Lecture Notes in Computer Science **10353**

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

More information about this series at http://www.springer.com/series/7407

Narayan Desai · Walfredo Cirne (Eds.)

Job Scheduling Strategies for Parallel Processing

19th and 20th International Workshops, JSSPP 2015
Hyderabad, India, May 26, 2015
and JSSPP 2016, Chicago, IL, USA, May 27, 2016
Revised Selected Papers

Springer

Editors
Narayan Desai
Google
Seattle
USA

Walfredo Cirne
Google
Mountain View
USA

ISSN 0302-9743 ISSN 1611-3349 (electronic)
Lecture Notes in Computer Science
ISBN 978-3-319-61755-8 ISBN 978-3-319-61756-5 (eBook)
DOI 10.1007/978-3-319-61756-5

Library of Congress Control Number: 2017945740

LNCS Sublibrary: SL1 – Theoretical Computer Science and General Issues

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

This volume contains the papers presented at the 19th and 20th Workshops on Job Scheduling Strategies for Parallel Processing (JSSPP'2015 and JSSPP'2016). The JSSPP Workshops take place in conjunction with the IEEE International Parallel Processing Symposia.

The proceedings of previous workshops are also available from Springer as LNCS volumes 949, 1162, 1291, 1459, 1659, 1911, 2221, 2537, 2862, 3277, 3834, 4376, 4942, 5798, 6253, 7698, 8429, and 8828. These volumes are available as printed books and online.

In 2015, the workshop was help in Hyderabad, India, on May 26. In 2016, it took place in Chicago, USA, on May 27. For each year, 4 papers were submitted, of which we accepted seven. All submitted papers went through a complete review process, with the full version being read and evaluated by an average of four reviewers. We would like to especially thank the Program Committee members and additional reviewers for their willingness to participate in this effort and their detailed, constructive reviews. The Program Committee for 2015 and 2016 comprised:

- Henri Casanova, University of Hawaii at Manoa
- Julita Corbalan, Technical University of Catalonia
- Dick Epema, Delft University of Technology
- Hyeonsang Eom, Seoul National University
- Dror Feitelson, The Hebrew University
- Liana Fong, IBM T.J. Watson Research Center
- Eitan Frachtenberg, Facebook
- Alfredo Goldman, University of São Paulo
- Allan Gottlieb, New York University
- Alexandru Iosup, Delft University of Technology
- Morris Jette, SchedMD LLC (2015 only)
- Srikanth Kandula, Microsoft
- Rajkumar Kettimuthu, Argonne National Laboratory
- Dalibor Klusáček, Masaryk University
- Madhukar Korupolu, Google
- Zhiling Lan, Illinois Institute of Technology
- Bill Nitzberg, Altair Engineering
- P.-O. Östberg, Umeå University
- Larry Rudolph, MIT
- Uwe Schwiegelshohn, Technical University of Dortmund
- Leonel Sousa, Universidade Técnica de Lisboa
- Mark Squillante, IBM T.J. Watson Research Center
- Wei Tang, Google
- Ramin Yahyapour, GWDG, University of Göttingen

As a primary venue of the parallel scheduling community, the Job Scheduling Strategies for Parallel Processors Workshop offers a good vantage point to witness its evolution. During these two decades, we have seen parallel scheduling grow in scope and importance, following the popularization of parallel systems. Fundamental issues in the area remain relevant today (e.g., scheduling goal and evaluation, workload modeling, performance prediction). Meanwhile, a new set of issues have emerged, owing to the new workloads, increased scale, and differing priorities of cloud systems. Together, the traditional and new issues make for a lively and discussion-rich workshop, where academic researchers and participants from industry meet and exchange ideas and experiences.

The JSSPP Workshops traditionally start with a keynote talk. In 2015, Benjamin Hindman from Mesosphere explored how to leverage multilevel schedulers to separate concerns and better accommodate competing perspectives (e.g., scheduling goals for the resource provider can differ substantially from those of the user) in parallel scheduling. In 2016, we surveyed big challenges and open problems in modern parallel scheduling. This volume includes a summary of the 2016 keynote.

Following the trend of previous years, we see parallel scheduling challenges arising at multiple levels of abstractions. The days of shared-memory vs. message-passing parallelism are definitely over. Parallelism today happens at all levels, including combining different clusters or clouds at the user-level to support a target application.

For node-level parallelism, the driving forces are the simultaneous increase in capacity and heterogeneity of a single node. As the number of cores sharing the same memory increases (often introducing non-trivial communication topologies) and special purpose parallel processors (like GPUs) become prevalent, new approaches and research remain relevant.

Kang et al. show how to minimize energy consumption in task migration within a many-core chip. Many-core chips are also the environment targeted by Chu et al., who focus on how to space-share these chips among competing applications. Singh and Auluck explore the judicious use of task replication in the real-time scheduling context. Tsujita and Endo investigate a data-driven approach to schedule GPU load, using Cholesky decomposition as a concrete, relevant use case. Negele et al. evaluate the use of lock-free data structures in the OS scheduler. While requiring a complete reworking of the operating system, their results show a promising payoff.

Cluster-level parallelisms are also driven by increases in scale and heterogeneity. But their scale seems to expose more systemic effects on how people interact with them, giving rise to the need for sophisticated user and workload modeling. Along these lines, Schlagkamp investigates the relationship between user behavior (think time, more precisely) and parallel scheduling. Emeras et al. describe Evalix, a predictor for job resource consumption that makes novel use of user information. In such an environment, sophisticated and realistic simulation is another clear need. Dutot et al. present BatSim, a language-independent simulator that allows for different levels of realism in the simulation (at different computational costs).

On distributed scheduling itself, Pascual et al. explore how space-filling curves can lead to better scheduling of large-scale supercomputers. Li et al. also targeted large-scale supercomputers, particularly on how to better leverage the multidimensional torus topology of machines like Blue waters. Klusáček and Chlumsky rely on the multilevel

scheduler support of Torque to introduce a job scheduler based on planning and metaheuristics, in opposition to simple queueing. Breitbart et al. explore which jobs can be co-scheduled such that memory bandwidth does not become a bottleneck, therefore negating the benefits of co-scheduling. Zhuang et al. focus on how to improve the selection of a disruption time for a cluster, so as to reduce the impact on its users.

Another key part of the JSSPP experience is the discussion of real-life production experiences, providing useful feedback to researchers, as well as refining best practices. Klusáček et al. describe the reconfiguration of MetaCentrum, covering motivation, process, and evaluation. Particularly interesting is the fact that such work "was supported by a significant body of research, which included the proposal of new scheduling approaches as well as detailed simulations based on real-life complex workload traces", showcasing the productive synergy between top-notch research and production practice that takes place at JSSPP.

Enjoy the reading!

We hope you can join us in the next JSSPP workshop, this time in Orlando, Florida, USA, on June 2, 2017.

May 2017

Walfredo Cirne
Narayan Desai

Contents

JSSPP 2015

Controlled Duplication Scheduling of Real-Time Precedence Tasks on Heterogeneous Multiprocessors 3
Jagpreet Singh and Nitin Auluck

On the Design and Implementation of an Efficient Lock-Free Scheduler 22
Florian Negele, Felix Friedrich, Suwon Oh, and Bernhard Egger

Scheduling for Better Energy Efficiency on Many-Core Chips 46
Chanseok Kang, Seungyul Lee, Yong-Jun Lee, Jaejin Lee, and Bernhard Egger

Data Driven Scheduling Approach for the Multi-node Multi-GPU Cholesky Decomposition 69
Yuki Tsujita and Toshio Endo

Real-Life Experience with Major Reconfiguration of Job Scheduling System 83
Dalibor Klusáček, Šimon Tóth, and Gabriela Podolníková

EVALIX: Classification and Prediction of Job Resource Consumption on HPC Platforms 102
Joseph Emeras, Sébastien Varrette, Mateusz Guzek, and Pascal Bouvry

Influence of Dynamic Think Times on Parallel Job Scheduler Performances in Generative Simulations 123
Stephan Schlagkamp

JSSPP 2016

Automatic Co-scheduling Based on Main Memory Bandwidth Usage 141
Jens Breitbart, Josef Weidendorfer, and Carsten Trinitis

Adaptive Space-Shared Scheduling for Shared-Memory Parallel Programs . . . 158
Younghyun Cho, Surim Oh, and Bernhard Egger

Batsim: A Realistic Language-Independent Resources and Jobs Management Systems Simulator 178
Pierre-François Dutot, Michael Mercier, Millian Poquet, and Olivier Richard

Planning and Metaheuristic Optimization in Production Job Scheduler 198
Dalibor Klusáček and Václav Chlumský

Topology-Aware Scheduling on Blue Waters with Proactive Queue Scanning and Migration-Based Job Placement 217
Kangkang Li, Maciej Malawski, and Jarek Nabrzyski

Analyzing the Performance of Allocation Strategies Based on Space-Filling Curves 232
Jose A. Pascual, Jose A. Lozano, and Jose Miguel-Alonso

Choosing Optimal Maintenance Time for Stateless Data-Processing Clusters: A Case Study of Hadoop Cluster 252
Zhenyun Zhuang, Min Shen, Haricharan Ramachandra, and Suja Viswesan

Open Issues in Cloud Resource Management 274
Narayan Desai and Walfredo Cirne

Author Index 279