Report from Dagstuhl Seminar 15281

Algorithms and Scheduling Techniques to Manage Resilience and Power Consumption in Distributed Systems

Edited by

Henri Casanova[1], Ewa Deelman[2], Yves Robert[3], and Uwe Schwiegelshohn[4]

1 **University of Hawaii at Manoa, US,** henric@hawaii.edu
2 **University of Southern California, Marina del Rey, US,** deelman@isi.edu
3 **ENS Lyon, FR, & University of Tennessee – Knoxville, US,** Yves.Robert@ens-lyon.fr
4 **TU Dortmund, DE,** uwe.schwiegelshohn@udo.edu

Abstract

Large-scale systems face two main challenges: failure management and energy management. Failure management, the goal of which is to achieve resilience, is necessary because a large number of hardware resources implies a large number of failures during the execution of an application. Energy management, the goal of which is to optimize of power consumption and to handle thermal issues, is also necessary due to both monetary and environmental constraints since typical applications executed in HPC and/or cloud environments will lead to large power consumption and heat dissipation due to intensive computation and communication workloads.

The main objective of this Dagstuhl seminar was to gather two communities: (i) system-oriented researchers who study high-level resource-provisioning policies, pragmatic resource allocation and scheduling heuristics, novel approaches for designing and deploying systems software infrastructures, and tools for monitoring/measuring the state of the system; and (ii) algorithm-oriented researchers, who investigate formal models and algorithmic solutions for resilience and energy efficiency problems. Both communities focused around workflow applications during the seminar, and discussed various issues related to the efficient, resilient, and energy efficient execution of workflows in distributed platforms.

This report provides a brief executive summary of the seminar and lists all the presented material.

Seminar July 6–10, 2015 – http://www.dagstuhl.de/15281
1998 ACM Subject Classification E.1 Data Structures, C.2.4 Distributed Systems, C.1.4 Parallel Architectures
Keywords and phrases Fault tolerance, Resilience, Energy efficiency, Distributed and high performance computing, Scheduling, Workflows
Digital Object Identifier 10.4230/DagRep.5.7.1

Algorithms and Scheduling Techniques to Manage Resilience and Power Consumption in Distributed Systems, *Dagstuhl Reports*, Vol. 5, Issue 7, pp. 1–21
Editors: Henri Casanova, Ewa Deelman, Yves Robert, and Uwe Schwiegelshohn
Dagstuhl Reports
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Executive Summary

Henri Casanova
Ewa Deelman
Yves Robert
Uwe Schwiegelshohn

Many computer applications are executed on large-scale systems that comprise many hardware components, such as clusters that can be federated into distributed cloud computing or grid computing platforms. The owners/managers of these systems face two main challenges: failure management and energy management.

Failure management, the goal of which is to achieve resilience, is necessary because a large number of hardware resources implies a large number of failures during the execution of an application. While hardware resources can be made more reliable via the use of redundancy, this redundancy increases component cost. As a result, systems deployed within budget constraints must be built from unreliable components, that have a finite Mean Time Between Failure (MTBF), i.e., commercial-of-the-shelf components. For instance, a failure would occur every 50 minutes in a system with one million components, even if the MTBF of a single component is as large as 100 years.

Energy management, the goal of which is to optimize power consumption and to handle thermal issues, is also necessary due to both monetary and environmental constraints. While in today's systems, processors are the most power-consuming components, it is anticipated that in future distributed systems, the power dissipated to perform communications and I/O transfers will make up a much larger share of the overall energy consumption. In fact, the relative cost of communication is expected to increase dramatically, both in terms of latency/overhead and of consumed energy. Consequently, the computation and communication workloads of typical applications executed in HPC and/or cloud environments will lead to large power consumption and heat dissipation.

These two challenges, resilience and energy efficiency, are currently being studied by many researchers. Some of these researchers come from a "systems" culture, and investigate in particular systems design and management strategies that enhance resilience and energy efficiency. These strategies include high-level resource-provisioning policies, pragmatic resource allocation and scheduling heuristics, novel approaches for designing and deploying systems software infrastructures, and tools for monitoring/measuring the state of the system. Other researchers come from an "algorithms" culture. They investigate formal definitions of resilience and energy efficiency problems, relying on system models of various degrees of accuracy and sophistication, and aiming to obtain strong complexity results and algorithmic solutions for solving these problems. These two communities are quite often separated in the scientific literature and in the field. Some of the pragmatic solutions developed in the former community appear algorithmically weak to the latter community, while some of the algorithmic solutions developed by the latter community appear impractical to the former community. Furthermore, the separation of application and system platform due to ubiquitous resource virtualization layers also interferes with an effective cooperation of algorithmic and system management methods, and in particular to handle resiliency and energy efficiency. To move forward, more interaction and collaboration is needed between the systems and the algorithms communities, an observation that was made very clear during the discussions in the predecessor Dagstuhl seminar [1].

The broader challenge faced by systems and algorithms designer is that the optimization metrics of interest (resilience, power consumption, heat distribution, performance) are intimately related. For instance, high volatility in power consumption due to the use of dynamic frequency and voltage scaling (DFVS) is known to lead to thermal hotspots in a datacenter. Therefore, the datacenter must increase the safety margin for their cooling system to handle these hotspots. As a result, the power consumed by the cooling system is increased, possibly increasing the overall power consumption of the whole system, even though the motivation for using DVFS in the first place was to reduce power consumption! When resilience is thrown into the mix, then the trade-offs between the conflicting resilience, performance, and energy goals become even more intertwined. Adding fault-tolerance to a system, for instance, by using redundant computation or by periodically saving the state of the system to secondary storage, can decrease performance and almost always increases hardware resource requirements and thus power consumption. The field is rife with such conundrums, which must be addressed via systems and algorithms techniques used in conjunction. In this seminar, we have brought together researchers and practitioners from both the systems and the algorithms community, so as to foster fruitful discussions of these conundrums, many of which were touched upon in the predecessor seminar but by no means resolved.

To provide a clear context, the seminar focused around workflow applications. Workflows correspond to a broad and popular model of computation in which diverse computation tasks (which many themselves follow arbitrary models of computation) are interconnected via control and data dependencies. They have become very popular in many domains, ranging from scientific to datacenter applications, and share similar sets of challenges and current solutions. Part of the motivation of using workflows, and thus to develop workflow management systems and algorithms, is that they make it possible to describe complex and large computations succinctly and portably. Most of the invited seminar participants have worked and are currently working on issues related to the efficient, resilient, and energy efficient execution of workflows in distributed platforms. They thus provide an ideal focal and unifying theme for the seminar.

A number of workflow tools is available to aid the users in defining and executing workflow applications. While these tools are thus designed primarily to support the end user, they are in fact ideal proving grounds for implementing novel systems and algorithms techniques to aim at optimizing performance, resilience, and energy efficiency. Therefore, these tools provide a great opportunity to enhance both the application and the software infrastructure to meet both the needs of the end users and of the systems owners/managers. These goals are very diverse and, as we have seen above, intertwined, so that re-designing algorithms and systems to meet these goals is a difficult proposition (again, higher resilience often calls for redundant computations and/or redundant communication, which in turn consumes extra power and can reduce performance). In a broad sense, we are facing complex multi-criteria optimization problems that must be (i) formalized in a way that is cognizant of the practical systems constraints and hardware considerations; (ii) solved by novel algorithms that are both fast (so that they can be used in an on-line manner) and robust (so that they can tolerated wide ranges of scenarios with possibly inaccurate information).

The goal of this seminar was to foster discussions on, and articulate novel and promising directions for addressing the challenges highlighted above. International experts in the field

[1] Dagstuhl Seminar 13381, *Algorithms and Scheduling Techniques for Exascale Systems* (2013), organized by Henri Casanova, Yves Robert and Uwe Schwiegelshohn (http://www.dagstuhl.de/13381).

have investigated how to approach (and hopefully at least partially address) the challenges that algorithms and system designers face due to frequent failures and energy usage constraints. More specifically, the seminar has addressed the following topics:

- Multi-criteria optimization problems as applicable to fault-tolerance / energy management
- Resilience techniques for HPC and cloud systems
- Robust and energy-aware distributed algorithms for resource scheduling and allocation in large distributed systems.
- Application-specific approaches for fault-tolerance and energy management, with a focus on workflow-based applications

Although the presentations at the seminar were very diverse in scope, ranging from practice to theory, an interesting observation is that many works do establish strong links between practice (e.g., particular applications, programming models) and theory (e.g., abstract scheduling problems and results). In particular, it was found that workflow applications, far from being well-understood, in fact give rise to a range of interrelated and interesting practical and theoretical problems that must be solved conjointly to achieve efficiency at large scale. Estimating task weights, scheduling with uncertainties, mapping at scale, remapping after failures, trading performance and energy, these are a few challenges that have been discussed at length during the seminar. Such observations make it plain that forums that blends practice and theory, as is the case with this seminar, are very much needed.

The seminar brought together 41 researchers from Austria, France, Germany, Japan, Netherlands, New Zealand, Poland, Portugal, Spain, Sweden, Switzerland, UK and USA, with interests and expertise in different aspect of parallel and distributed computing. Among participants there was a good mix of senior researchers, junior researchers, postdoctoral researchers, and Ph.D. students. Altogether there were 29 presentations over the 5 days of the seminar, organized in morning and late-afternoon sessions. The program was as usual a compromise between allowing sufficient time for participants to present their work, while also providing unstructured periods that were used by participants to pursue ongoing collaborations as well as to foster new ones. The feedback provided by the participants show that the goals of the seminar, namely to circulate new ideas and create new collaborations, were met to a large extent.

The organizers and participants wish to thank the staff and the management of Schloss Dagstuhl for their assistance and support in the arrangement of a very successful and productive event.

2 Table of Contents

3 Overview of Talks

3.1 Heterogeneous supercomputing systems power-aware scheduling and resiliency

Sadaf Alam (CSCS – Swiss National Supercomputing Centre, CH)

Swiss National Supercomputing Centre (CSCS) operates a highly energy efficient, Petscale hybrid Cray XC30 system, which has been in operation since the beginning of 2014. The system has a wide variety of user level and system logging power management and measurement tools that allow for fine grain power measurements of applications. However, the current scheduling and resource management scheme does not take into consideration the power requirements of applications. There is a potential for optimizing operating cost by carefully scheduling jobs based on an additional resource management parameter in addition to the node requirements and wall clock time. An analysis of system and job submission logs could provide insight into power aware resource management of the system. Likewise, resiliency on heterogeneous nodes have additional dimensions due to different types of execution units, memory as well as I/O. Correlating different error codes with failures allow us to detect, isolate and resolve issues with minimum impact to operations.

3.2 Analyzing real cluster data for formulating robust allocation algorithms in cloud platforms

Olivier Beaumont (INRIA – Bordeaux, FR)

Joint work of Beaumont, Olivier; Eyraud-Dubois, Lionel; Lorenzo-del-Castillo, Juan-Angel

A problem commonly faced in Computer Science research is the lack of real usage data that can be used for the validation of algorithms. This situation is particularly true and crucial in Cloud Computing. The privacy of data managed by commercial Cloud infrastructures, together with their massive scale, makes them very uncommon to be available to the research community. Due to their scale, when designing resource allocation algorithms for Cloud infrastructures, many assumptions must be made in order to make the problem tractable.

This talk provides an analysis of a cluster data trace recently released by Google and focuses on a number of questions which have not been addressed in previous studies. In particular, we describe the characteristics of job resource usage in terms of dynamics (how it varies with time), of correlation between jobs (identify daily and/or weekly patterns), and correlation inside jobs between the different resources (dependence of memory usage on CPU usage) and at failures (are they independent or correlated). From this analysis, we propose a way to formalize the allocation problem on such platforms, which encompasses most job features from the trace with a small set of parameters.

3.3 Which verification for soft error detection?

Anne Benoit (ENS Lyon, FR)

Joint work of Bautista-Gomez Leonardo; Benoit, Anne; Cavelan, Aurélien; Raina, Saurabh K.; Robert, Yves; Sun, Hongyang

Many methods are available to detect silent errors in high-performance computing (HPC) applications. Each comes with a given cost and recall (fraction of all errors that are actually detected). The main contribution of this paper is to show which detector(s) to use, and to characterize the optimal computational pattern for the application: how many detectors of each type to use, together with the length of the work segment that precedes each of them. We conduct a comprehensive complexity analysis of this optimization problem, showing NP-completeness and designing an FPTAS (Fully Polynomial-Time Approximation Scheme). On the practical side, we provide a greedy algorithm whose performance is shown to be close to the optimal for a realistic set of evaluation scenarios.

3.4 Practical foundations for resilient applications

George Bosilca (University of Tennessee, US)

Despite the fact that faults are accepted as normal occurrences, the existing programming paradigms are lacking methodologies and tools to deal with faults in a holistic manner. This talk covers an extension to the MPI paradigm allowing applications or libraries to provide transparent user level fault management solutions. Based on such solutions we describe several mixed solution, encompassing both traditional checkpoint/restart and application-specific approaches, to minimize the resilience overhead. Additionally, we introduce theoretical models to evaluate such mixed resilience models executed on particular hardware environments in order to assess the benefits and overhead of resilient applications at exascale.

3.5 Task resource consumption prediction for scientific applications and workflows

Rafael Ferreira da Silva (USC – Marina del Rey, US)

Estimates of task runtime, disk space usage, and memory consumption, are commonly used by scheduling and resource provisioning algorithms to support efficient and reliable scientific application executions. Such algorithms often assume that accurate estimates are available, but such estimates are difficult to generate in practice. In this work, we first profile real scientific applications and workflows, collecting fine-grained information such as process I/O, runtime, memory usage, and CPU utilization. We then propose a method to automatically characterize task requirements based on these profiles. Our method estimates task runtime, disk space, and peak memory consumption. It looks for correlations between the parameters of a dataset, and if no correlation is found, the dataset is divided into smaller subsets using the statistical recursive partitioning method and conditional inference trees to identify patterns that characterize particular behaviors of the workload. We then propose an estimation process to predict task characteristics of scientific applications based on the collected data. For scientific workflows, we propose an online estimation process based on the MAPE-K loop, where task executions are monitored and estimates are updated as more information becomes available. Experimental results show that our online estimation process results in much more accurate predictions than an offline approach, where all task requirements are estimated prior to workflow execution.

3.6 Cost constrained, static and dynamic workflow scheduling on IaaS clouds

Michael Gerhards (FH Aachen – Jülich, DE)

This talk presents a novel cloud-aware workflow scheduling approach. Scientific computing infrastructures of the last years were dominated by grids forming resource pools provided by independent organizations. These resources were provided and accessed following a set of rules that finally are the foundation of the virtual organization (VO). The cloud computing paradigm offers alternative computing infrastructures such as Amazon EC2 and Microsoft Azure. While both paradigms focus on characteristics such as broad network access, resource pooling, and measured service, cloud computing additionally prioritizes on-demand self-service and rapid elasticity mostly offered by a pay-as-you-go business model. With respect to scheduling, these cloud specific characteristics introduces new opportunities and challenges.

Large-scale scientific workflow applications composed of inter-dependent computational tasks are an important group of applications. Since the scheduling of these workflows is NP-complete, practical solutions will have to sacrifice optimality for the sake of efficiency. The literature distinguishes between two types of scheduling mechanisms, namely static scheduling and dynamic scheduling. In the static scheduling mechanism, all decisions are made offline before starting the execution of an application. Here, the question arises, how the knowledge needed to perform static scheduling is acquired. However, the workflow's runtime behaviors might differ from offline-planning. Hence, the static resource allocation plan might become uneconomic or might even fail in meeting a deadline. In the contrary, in the dynamic scheduling mechanism, all decisions are made at runtime under consideration of the system feedback. Hence, dynamic scheduling provides a more adaptive solution. However, the commonly implemented Master-Slave model assumes the existence of slave resources and does therefore neglect the problem of selecting a cost optimized set of resources in the cloud.

The outlined problems of both scheduling mechanisms motivate the main research hypothesis of this thesis: "How to combine the benefits of cloud-aware static and dynamic workflow scheduling mechanisms consistently in one solution?" More precisely, the following central research objectives are addressed:

- Identification of cost reduction strategies for workflow scheduling on pay-as-you-go clouds
- Estimation of task runtime profiles for heterogeneous resources based on provenance data
- Deriving of a flow pattern based polynomial complex static workflow scheduling algorithm
- Deriving of a resource allocation plan based low complex dynamic workflow scheduling algorithm

3.7 Resilient multigrid solvers

Dominik Goeddeke (TU Dortmund, DE)

Joint work of Altenbernd, Mirco; Goeddeke, Dominik; Ribbrock, Dick

We discuss fault tolerance schemes for data loss in multigrid solvers, that essentially combine ideas of checkpoint-restart with algorithm-based fault tolerance. We first demonstrate and prove that multigrid is self-stabilising with respect to single faults. With increasing fault rates however, checkpoints are required. To improve efficiency compared to conventional checkpointing, we exploit the inherent data compression of the multigrid hierarchy, and relax the synchronicity requirement through a local failure local recovery approach. We experimentally identify the root cause of convergence degradation in the presence of data loss using smoothness considerations. While these techniques primarily aim at the case of lost nodes, towards the end of the talk we also discuss a novel black-box approach based on the Full Approximation Scheme (FAS) that encapsulates all bitflips in the smoother and protects the traversal through the hierarchy with checksums.

3.8 Bridging the gap between performance and bounds of cholesky factorization on heterogeneous platforms

Julien Herrmann (ENS Lyon, FR)

Joint work of Agullo, Emmanuel; Beaumont, Olivier; Eyraud-Dubois, Lionel; Herrmann, Julien; Kumar, Suraj; Marchal, Loris; Thibault, Samuel

We consider the problem of allocating and scheduling dense linear application on fully heterogeneous platforms made of CPUs and GPUs. More specifically, we focus on the Cholesky factorization since it exhibits the main features of such problems. Indeed, the relative performance of CPU and GPU highly depends on the sub-routine: GPUs are for instance much more efficient to process regular kernels such as matrix-matrix multiplications rather than more irregular kernels such as matrix factorization. In this context, one solution consists in relying on dynamic scheduling and resource allocation mechanisms such as the ones provided by PaRSEC or StarPU. In this paper we analyze the performance of dynamic schedulers based on both actual executions and simulations, and we investigate how adding static rules based on an offline analysis of the problem to their decision process can indeed improve their performance, up to reaching some improved theoretical performance bounds which we introduce.

3.9 Accurately measuring MPI collectives with synchronized clocks

Sascha Hunold (TU Wien, AT)

We consider the problem of accurately measuring the time to complete an MPI collective operation, as the result strongly depends on how the time is measured. Our goal is to develop an experimental method that allows for reproducible measurements of MPI collectives. When executing large parallel codes, MPI processes are often skewed in time when entering a collective operation. However, for the sake of reproducibility, it is a common approach to synchronize all processes before they call the MPI collective operation. We therefore take a closer look at two commonly used process synchronization schemes: (1) relying on $MPI_Barrier$ or (2) applying a window-based scheme using a common global time. We analyze both schemes experimentally and show the pros and cons of each approach. As window-based schemes require the notion of global time, we thoroughly evaluate different clock synchronization algorithms in various experiments. We also propose a novel clock synchronization algorithm that combines two advantages of known algorithms, which are (1) taking the inherent clock drift into account and (2) using a tree-based synchronization scheme to reduce the synchronization duration.

3.10 Energy-aware scheduling for task-based applications in distributed platforms

Fredy Juarez (Barcelona Supercomputing Center, ES)

Joint work of Badia, Rosa; Ejarque, Jorge; Juarez, Fredy

Green Computing is a recent trend in computer science which tries to reduce the energy consumption and carbon footprint produced by computers. One of the methods to reduce this consumption is providing scheduling policies for taking into account not only the processing time but also the energy consumption. We propose a real-time dynamic scheduling system to efficiently execute task-based applications on Distributed Computing platforms such as Cluster, Grids and Clouds. The proposed scheduler minimizes a multi-objective function which combines the energy-consumption and execution time according to the energy-performance importance factor provided by the user or cloud provider. Due to the time limitation required for run-time schedulers, the implementation of large-time optimization algorithms are not suitable. Therefore, we combine a set of heuristic rules plus the resource allocator algorithm to get good real-time scheduling solutions in an affordable time scale.

3.11 Platforms to fit and platforms to experiment

Katarzyna Keahey (Argonne National Laboratory, US)

The data processing needs of applications are undergoing a significant change, increasingly emphasizing on-demand availability and real-time or simply predictable response time, driven by availability of dynamic data streams from experimental instruments, social networks, and specialized sensing devices; a new disruptive factor promising to be the primary driver of discovery in environmental and other sciences over the next decade. Applications of this type radically change what is required of scientific resources. Where before a resource would provide a static offering and require an application to adapt itself to hardware and configuration trade-offs offered by the resource, now the requirement for on-demand availability is increasingly requiring a resource to be able to adapt itself the requirements of the application in time to service time-sensitive requests. In other words, resource providers are asked to offer a programmable platform, that can adapt itself to environment and hardware requirements of an application.

This shifts emphasis of many questions we used to ask about software. Where before adapting the application was the main focus of optimization, now time is spent in developing technologies that adapt the platform. Where before available platforms could be explored manually and at a coarse grain, now automated discovery and fine-grained understanding of the platform's capabilities become of paramount importance. And finally, once a platform becomes dynamic its ability to adapt may be used during a run's lifecycle as well as before providing dynamic support for malleable applications. A factor essential to achieving all this is application's ability to comprehensively define its needs.

This talk describes two types of technologies essentially to create a programmable platform: "platform to fit" technologies, that adapt resources to the needs of the application, and "fitting datacenter" technologies that ensure that the resource is presented such that it forms good building blocks for such adaptability. The critical element of a "fitting datacenter" is a resource manager that provides resource leases to users and applications that offer a significant degree of programmability both in terms of environment configuration and the shape of a resource slot. A key element of such resource manager are containers, implemented via a variety of technologies such as virtual machines (e.g., KVM or Xen) or technologies based on OS-based constructs such as Docker or LXE. To accommodate both on-demand availability and high utilization such resource manager has to find ways to efficiently interleave on-demand leases with the demands of batch and high throughput computing codes. The "platform to fit" leverages on-demand availably of a variety of resources to provide adaptability in the platform. Resource adaptability can happen in the compute dimension, where integrating additional resources by deploying on-demand VMs can keep response times stable when dealing with peak traffic from dynamic data streams and can be regulated by a variety of policies. Similarly available storage capacity can be adapted to the needs of an application or a workload, but dynamically adding and releasing storage volumes given efficient filesystem supporting such operation. And finally the available I/O bandwidth can also be adapted by placing a caching system in front of storage offerings with different I/O bandwidths at different price points; this means that data could be managed transparently between fast disks during computation or periods of intensive access, and archival storage when periodically not used.

Experimental testbeds for computer science are critical for experimenting with these types of technologies. The new NSF-funded Chameleon testbed (www.chameleoncloud.org) has been built specifically to provide such experimental capabilities. It's hardware makup, consisting of 15,000 cores distributed over two sites connected by a 100Gbps network, and concentrated primarily in one homogenous partition, as well as 5 PB of storage provide a good canvas for Big Data and Big Compute experiments. Chameleon offers bare hardware reconfiguration allowing users to recreate as nearly as possible conditions available in their own lab and offers reproducibility. The testbed is currently in the Early User stage and will be publicly available by fall of 2015.

3.12 Some thoughts about extreme scaling and power

Dieter Kranzlmüller (LMU München, DE)

This talk discusses the (scheduling) problems on the supercomputer SuperMUC Phase 1 and 2, operating at the Leibniz Supercomputing Centre (LRZ). The issues include scalability, power, scheduling, and infrastructure issues.

3.13 Communication lower bounds for matrix-matrix multiplication

Julien Langou (University of Colorado Denver, US)

We consider communication lower bounds for matrix-matrix multiplication in the sequential case. Our new proof technique improves the known lower bound for the number of reads and writes in the sequential memory case.

3.14 Workflow scheduling and optimization on clouds

Maciej Malawski (AGH University of Science & Technology, Krakow, PL)

The area of my research includes the problem of workflow scheduling on IaaS clouds.

The first part of the talk focuses on data-intensive workflows, and addresses the problem of scheduling workflow ensembles under cost and deadline constraints in clouds. We developed a simulation model for handling file transfers between tasks, featuring the ability to dynamically calculate bandwidth and supporting a configurable number of replicas, thus allowing us to simulate various levels of congestion. The resulting model is capable of representing a wide range of storage systems available on clouds: from in-memory caches (such as memcached), to distributed file systems (such as NFS servers) and cloud storage (such as Amazon S3). Next, we propose and evaluate a novel scheduling algorithm that minimizes the number of transfers by taking advantage of data caching and file locality.

The second part of the talk covers other aspects of cloud workflow scheduling. They include problems with inaccurate run time estimates and task granularity. When addressing the problem of scheduling on multiple clouds, we developed optimization methods based on mixed integer programming. We have also performed experiments with performance evaluation of applications on clouds and developed a performance model of workflow for ISMOP flood prevention project.

3.15 Scheduling trees of malleable tasks for sparse linear algebra

Loris Marchal (ENS Lyon, FR)

Joint work of Guermouche, Abdou; Marchal, Loris; Simon, Bertrand; Vivien, Frédéric

Scientific workloads are often described by directed acyclic task graphs. This is in particular the case for multifrontal factorization of sparse matrices – the focus of this talk – whose task graph is structured as a tree of parallel tasks. Prasanna and Musicus advocated using the concept of malleable tasks to model parallel tasks involved in matrix computations. In this powerful model each task is processed on a time-varying number of processors. Following Prasanna and Musicus, we consider malleable tasks whose speedup is p^α, where p is the fractional share of processors on which a task executes, and α $(0 < \alpha <= 1)$ is a task-independent parameter. Firstly, we use actual experiments on multicore platforms to motivate the relevance of this model for our application. Then, we study the optimal time-minimizing allocation proposed by Prasanna and Musicus using optimal control theory. We greatly simplify their proofs by resorting only to pure scheduling arguments. Building on the insight gained thanks to these new proofs, we extend the study to distributed (homogeneous or heterogeneous) multicore platforms. We prove the NP-completeness of the corresponding scheduling problem, and we then propose some approximation algorithms.

3.16 The inevitable end of Moore's law beyond Exascale will result in data and HPC convergence

Satoshi Matsuoka (Tokyo Institute of Technology, JP)

The so-called "Moore's Law", by which the performance of the processors will increase exponentially by factor of 4 every 3 years or so, is slated to be ending in 10–15 year timeframe due to the lithography of VLSIs reaching its limits around that time, and combined with other physical factors. This is largely due to the transistor power becoming largely constant, and as a result, means to sustain continuous performance increase must be sought otherwise than increasing the clock rate or the number of floating point units in the chips, i.e., increase in the FLOPS. The promising new parameter in place of the transistor count is the perceived increase in the capacity and bandwidth of storage, driven by device, architectural, as well as packaging innovations: DRAM-alternative Non-Volatile Memory (NVM) devices, 3-D memory and logic stacking evolving from VIAs to direct silicone stacking, as well as next-generation terabit optics and networks. The overall effect of this is that, the trend to increase the computational intensity as advocated today will no longer result in performance increase, but rather, exploiting the memory and bandwidth capacities will instead be the right methodology. However, such shift in compute-vs-data tradeoffs would not exactly be return to the old vector days, since other physical factors such as latency will not change. As such, performance modeling to account for the evolution of such fundamental architectural change in the post-Moore era would become important, as it could lead to disruptive alterations on how the computing system, both hardware and software, would be evolving towards the future.

3.17 Lazy shadowing: a power-aware resilience scheme for extreme scale systems

Rami Melhem (University of Pittsburgh, US)

As the demand for high performance computing (HPC) and cloud computing accelerates, the underlying infrastructure is expected to ensure reliability while maintaining high performance and cost-effectiveness, even with multifold increase in the number of computing, storage and communication components. Current resilience approaches rely upon either time redundancy (re-execution after failure) or hardware redundancy (executing multiple copies). The first approach is subject to a significant delay while the second approach requires additional hardware and increases the energy consumption for a given service. In general, the trade-off between performance, fault-tolerance and power consumption calls for new frameworks which is energy and performance aware when dealing with failures. In this talk, Shadow Computing is introduced to address the above trade-off challenge by associating with each main process a shadow which executes at a reduced rate through either voltage/frequency scaling or by co-locating multiple shadows on the same processor. Adjusting the rate of execution enables a parameterized trade-off between response time, energy consumption and hardware redundancy.

3.18 Modelling Matlab/Simulink periodic applications using Synchronous DataFlow Graphs

Alix Munier (UPMC – Paris, FR)

Matlab/Simulink is a simulation tool widely used in an industrial context like automotive to design embedded electronic applications.

The aim of this talk is to prove that communications between Simulink periodic blocks can be described using a Synchronous Dataflow Graph (SDF in short). Therefore, the amounts of data exchanged between the blocks are modeled by closed formulas and are completely characterized without any simulation phase. The major consequence is that applications designed using Simulink can take advantage of the (semi)-automatic tools for optimized execution of a SDF on a many-core architecture.

3.19 HAEC – Highly Adaptive Energy-Efficient Computing

Wolfgang E. Nagel (TU Dresden, DE)

The visionary goal of the Collaborative Research Center HAEC (highly adaptive energy-efficient computing) is to research technologies to enable computing systems with high energy-efficiency without compromising on high performance. HAEC will concentrate on researching larger server systems, from applications to hardware. To achieve the goal of an integrated approach of highly adaptive energy-efficient computing (HAEC), the problem is approached at all levels of technology involved, the hardware, the computer architecture, and operating system, the software modeling as well as the application modeling and runtime control levels. A novel concept, namely the HAEC Box, of how computers can be built by utilizing innovative ideas of optical and wireless chip-to-chip communication is explored. This would allow a new level of run-time adaptivity of future computers, creating a platform for flexibly adapting to the needs of the computing problem. The talk will describe the general approach and the implications on scheduling challenges in future systems.

3.20 Application-aware approaches to resilience at extreme scales

Manish Parashar (Rutgers University, US)

Application resilience is a key challenge as we target exascale, and process/node failures represent an important class of failures that must be addressed. However, typical approaches for handling these failures, such as those based on terminating jobs and restarting them from the last stored checkpoints are not expected to scale to exascale. In this talk I will explore application driven approaches that use knowledge of the application and/or the methods used to reduce overheads due to fault tolerance for MPI-based parallel applications, and increase their resilience. Specifically, I will present approaches for online global and local recovery, implicitly coordinated checkpointing, and failure masking. I will also describe Fenix, a scalable framework for enabling online recovery from process/node/blade/cabinet failures. Finally, I will present evaluations of the developed approaches and of Fenix using the S3D combustion simulation running on the Titan Cray-XK7 production system at ORNL.

3.21 Cost-efficient resource management for scientific workflows on the cloud

Ilia Pietri (University of Manchester, UK)

Scientific workflows, which describe complex computational problems in many scientific fields, may be executed on high performance systems such as Clouds. Cloud providers now offer to users CPU provisioning as different combinations of CPU frequencies and prices. With the increasing number of options, the problem of choosing cost-efficient configurations is becoming more challenging. This talk discusses approaches to minimize the energy cost from the provider's perspective and the user monetary cost while achieving good performance in terms of execution time. It presents a performance model to estimate the workflow makespan under different configurations and frequency selection approaches to determine the CPU frequencies to operate the available resources and execute the workflow tasks, based on both the application and system characteristics.

3.22 A decade of modelling parallel computing systems

Sabri Pllana (Linnaeus University, Växjö, SE)

Joint work of Suraj Prabhakaran, Marcel Neumann, Felix Wolf

In mature engineering disciplines such as civil engineering, before an artefact (for instance a bridge) is built, the corresponding model is developed. We argue that a practice "model first then build the code" would benefit also software engineers. Since programming parallel systems is considered significantly more complex than programming sequential systems, the use of models in the context of parallel computing systems is of particular importance. In this talk we will highlight some of our models of parallel computing systems that we have developed in the last decade. We will first address various modelling aspects in the context of clusters of SMPs, continue thereafter with the Grid, and conclude this talk with heterogeneous computing systems.

3.23 Dynamic node replacement and adaptive scheduling for fault tolerance

Suraj Prabhakaran (TU Darmstadt, DE)

Joint work of Neumann, Marcel; Prabhakaran, Suraj; Wolf, Felix

Batch systems traditionally support only static resource management wherein a job's resource set is unchanged throughout execution. Node failures force the batch systems to restart affected jobs on a fresh allocation (typically from a checkpoint) or replace failed nodes with statically allocated spare nodes. As future systems are expected to have high failure rates, this solution leads to increased job restart overhead, additional long waiting times before job restart and excessive resource wastage. In this talk, we present an extension of the TORQUE/Maui batch system with dynamic resource management facilities that enable instant replacement of failed nodes to affected jobs without requiring a job restart. The proposed batch system supports all job types for scheduling – rigid, moldable, evolving and malleable. We present an algorithm for the combined scheduling of all job types and show how the unique features of various jobs and the scheduling algorithm can expedite node replacements. The overall expected benefit of this approach is a highly resilient cluster environment that ensures timely completion of jobs and maintains high throughput even under frequent node failures.

3.24 Hybrid clouds: beyond classical buy-or-lease decisions

Stephan Schlagkamp (TU Dortmund, DE)

Joint work of Schlagkamp, Stephan; Schwiegelshohn, Uwe; Tchernykh, Andrei

An increasing number of companies favor hybrid clouds since this approach does not require vital company data to leave the premises while it still allows efficient handling of demand peaks. Those companies must determine the size of the private cloud component by considering cost and demand forecasts. We formulate an optimization problem addressing cost optimality of a hybrid cloud at strategic level. Furthermore, we present a simple method to calculate a cost optimal size of the private cloud given a statistical demand prediction. Such demand prediction respects unavoidable deviations in prediction. Moreover, we analyze robustness of our method against errors in cost and demand estimations.

3.25 Duplicate-free state-space model for optimal task scheduling

Oliver Sinnen (University of Auckland, NZ)

Joint work of Orr, Michael; Sinnen, Oliver

The problem of task scheduling with communication delays ($P|prec, c_{ij}|C_{max}$) is NP-hard, and therefore solutions are often found using a heuristic method. However, an optimal schedule can be very useful in applications such as time critical systems, or as a baseline for the evaluation of heuristics. Branch-and-bound algorithms such as A* have previously been shown to be a promising approach to the optimal solving of this problem, using a state-space model which we refer to as exhaustive list scheduling. However, this model suffers from the possibility of producing high numbers of duplicate states. In this paper we define a new state-space model in which we divide the problem into two distinct subproblems: first we decide the allocations of all tasks to processors, and then we order the tasks on their allocated processors in order to produce a complete schedule. This two-phase state-space model offers no potential for the production of duplicates. An empirical evaluation shows that the use of this new state-space model leads to a marked reduction in the number of states considered by an A* search in many cases, particularly for task graphs with a high communication-to-computation ratio. With additional refinement, and the development of specialised pruning techniques, the performance of this state-space model could be further improved.

3.26 Energy-aware task management for heterogeneous low-power systems

Leonel Sousa (Technical University of Lisboa, PT)

This talk presents a framework for energy-aware task management in heterogeneous embedded platforms, which integrates a set of novel application-aware management mechanisms for efficient resource utilization, frequency scaling and task migration. These mechanisms rely on a new management concept, which promotes performance fairness among running tasks to attain energy savings, while respecting the target performance of parallel applications. The proposed framework integrates several components for accurate run-time monitoring and application self-reporting. Experimental results show that energy savings of up to 39% were achieved in a state-of-the-art embedded platform for a set of real-world SPEC CPU2006 and PARSEC benchmarks.

3.27 From static scheduling towards understanding uncertainty

Andrei Tchernykhk (CICESE Research Center, US)

Clouds differ from previous computing environments in the way that they introduce a continuous uncertainty into the computational process. The uncertainty brings additional challenges to both end-users and resource providers. It requires waiving habitual computing paradigms, adapting current computing models to this evolution, and designing novel resource management strategies to mitigate uncertainty and handle it in an effective way. In this talk, we address scheduling algorithms for different scenarios of HPC, Grid and Cloud Infrastructures. We provide some theoretical and experimental bounds for static, dynamic and adaptive approaches. We discuss the role of uncertainty in the resource/service provisioning, investment, operational cost, programming models. We discuss several major sources of uncertainty: dynamic elasticity, dynamic performance changing, virtualization, loosely coupling application to the infrastructure, among many others.

3.28 Quantifying resiliency in the extreme scale hpc co-design space

Jeffrey S. Vetter (Oak Ridge National Laboratory, US)

Joint work of Meredith, Jeremy; Vetter, Jeffrey S.

A key capability for the co-design of extreme-scale HPC systems is the accurate modeling of performance, power, and resiliency. We have developed the Aspen performance modeling language that allows fast exploration of the holistic design space. Aspen is a domain specific language for structured analytical modeling of applications and architectures. Aspen specifies a formal grammar to describe an abstract machine model and describe an application's behaviors, including available parallelism, operation counts, data structures, and control flow. Aspen is designed to enable rapid exploration of new algorithm and architectures. Recently, we have used Aspen to model application resiliency based on data vulnerability. To facilitate this vulnerability classification, it is important to have accurate, quantitative techniques that can be applied uniformly and automatically across real-world applications. Traditional methods cannot effectively quantify vulnerability, because they lack a holistic view to examine system resilience, and come with prohibitive evaluation costs. To attack this problem, we introduce a data-driven, practical methodology to analyze these application vulnerabilities using a novel resilience metric: the data vulnerability factor (DVF). DVF integrates knowledge from both the application and target hardware into the calculation. To calculate DVF, we extend the Aspen performance modeling language to provide a structured, fast modeling solution.

3.29 Voltage overscaling algorithms for energy-efficient computations with timing errors

Frédéric Vivien (ENS Lyon, FR)

Joint work of Cavelan, Aurélien; Robert, Yves; Sun, Hongyang; Vivien, Frédéric

In this work, we discuss several scheduling algorithms to execute tasks with voltage overscaling. Given a frequency to execute the tasks, operating at a voltage below threshold leads to significant energy savings but also induces timing errors. A verification mechanism must be enforced to detect these errors. Contrarily to fail-stop or silent errors, timing errors are deterministic (but unpredictable). For each task, the general strategy is to select a voltage for execution, to check the result, and to select a higher voltage for re-execution if a timing error has occurred, and so on until a correct result is obtained. Switching from one voltage to another incurs a given cost, so it might be efficient to try and execute several tasks at the current voltage before switching to another one. Determining the optimal solution turns out to be unexpectedly difficult. However, we provide the optimal algorithm for a single task and for a chain of tasks, the optimal algorithm when there are only two voltages, and the optimal level algorithm for several tasks, where a level algorithm is defined as an algorithm that executes all remaining tasks when switching to a given voltage. Furthermore, we show that the optimal level algorithm is in fact globally optimal (among all possible algorithms) when voltage switching costs are linear. Finally, we report a set of simulations to assess the potential gain of voltage overscaling algorithms.

Participants

- Sadaf Alam
CSCS – Lugano, CH
- Olivier Beaumont
INRIA – Bordeaux, FR
- Anne Benoit
ENS – Lyon, FR
- George Bosilca
University of Tennessee, Knoxville, US
- Henri Casanova
University of Hawaii at Manoa – Honolulu, US
- Ewa Deelman
USC – Marina del Rey, US
- Rafael Ferreira da Silva
USC – Marina del Rey, US
- Carsten Franke
ABB Corporate Research – Baden-Dättwil, CH
- Michael Gerhards
FH Aachen – Jülich, DE
- Dominik Göddeke
Universität Stuttgart, DE
- Julien Herrmann
ENS – Lyon, FR
- Sascha Hunold
TU Wien, AT
- Fredy Juarez
Barcelona Supercomputing Center, ES
- Kate Keahey
Argonne National Laboratory, US
- Thilo Kielmann
VU University Amsterdam, NL
- Dieter Kranzlmüller
LMU München, DE
- Julien Langou
Univ. of Colorado – Denver, US
- Maciej Malawski
AGH University of Science & Technology – Krakow, PL
- Loris Marchal
ENS – Lyon, FR
- Satoshi Matsuoka
Tokyo Institute of Technology, JP
- Rami Melhem
University of Pittsburgh, US
- Bernd Mohr
Jülich Supercomputing Centre, DE
- Alix Munier
UPMC – Paris, FR
- Jaroslaw Nabrzyski
Univ. of Notre Dame, US
- Wolfgang E. Nagel
TU Dresden, DE
- vManish Parashar
Rutgers Univ. – Piscataway, US
- Ilia Pietri
University of Manchester, GB
- Sabri Pllana
Linnaeus University – Växjö, SE
- Suraj Prabhakaran
TU Darmstadt, DE
- Padma Raghavan
Pennsylvania State University – University Park, US
- Yves Robert
ENS – Lyon, FR
- Stephan Schlagkamp
TU Dortmund, DE
- Uwe Schwiegelshohn
TU Dortmund, DE
- Oliver Sinnen
University of Auckland, NZ
- Renata Slota
AGH University of Science & Technology – Krakow, PL
- Veronika Sonigo
University of Franche-Comté – Besançon, FR
- Leonel Sousa
Technical Univ. of Lisboa, PT
- Andrei Tchernykh
CICESE Research Center, US
- Jeffrey S. Vetter
Oak Ridge National Lab., US
- Frédéric Vivien
ENS – Lyon, FR
- Felix Wolf
TU Darmstadt, DE