale and highly hierarchical architecture with nodes of many-core processors and accelerators. That implies that existing systems, language, programming paradigms and parallel algorithms should be reconsidered. To manage these ultra large-scale parallel systems, we require new adaptive runtime systems, allowing to manage huge distributed data, minimizing the energy consumption, and with fault resilient properties. Moreover, accelerating technology such as GPGPU and many-core processors, is a crucial domain for post petascale computing. Their efficient programming in these large scale systems is also an important challenge. In this talk, I will talk about issues and challenges of proramming models for post-petascale computing with the status of Japanese NGS project and plans for exascale computing in future.

Marc Snir, UIUC

The talk will position exascale research in the context of the pending slow-down in the exponential increase in chip densities and discuss some fundamental research problems that need to be addressed in order to reach exascale performance at reasonable expense.

Wen Mei Whu, UIUC

Extreme-Scale Heterogeneous Computing

I will give an update on our recent progress in extreme-scale heterogeneous computing. In the area of applications, the UIUC applications community has been busy developing and releasing new libraries and application packages for GPU computing. In the tools area, new CUDA and OpenCL performance analysis, optimization, portability, and benchmarking tools are under active development with funding from DOE, NIH, NSF, and industry. In the area of algorithm design, we are teaching a new graduate course on eight widely applicable algorithm optimization strategies for developing successful data parallel algorithms. In the area of system development, several UIUC departments and centers recently collaborated with NVIDIA to construct the 128-node EcoG GPU cluster that achieves dramatic improvement in energy efficiency over previous Green500 clusters. There are ample opportunities for collaboration in all four areas.

Arun Rodrigues, Sandia

The UHPC X-Caliber Project

Reaching the goal of exascale will require massive improvements in hardware performance, power consumption, software scalability, and usability. To address these issues Sandia National Labs is leading the UHPC X-Caliber Project, with the goal of producing a prototype single cabinet capable of a sustained petaflop by 2018. Our approach focuses on solving the data movement problem with advanced memory and silicon photonics, enabled by advances in fabrication and 3D packaging, and unified by a new execution model. I present the current X-Caliber architecture, our design space, and the co-design philosophy which will guide the project.

Michael Norman, SDSC

Extreme Scale AMR for Hydrodynamic Cosmology

Cosmological simulations present well-known difficulties scaling to large core counts because of the large spatial inhomogeneities and vast range of length scales induced by gravitational instability. These difficulties are compounded when baryonic physics is included which introduce their own multiscale challenges. In this talk I review efforts to scale the Enzo adaptive mesh refinement hydrodynamic cosmology code to O(100,000) cores, and I also discuss Cello--an extremely scalable AMR infrastructure under development at UCSD for the next generation of computer architectures which will underpin petascale Enzo.

Eric Bohm, NCSA

Scaling NAMD into the Petascale and Beyond

Many challenges arise when employing ever larger supercomputers for the simulation of biological molecules in the context of a mature molecular dynamics code. Issues stemming from the scaling up of problem size, such as input and output require both parallelization and revisions to legacy file formats. Order of magnitude increases in the number of processor cores evoke problems with O(P) structures, load balancing, and performance analysis. New architectures present code optimization opportunities (VSX SIMD) which must be carefully applied to provide the desired performance improvements without dire costs in implementation time and code quality. Looking beyond these imminent concerns for sustained petaflop performance on Blue Waters, we will also consider scalability concerns for future exascale machines.

Bill Kramer, NCSA

Blue Waters: A Super-System to Explore the Expanse and Depth of 21st Century Science

While many people think that Blue Waters means a single Power7 IH supercomputer, in reality, the Blue Waters Project is deploying an entire system architecture that includes an eco-system surrounding the Power7 IH system to make it highly effective, ultra-scale science and engineering. This is what we term the Blue Waters "Super System" which we will describe in detail in this talk along with its corresponding service architecture.

Ana Gainaru, UIUC/NCSA

Framework for Event Log Analysis in HPC

In this talk, we present a fault analysis framework that combines different event analysis modules. We present the clustering module that extracts message patterns from log files. We also describe how looking at event repetitions as signals could help system administrators mine information about failure causes post-mortem or even how it could help the system take proactive measurement. The modules are working in a pipeline manner, the first one feeding event templates to the second module, which is used to decide if the event signals are periodic, partial periodic or noise. We analyse if a change in the characteristics of an event influences modifications in other signals.

Thomas Ropars, INRIA

Latest Results in Rollback-Recovery Protocols for Send-Deterministic Applications.

In very large scale HPC systems, rollback-recovery techniques are mandatory to ensure applications correct termination despite failures. Nowadays coordinated checkpointing is almost always used, mainly because it is simple to implement and use. However coordinated checkpointing has several drawbacks: i) at checkpoint time, it is stressing the file system when the image of all application processes have to be saved "at the same time"; ii) recovery is energy consuming because a single failure makes all application processes rollback to their last checkpoint. Alternatives based on message logging have never been widely adopted mainly because of the additional cost induced by logging all the messages during failure free execution. It has been shown that most MPI HPC applications are send-deterministic, i.e. that the sequence of message sendings in an execution is deterministic. We are trying to take advantage of this property to design new rollback-recovery protocols overcoming the limits of existing approaches. In this talk, we first present an uncoordinated checkpointing protocol that does not suffer from the domino effect, while logging only a small subset of the application messages. We present experimental results showing its good performance on failure free execution on an high performance network. We also show how applying process clustering to this protocol can limit the number of processes to rollback in the event of a single failure to 50% on average. Then we introduce a new protocol combining cluster-based coordinated checkpointing and inter-cluster message logging to further reduce the amount of rolled-back computation after a failure.

Esteban Meneses, UIUC

Clustering Message Passing Applications to Enhance Fault Tolerance Protocols

This talk describes the effort of an ongoing collaboration to find meaningful clusters in a parallel computing application using its communication behavior. We start by showing the communication pattern of various MPI benchmarks and how we can use standard graph partitioning techniques to group the ranks into subsets. For Charm++ applications, we describe the changes on the runtime system to dynamically find the clusters even in the presence of object migration. The information about clusters is used to improve two major message logging protocols for fault tolerance. In one case, we manage to reduce its memory overhead, while in the other we are able to limit the number of processes to roll back during recovery.

Leonardo Bautista, Titech

Transparent low-overhead checkpoint for GPU-accelerated clusters

Fast checkpointing will be a necessary feature for future large-scale systems. Particularly, large GPU-accelerated systems lack of an efficient checkpoint-restart mechanism able to checkpoint CUDA applications in a transparent fashion (without code modification). Most of current fault tolerance techniques do not support CUDA applications or have severe limitations. We propose a transparent low-overhead checkpointing technique for GPU accelerated clusters that avoid the I/O bottleneck by using erasure codes and SSDs on the compute nodes. We achieve this by combining mature production tools, such as BLCR and OpenMPI, with our previous work and some new developed components.

Mathias Jacquelin, INRIA/ENS Lyon

Comparing archival policies for BlueWaters

In this work, we introduce two archival policies tailored for the tape storage system that will be available on BlueWaters. We also show how to adapt the well known RAIT strategy (the counterpart
of RAID policy for tapes) for BlueWaters. We provide an analytical model of the tape storage platform of BlueWaters, and use it to asses and analyze the performance of the three policies through simulations. We use random workloads whose characteristics model various realistic scenarios. The throughput of the system, as well as the average (weighted) response time for each user, are the main objectives.

Gabriel Antoniu, INRIA/IRISA

Concurrency-optimized I/O for visualizing HPC simulations: An Approach Using Dedicated I/O cores

Research at the Joint INRIA-UIUC Lab for Petascale computing is currently in progress in several directions, with the global goal of efficiently exploiting this machine that whill serve to run heavy, data-intensive or computation-intensive simulations. Such simulations usually require to be coupled with visualization tools. On supercomputers, previous studies already showed the need of adapting the I/O path from data generation to visualization. We focus on a particular tornado simulation that is intended to be run on BlueWaters. This simulation currently generates large amount of data in many files, in a way that is not adapted for afterward visualization. We describe an approach to this problem based on the usage of dedicated I/O cores. As a further step, we intend to explore the use of BlobSeer, a large-scale data management service, as an intermediate layer between the simulation, the filesystem and visualization tools. We propose to go further in this approach by enabling BlobSeer to run on dedicated cores and schedule I/O operations coming from the simulation.

Olivier Richard, Joseph Emeras, INRIA/U. Grenoble

Studying the RJMS, applications and File System triptych: a first step toward experimental approach

In a High Performance Computing infrastructure, it is particularly difficult to master the architecture as a whole. With the physical infrastructure, the platform management software and the users' applications, understanding the global behavior and diagnosing problems is quite challenging. And it is even more true in a petascale context with thousands of compute nodes to manage and a high occupation rate of the resources. A global study of the platform will thus consider the Resource and Job Management System (RJMS), the File System and the Applications triptych as a whole. Studying their behavior is complicated because it means having some knowledge of the applications requirements in terms of physical resources and access to the File System. In this presentation, we propose a first step toward an experimental approach that mix the use of Jobs Workloads patterns and File System access patterns that, once combined, will give a full set of jobs behaviors. These synthetic jobs will then be used to test and benchmark infrastructure, considering the RJMS and the File System.

Torsten Hoefler, NCSA

Application Performance Modeling on Petascale and Beyond

Performance modeling of parallel application is gaining more importance. It can not only help to predict scalability and find performance bottlenecks but it can also help to understand trade-offs in the design space of computing systems and drive hardware-software co-design of future computing systems. We will discuss established performance modeling techniques and propose a mixed approach to analytic application performance modeling. We then discuss open problems and possible future research directions.

Frédéric Viven, INRIA/ENS Lyon

On scheduling the checkpoints of exascale applications

Checkpointing is one of the tools used to provide resilience to applications run on failure-prone platforms. It is usually claimed that checkpoints should occur periodically, as such a policy is optimal. However, most of the existing proofs rely on approximations. One such assumption is that the probability that a fault occurs during the execution of an application is very small, an assumption that is no longer valid in the context of exascale platforms. We have begun studying this problem in a fully general context. We have established that, when failures follow a Poisson law, the periodic checkpointing policy is optimal. We have also showed an unexpected result: in some cases, when the platform is sufficiently large, the checkpointing costs sufficiently expensive, or the failures frequent enough, one should limit the application parallelism and duplicate tasks, rather than fully parallelize the application on the whole platform.

Jean-François Mehaut INRIA/U. Grenoble

Charm++ on NUMA Platforms: the impact of SMP Optimizations and a
NUMA-aware Load Balancing

Cache-coherent Non-Uniform Memory Access (ccNUMA) platforms based on multi-core chips are now a common resource in High Performance Computing. To overcome scalability issues in such platforms, the shared memory is physically distributed among several memory banks. Its memory access costs may vary depending on the distance between processing units and data. The main challenge of a ccNUMA platform is to manage efficiently threads, data distribution and communication over all the machine nodes. Charm++ is a parallel programming system that provides a portable programming model for platforms based on shared and distributed memory. In this work, we revisit some of the implementation decisions currently featured on Charm++ on the context of ccNUMA platforms. First, we studied the impact of the new -- shared-memory based -- inter-object communication scheme utilized by Charm+. We show how this shared-memory approach can impact the performance of Charm+ on ccNUMA machines. Second, we conduct a performance evaluation of the CPU and memory affinity mechanisms provided by Charm++ on ccNUMA platforms. Results show that SMP optimizations and affinity support can improve the overall performance of our benchmarks in up to 75%. Finally, in light of these studies, we have designed and implemented a NUMA-aware load balancing algorithm that addresses the issues found. The performance evaluation of our prototype showed results as good as the ones obtained by GreedyLB and significant improvements when compared to GreedyCommLB.

Thierry Gautier INRIA

On the cost of managing data flow dependencies for parallel programming.

Several parallel programming languages or libraries (TBB, Cilk+, OpenMP) allows to spawn independent tasks at runtime. In this talk, I will give an overview of the work about the Kaapi runtime system and its management of dependencies between tasks scheduled by a work stealing algorithm. I will show you that at a lower cost than TBB or Cilk+, it is possible to program with data flow dependencies.

Raymond Namyst INRIA/Univ. Bordeaux

Bridging the gap between runtime systems and programming languages on heterogeneous GPU clusters

In this talk, I will give an overview of our recent work about the StarPU runtime system. I will also present a number of extensions that leverage StarPU and bridge the gap with programming environments such as OpenCL or StarSuperscalar, and which provide better integration potential with programming standards such as MPI, OpenMP, etc.

Christian Perez INRIA/ENS Lyon

High Performance Component with Charm++ and OpenAtom

Software component models appear as a solution to handle the complexity and the evolution of applications. It turns out to be a powerful abstraction mechanism for dealing with parallel and heterogeneous machines as it enable the structure of an application to be manipulated, and hence specialized. HLCM is a hierarchical component model with support for genericity & connector that enables to adapt an application to the resources as well as to input parameters. HLCM is an abstract model as it does not depend on on a particular primitive component implementation. This talk will present our ongoing work on defining and implementing HLCM/Charm+, a specialization of HLCM with primitive component expressed in Charm. It will also provide information on a study on the benefits HLCM/Charm+ can bring to OpenAtom.

Luke Olson UIUC

On the status of algebraic (multigrid) preconditioners

In this talk we highlight some recent progress in extending algebraic multigrid preconditioning to more complex applications, focussing on high-order and non-conforming discretizations, and highlighting success on accelerated platforms (GPU). We also present several challenges where research on algebraic methods would be enhanced through collaboration in the Joint Lab.

Simplice Donfac INRIA/Univ. Paris Sud

Improving data locality in communication avoiding LU and QR factorizations

In previous work we have shown that communication avoiding algorithms combined with dynamic scheduling lead to good performance on multicore architectures. In this work, we investigate the possibility of combining static and dynamic scheduling as well as improving data locality in communication avoiding LU and QR factorizations. We evaluate the performance obtained on a single node of power 5 and power 7 machines.

Brian Amedro INRIA/Univ. Nice

Improving asynchrony in an Active Object model

This work is applied on the Asynchronous Sequential Process (ASP) model, and its Java implementation ProActive, a middleware for parallel and distributed computing. We investigate the automatic introduction of asynchronous channels between activities, while maintaining the causal ordering of messages. In order to do that, we will use informations provided by the programmer on the nature of messages and on the behavior of the program. With some language constructs and general middleware techniques, we can optimize communications while ensuring that no causal ordering between messages will be lost.

Sebastien Fourestier INRIA/U. Bordeaux

Graph repartitioning with Scotch and other on going work.
Sebastien Fourestier

Scotch is a software package for sequential and parallel graph partitioning, static mapping, sparse matrix block ordering, and sequential mesh and hypergraph partitioning. As a research project, it is subject to continuous improvement, resulting from several on-going research tasks. Our talk will focus on two of them: graph repartitioning, which was not previously addressed, and scalability concerns. We will also briefly present other ongoing work, in the context of our new roadmap.

Marc Baboulin INRIA/Univ. Paris Sud

Accelerating linear algebra computations with hybrid GPU-multicore systems

We describe how hybrid multicore+GPU systems can be used to enhance performance of linear algebra libraries in high performance computing.
We illustrate this approach with the solution of general linear systems based on a hybrid LU factorization where we split the computation over a multicore and a graphic processor, and use particular statistical techniques to reduce the amount of pivoting and communication between the hybrid components.
We also show how mixed precision algorithms can be used for accelerating performance.

Désiré Nuentsa_wakam INRIA/IRISA

Parallel Implementation of deflated GMRES in the PETSc package

The deflation process is effective to prevent stagnation in the GMRES iterative method. However, it induces extra operations as the spectral information should be computed during each restart. In this work, we develop an adaptive strategy that switchs to the deflated version when the stagnation is detected in the iterative process. Then we provide a parallel implementation as a new KSP type in the PETSc package. Several tests are performed to show the usefulness of this approach on real applications.

Daisuke Takahashi, U. Tsukuba

Optimization of a Parallel 3-D FFT with 2-D Decomposition

In this talk, an optimization method for parallel 3-D fast Fourier transform (FFT) with 2-D decomposition is presented.The 2-D decomposition effectively improves performance by reducing the communication time for larger numbers of MPI processes. The another way to reduce the communication overhead is to overlap communication and computation. An overlapping method for the parallel 3-D FFT is also presented. Performance results of parallel 3-D FFTs on clusters of multi-core processors are reported.

Alex Yee, UIUC

A Single-Transpose implementation of the Distributed out-of-order 3D-FFT

The classic approach to computing the distributed in-order 3D-FFT requires up to 3 expensive all-to-all communication transpose steps. Given the memory-bound nature of the FFT, these transposes are dominant factors in the total run-time. Here we present a new approach that reduces the number of transposes to 2 for the in-order transform, and 1 for the out-of-order transform.

Jeongnim Kim, NCSA, UIUC

Toward petaflop 3D FFT on clusters of SMP

A wide range of scientific applications employs 3D FFT. Sustained petaflop performance of 3D FFT is necessary to meet the NSF Direct Numerical Simulation (DNS) turbulence benchmark on the Blue Waters which represents the current generation of HPC platforms, clusters of multi/many-core SMPs. I present the analysis of 3D FFT implementations and the optimization strategies on the BW. Also discussed is the design of parallel 3D FFT library that can meet the diverse requirements of applications using 3D FFT.