EURO-PAR
CONFERENCE 2021

KEY DATES

Abstracts submission: 8 February 2021 AoE *(extended and final)*

Paper submission: 19 February 2021 AoE *(extended and final)*

Workshop proposals: 12 February 2021

Author notification: 30 April 2021

PAPER SUBMISSION GUIDELINES

- Papers must be in PDF format and should not exceed 14 pages (including references) and 7500 words.
- Papers must be formatted in the Springer LNCS style: www.springer.com/gp/computer-science/lncs/conference-proceedings-guidelines
- Papers that don't meet these requirements might be rejected without a review.
- Contributions submitted elsewhere or currently under review will not be considered.
- All submitted papers will be checked for originality by Springer iThenticate. Papers which show an insufficient originality might be rejected without a review.
- Paper submissions are made through EasyChair, using the link below.
- **Double-blind is optional and not required.**

PAPER SUBMISSION

Topics

TOPIC 1: COMPILERS, TOOLS AND ENVIRONMENTS

Global Chair: Frank Hannig, Friedrich-Alexander-Universität Erlangen-Nürnberg, Germany
Local Chair: Gabriel Falcao, Instituto de Telecomunicações, Universidade de Coimbra, Portugal

This topic addresses programming tools and system software for all kinds of parallel computer architectures, ranging from low-power embedded high-performance systems, multi- and many-core processors, accelerators to large-scale computers and cloud computing. Focus areas include compilation and software testing to design effectively well-defined components and verify their necessary structural, behavioral, and parallel interaction properties. It deals with tools, analysis software, and runtime environments to address the challenges of programming and executing the parallel architectures mentioned above. Moreover, the topic deals with methods and tools for optimizing non-functional properties such as performance, programming productivity, robustness, energy efficiency, and scalability.

Focus areas (not limited to):

- Compilation for parallel processor architectures (multi-cores, vector units, heterogeneous processors, etc.) and accelerators (e.g., GPUs, reconfigurable hardware)
- Compilation for parallel embedded systems, low-power edge and mobile devices, MPSoCs, in-memory computing
- Iterative, just-in-time, feedback-oriented, dynamic, and machine-learning-based compilation
- Optimizing target-architecture-agnostic compilers and translators (e.g., source-to-source translation) for domain-specific languages
- Autotuning techniques and tools
- Program development tools and programming environments
- Debugging and correctness tools
- Runtime support systems, tools, libraries, and environments for message passing and distributed computing, such as clusters, clouds, and grids
- Optimization of non-functional properties (e.g., performance, programming productivity, robustness, reliability, power, and energy)
- Software for testing, instrumentation, monitoring, analysis, and visualization of as well functional as non-functional properties
- Interaction between application, compiler, runtime system, and operating system
- Interoperable tool environments, integration of tools, compilers, and runtime systems
- Tool infrastructure and scalability, tools for extreme-scale systems
- Tool evaluations and comparisons in production environments
- Success stories of compilers, programming tools, libraries, and environments

TOPIC 2: PERFORMANCE AND POWER MODELING, PREDICTION AND EVALUATION

Global Chair: Didem Unat, Koç University, Turkey
Local Chair: Aleksandar Ilic, INESC-ID, Universidade de Lisboa, Portugal

In recent years, a range of novel methods and tools have been developed for the evaluation, design, and modeling of parallel and distributed systems and applications. At the same time, the term 'performance' has broadened to also include scalability and energy efficiency, and touching reliability and robustness in addition to the classic resource-oriented notions. The aim of this topic is to gather researchers working on different aspects of performance modeling, evaluation, and prediction, be it for systems or for applications running on the whole range of parallel and distributed systems (multi-core and heterogeneous architectures, HPC systems, grid and cloud contexts etc.). Authors are invited to submit novel research in all areas of performance modeling, prediction and evaluation, and to help bring together current theory and practice.

Focus areas (not limited to):

- Design of experiments, reproducible experiments
- Novel techniques and tools for performance measurement, evaluation, and prediction
- Advanced simulation techniques and tools
- Measurements, benchmarking, and tracing
- Workload modeling
- Performance-driven code optimization
- Verification and validation of performance models
- Performance visualization
- Power consumption modeling and prediction
- Performance modeling and simulation of emerging exascale systems

TOPIC 3: SCHEDULING AND LOAD BALANCING

Global Chair: Oliver Sinnen, University of Auckland, New Zealand
Local Chair: Jorge Barbosa, Universidade do Porto, Portugal

New computing systems offer the opportunity to reduce the response times and the energy consumption of the applications by exploiting the levels of parallelism. Heterogeneity and complexity are the main characteristics of modern architectures, being the optimal exploitation of such platforms challenging. Scheduling and load balancing techniques are key instruments to achieve higher performance, lower energy consumption, reduced resource usage, and real-time properties of applications.

This topic invites papers on all aspects related to scheduling and load balancing on parallel and distributed machines, from theoretical foundations for modelling and designing efficient and robust scheduling policies to experimental studies, applications and practical tools and solutions. It applies to multi-/manycore processors, embedded systems, servers, heterogeneous and accelerated systems, HPC clusters as well as distributed systems such as clouds and global computing platforms.

Focus areas (not limited to):

- Scheduling algorithms for homogeneous and heterogeneous platforms
- Theoretical foundations of scheduling algorithms
- Real-time scheduling on parallel and distributed machines
- Robustness of scheduling algorithms
- Feedback-based load balancing
- Multi-objective scheduling
- Resilient scheduling
- Scheduling, coordination and overhead at extreme scales
- On-line scheduling
- Energy and temperature awareness in scheduling and load balancing
- Workload characterization and modelling
- Workflow scheduling
- Performance models for scheduling and load balancing
- Reproducibility of scheduling

TOPIC 4: DATA MANAGEMENT, ANALYTICS AND MACHINE LEARNING

Global Chair: Alex Delis, University of Athens, Greece
Local Chair: Helena Aidos, LASIGE, Universidade de Lisboa, Portugal

Many areas of science, industry, and commerce are producing extreme-scale data that must be processed — stored, managed, analyzed — in order to extract useful knowledge. This topic seeks papers in all aspects of distributed and parallel data management and data analysis. For example, cloud and grid data-intensive processing, parallel and distributed machine learning, HPC in situ data analytics, parallel storage systems, scalable data processing workflows, and distributed stream processing are all in the scope of this topic.

Focus areas (not limited to):

- Parallel, replicated, and highly-available distributed databases
- Cloud and HPC storage architectures and systems
- Scientific data analytics (Big Data or HPC based approaches)
- Middleware for processing large-scale data
- Programming models for parallel and distributed data analytics
- Workflow management for data analytics
- Coupling HPC simulations with in situ data analysis
- Parallel data visualization
- Distributed and parallel transaction, query processing and information retrieval
- Internet-scale data-intensive applications
- Sensor network data management
- Data-intensive clouds and grids
- Parallel data streaming and data stream mining
- New storage hierarchies in distributed data systems
- Parallel and distributed machine learning, knowledge discovery and data mining
- Privacy and trust in parallel and distributed data management and analytics systems
- IoT data management and analytics

TOPIC 5: CLUSTER, CLOUD AND EDGE COMPUTING

Global Chair: Radu Prodan, University of Klagenfurt, Austria
Local Chair: Luís Veiga, INESC-ID, Universidade de Lisboa, Portugal

This topic focus on the interdependencies between Cluster and Cloud Computing in addition to the results specifically addressing issues belonging only to one of these areas. In Cluster Computing, important research topics focus on performance, reliability, and energy efficiency as well as the impact of novel processor architectures. Research issues in Cloud Computing include various forms of virtualization and their impact on performance, resource management, and business models that address system owner and user interests. Further, it is interesting to address Cloud Computing on top of several smaller clusters and its advantages with respect to reliability and load balancing on a high abstraction level as well as the consideration of networks. Recently, Edge and Fog computing raised new challenges related to data management, security and privacy, decentralized infrastructure control, increased heterogeneity, device mobility, extreme scale monitoring, fault tolerance, energy efficiency and work distribution under resource constraints, to name a few.

Since many research studies in this area use experimental evaluation, we expect the authors reporting such studies to provide sufficient study details, if necessary complemented by a possibly web-based supplement, to allow a technical evaluation during the review process, and reproducibility and replicability of results if the submission is accepted.

Focus areas (not limited to):

- Cloud-enabled applications and platforms
- Interoperability and portability in Cloud Computing
- Aggregation and federation of Clouds
- Hybrid, Fog and Edge computing
- Energy efficiency in Cluster and Cloud Computing
- Resource/Service/Information discovery in Clouds
- Resource management and scheduling in Clusters and Clouds
- Cloud programming models, tools, and algorithms
- Dependability, adaptability, and scalability of Cloud applications
- Security and privacy for Clouds
- Workflow management in Clouds and Clusters
- Accounting, billing and business models for Cloud Computing
- Management of resources and applications in Clusters and Clouds
- Quality-of-Service and Service-Level-Agreement in Clouds
- Containers and serverless computing
- Decentralized infrastructure federation; blockchain
- Decentralized and peer-to-peer systems and applications

TOPIC 6: THEORY AND ALGORITHMS FOR PARALLEL AND DISTRIBUTED PROCESSING

Global Chair: Andrea Pietracaprina, Università di Padova, Italy
Local Chair: João Lourenço, Universidade Nova de Lisboa, Portugal

Nowadays parallel and distributed processing is ubiquitous. Multicore processors are available on smartphones, laptops, servers and supercomputing nodes. Also, many devices cooperate in fully distributed and heterogeneous systems to provide a wide array of services. Despite recent years have witnessed astonishing progress in this field, many research challenges remain open concerning fundamental issues as well as the design and analysis of efficient, scalable, and robust algorithmic solutions with provable performance and quality guarantees.

High quality, original papers are solicited on theoretical and algorithmic aspects of parallel and distributed computation.

Focus areas (not limited to):

- Theoretical foundations, models, and complexity issues
- Emerging paradigms for parallel and distributed computation
- Lower bounds
- Design, analysis and engineering of distributed and parallel algorithms
- Data structures for parallel and distributed algorithms
- Algorithms for combinatorial and graph problems
- Algorithms and models for big Data/Data-intensive computing
- Learning and mining algorithms
- Algorithms for routing and information dissemination in communication networks
- Algorithms for social networks
- Fault tolerant and self-stabilizing algorithms
- Power/energy-efficient algorithms
- Algorithms on GPUs, FPGAS and other accelerators
- Algorithms for cloud and edge computing
- Algorithmic game theory related to parallel and distributed systems
- Theoretical aspects of dependable, secure and privacy-preserving distributed systems

TOPIC 7: PARALLEL AND DISTRIBUTED PROGRAMMING, INTERFACES, AND LANGUAGES

Global Chair: Alfredo Goldman, University of São Paulo, Brazil
Local Chair: Ricardo Chaves, INESC-ID, Universidade de Lisboa, Portugal

Parallel and distributed applications require appropriate programming abstractions and models, efficient design tools, parallelization techniques and practices. This topic is open for the presentation of new results and practical experience in this domain: Efficient and effective parallel languages, innovative programming interfaces and software architectures, libraries and frameworks, as well as solid practical and experimental validation, towards high-performance, correct, portable, and scalable parallel programs.

Contributions that assess programming abstractions, models, and methods for improving: usability, performance prediction, scalability, and self-adaptation, or for providing: rapid prototyping and fault-tolerance are welcomed. They can be applied, for instance, in dynamic heterogeneous parallel and distributed infrastructures. The authors are invited to include quantitative evaluations of their claims.

Focus areas (not limited to):

- Programming paradigms and techniques for novel infrastructures like accelerators, exascale systems, low power architectures, and clouds
- Design and implementation, performance analysis and performance portability of programming models across parallel and distributed platforms
- Innovative paradigms, programming models, languages, and libraries for parallel and distributed applications
- Programming models and techniques for heterogeneity, self-adaptation and fault tolerance
- Programming tools for application design, implementation, and performance-tuning
- Application case-studies for benchmarking and comparative studies of parallel programming models
- Domain-specific libraries and languages
- Parallel and distributed programming productivity, usability, and component-based parallel programming
- Software architectures for Parallel and Distributed Computing

TOPIC 8: MULTICORE AND MANYCORE PARALLELISM

Global Chair: Enrique S. Quintana Ortí, Universitat Politècnica de València, Spain
Local Chair: Nuno Roma, INESC-ID, Universidade de Lisboa, Portugal

Modern homogeneous and heterogeneous multicore and manycore architectures are now part of the high-end, embedded, and mainstream computing scene and can offer impressive performance for many applications. This architecture trend has been driven by the need to reduce power consumption, increase processor utilization, and deal with the memory-processor speed gap. However, the complexity of these new architectures has created several programming challenges, and achieving performance on these systems is often a difficult task. This topic seeks to explore productive programming of multi- and manycore systems, as well as stand-alone systems with large numbers of cores like GPUs and various types of accelerators; this can also include hybrid and heterogeneous systems with different types of multicore processors. It focuses on novel research and solutions in the form of programming models, algorithms, languages, compilers, libraries, runtime and analysis tools to increase the programmability of multicore, many-core, and heterogeneous systems, in the context of general-purpose, high-performance, and embedded parallel computing.

Focus areas (not limited to):

- Programming techniques, models, frameworks and languages
- Compiler optimizations and techniques
- Lock-free algorithms, transactional-memories
- Libraries and runtime systems
- Tools for discovering and understanding parallelism
- Advances in algorithms and data-structures
- Hardware support for programming models and runtime systems
- Models, methods and tools for innovative many-core architectures
- Performance and power trade-offs and scalability
- Innovative applications and case studies

TOPIC 9: PARALLEL NUMERICAL METHODS AND APPLICATIONS

Global Chair: Stanimire Tomov, University of Tennessee, USA
Local Chair: Sergio Jimenez, Universidad de Extremadura, Spain

The need for high-performance computation is driven by the need for large-scale simulation and data analysis in science and engineering, finance, life sciences, etc. This requires the design of highly scalable numerical methods and algorithms that are able to efficiently exploit modern computer architectures. The scalability of these algorithms and methods and their ability to efficiently utilize high-performance heterogeneous resources is critical to improving the performance of computational and data science applications.

This conference topic aims to provide a forum for presenting and discussing recent developments in parallel numerical algorithms and their implementation on current parallel architectures, including many-core and hybrid architectures. We encourage submissions that address algorithmic design, implementation details, performance analysis, as well as integration of parallel numerical methods in large-scale applications.

Focus areas (not limited to):

- Numerical linear algebra for dense and sparse matrices
- Synchronization-reducing and communication-avoiding algorithms
- Optimization and non-linear problems
- Parallel metaheuristics and hyperheuristics
- Mixed precision algorithms exploiting low-precision hardware
- High-dimensional problems and reduction methods
- Numerical methods for large-scale data analysis
- Uncertainty quantification
- Applications of numerical algorithms in science, engineering, and data analysis

TOPIC 10: HIGH-PERFORMANCE ARCHITECTURES AND ACCELERATORS

Global Chair: Samuel Thibault, Université de Bordeaux, France
Local Chair: Pedro Tomás, INESC-ID, Universidade de Lisboa, Portugal

Different computing platforms provide distinct potentials for achieving massive performance, depending on how applications leverage the platform's architectural features or customization levels. Beyond general-purpose multi-processors, example computing platforms include graphics processing units (GPUs), multi-/many-core co-processors, as well as customizable devices, such as FPGA-based systems, streaming data-flow architectures or low-power embedded systems.

Hence, this topic explores new directions across this variety of architecture possibilities, with contributions along the whole design stack. On the architecture side, the scope spans system architectures, processor micro-architecture, multi-threading and the memory hierarchy, as well as the architectural support for parallelism, and the impact of emerging hardware technologies. On the compilation side, topics of interest include programmer productivity issues, concurrent and /or sequential language aspects, vectorization, program analysis, program transformation, automatic discovery and/or management of parallelism at all levels, autotuning and feedback directed compilation, and the interaction between the compiler and the system at large. On the runtime side, possible contributions include scheduling and coordination of accelerators, CPU, and network communications, debugging, benchmarking, profiling, modeling and simulation tools. Application-related submissions that contribute with new insights into fundamental architectural problems or solutions are also welcomed.

Focus areas (not limited to):

- Parallel computer architecture design, including ILP, DLP, multi-threadeding, and multi/many-core processors
- Application-specific, reconfigurable and embedded parallel systems
- New accelerator architectures
- Memory hierarchy, emerging memory technologies, and 3D stacked memories
- Compiling for general-purpose, heterogeneous and emerging processors/architectures
- Compiler, run-time, and architectural support for dynamic adaptation
- Compilers for Domain Specific Languages
- Interaction between compiler, runtime system, application, hardware, and operating systems
- Tools for debugging, profiling, and optimizing programs on heterogeneous architectures and accelerators
- Performance, power and energy models for accelerators
- Hardware designs and compiler optimizations for power-constrained and energy-efficient computing
- Software and hardware fault-tolerance techniques
- Hybrid and heterogeneous computing mixing several, possibly different types of processors and/or accelerators
- Parallel algorithms and applications for accelerators

SPONSORS

Google Springer