

Studies in Computational Intelligence

Data, Semantics and Cloud Computing

Volume 759

Series editor

Amandeep S. Sidhu, Biological Mapping Research Institute, Perth, WA, Australia
e-mail: dscc@biomap.org

More information about this series at <http://www.springer.com/series/11756>

Rong Kun Jason Tan · John A. Leong
Amandeep S. Sidhu

Optimized Cloud Based Scheduling

Rong Kun Jason Tan
Curtin Sarawak Research Institute
Curtin University
Miri, Sarawak
Malaysia

Amandeep S. Sidhu
Biological Mapping Research Institute
Perth, WA
Australia

John A. Leong
Curtin Sarawak Research Institute
Curtin University
Miri, Sarawak
Malaysia

ISSN 1860-949X ISSN 1860-9503 (electronic)
Studies in Computational Intelligence
Data, Semantics and Cloud Computing
ISBN 978-3-319-73212-1 ISBN 978-3-319-73214-5 (eBook)
<https://doi.org/10.1007/978-3-319-73214-5>

Library of Congress Control Number: 2017962036

© Springer International Publishing AG 2018

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

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

Contents

1	Introduction	1
1.1	Project Overview	1
1.2	Project Description	3
1.3	Project Objectives	4
1.4	Motivations and Significances of the Project	4
1.5	Outline	9
2	Background	11
2.1	Literature Reviews	11
2.1.1	Scale of Things—Peta and Exascale	11
2.1.2	Current Resource Availability	12
2.1.3	High Performance Computing	13
2.1.4	Cloud Computing	14
2.1.5	Existing HPC Implementation Architecture	19
2.1.6	Significance of HPC Clusters on Hybrid Cloud	24
2.1.7	Hybrid Cloud Future Improvements	25
2.1.8	HPC + Cloud Contrasted with Existing Implementation Architectures	27
2.1.9	Profit-Based Scheduling	28
2.1.10	Preemptable Scheduling	29
2.1.11	Priority-Based Scheduling	30
2.2	Various Aspects and Current Issues	31
2.2.1	Service Provisioning	31
2.2.2	Service Allocation	32
2.2.3	Service Adaptation	32
2.2.4	Service Mapping	33
3	Benchmarking	35
3.1	Cloud Benchmarking Instance Specifications and Assumptions	35

3.2	Classic Benchmark Test Categories	36
3.2.1	Dhrystone Benchmark	36
3.2.2	Linpack Benchmark	36
3.2.3	Livermore Loops	37
3.2.4	Whetstone Benchmark	37
3.3	Disk, USB and LAN Benchmarks	37
3.4	Multithreading Benchmarks	38
3.4.1	Simple Add Tests	39
3.4.2	Whetstone Benchmark	39
3.4.3	MP FLOPS Program	40
3.4.4	MP Memory Speed Tests	40
3.4.5	MP Memory Bus Speed Tests	40
3.4.6	MP Memory Random Access Speed Benchmark	40
3.5	OpenMP Benchmarks for Parallel Processing Performance	42
3.5.1	MemSpeed	42
3.5.2	Original Open MP Benchmark	42
3.6	Memory BusSpeed Benchmark	42
3.6.1	BusSpeed Test	42
3.6.2	Random Memory Benchmark	43
3.6.3	SSE Benchmark	43
4	Computation of Large Datasets	47
4.1	Challenges and Considerations in Computing	
	Large-Scale Data	48
4.1.1	Extreme Parallelism and Heterogeneity	48
4.1.2	Data Transfer Costs	48
4.1.3	Increased Failure Rates	49
4.1.4	Power Requirements at Exascale	49
4.1.5	Memory Requirements at Exascale	50
4.2	Work Done to Enable Computation of Large Datasets	50
4.2.1	OpenStack and Azure Implementation	51
4.2.2	Cloud Interconnectivity	56
4.2.3	HPC + Cloud	58
4.2.4	Simulations and Tools for Large Scale	
	Data Processing	59
4.3	Summary, Discussion and Future Directions	63
4.3.1	What Might the Model Look like?	63
4.3.2	Application Primitives—Key to Performance	64
5	Optimized Online Scheduling Algorithms	65
5.1	Dynamic Task Splitting Allocator (DTSA)	65
5.2	Procedural Parallel Scheduling Heuristic (PPSH)	67
5.2.1	Mapping Phase	68
5.2.2	Shuffling Phase	70

5.2.3	Eliminating Phase	73
5.2.4	Finishing Phase	75
5.3	Scalable Parallel Scheduling Heuristic (SPSH)	75
5.4	Calculation of Time	79
6	Performance Evaluation	83
6.1	Complexity Analysis	83
6.1.1	Algorithm 1: Dynamic Task Splitting Allocator (DTSA)	83
6.1.2	Algorithm 2: PPSH Phase 1—Mapping Phase	83
6.1.3	Algorithm 3: PPSH Phase 2—Shuffling Phase	84
6.1.4	Algorithm 4: PPSH Phase 3—Eliminating Phase	84
6.1.5	Algorithm 5: PPSH Phase 4—Finishing Phase	84
6.1.6	Algorithm 6: Scalable Parallel Scheduling Heuristic (SPSH)	84
6.2	Experimental Results	85
7	Conclusion and Future Works	91
	Bibliography	93

List of Figures

Fig. 1.1	Five tier cloud computing structure	6
Fig. 1.2	Details of cloud computing structure	6
Fig. 2.1	Sequencing progress versus compute and Storage	12
Fig. 2.2	Current cloud services	18
Fig. 2.3	On premise local HPC cluster	20
Fig. 2.4	HPC cluster implemented on grids	22
Fig. 2.5	HPC cluster implemented on the cloud	23
Fig. 2.6	HPC cluster implemented on the hybrid cloud	24
Fig. 2.7	Key distinctive of various HPC implementation architectures	28
Fig. 3.1	Classic benchmark results (higher value denotes better performance).	38
Fig. 3.2	Disk, USB and LAN benchmarks (higher value denotes better performance).	39
Fig. 3.3	Multithreading benchmarking (higher value denotes better performance).	41
Fig. 3.4	OpenMP benchmarking (higher value denotes better performance).	43
Fig. 3.5	Memory BusSpeed benchmark results (higher value denotes better performance).	44
Fig. 4.1	Open source cloud computing framework.	52
Fig. 4.2	Architecture of three nodes in an open source cloud	53
Fig. 4.3	Windows Azure Pack (WAP) architecture	55
Fig. 4.4	Virtual network topology	57
Fig. 4.5	On premises network detailed topology	57
Fig. 4.6	HPC + Cloud framework	58
Fig. 4.7	Completion time of 500 tasks for local cluster and hybrid environment	61
Fig. 4.8	Basic infrastructure of hybrid cloud environment	63
Fig. 5.1	DTSA model	66
Fig. 5.2	PPSH workflow	67

Fig. A.1	Simulation results of single node with multi-core.	72
Fig. A.2	Simulation results of two nodes with multi-core.	72
Fig. A.3	Simulation results of three nodes with multi-core.	72
Fig. 5.3	Scalable levels in IaaS cloud environment	76
Fig. 5.4	Service instance passive delay (SIPD)	80
Fig. 6.1	Parallel performance in single node	86
Fig. 6.2	Parallel performance between single and two nodes.	87
Fig. 6.3	Parallel performance between two and three nodes.	88
Fig. 6.4	Simulation results of SPSH	89

List of Tables

Table 1.1	Comparison between grid and cloud computing	5
Table 1.2	Comparisons of IAAS, PAAS and SAAS.	5
Table 1.3	Comparisons between traditional and big data	7
Table 1.4	Outline	8
Table 2.1	Windows Azure benchmark results.	17
Table 2.2	Performance metrics for algorithms	31
Table 3.1	Cloud benchmarking instance specifications	36
Table 3.2	Classic benchmark test results (higher value denotes better performance).	36
Table 3.3	Disk, USB and LAN benchmark test results.	38
Table 3.4	Multithreading benchmarking test results (higher value denotes better performance).	41
Table 3.5	OpenMP benchmarking test results (higher value denotes better performance).	42
Table 3.6	Memory BusSpeed benchmark results (higher value denotes better performance).	43
Table 3.7	Overall results.	44
Table 4.1	Tasks details	59
Table 4.2	Local cluster node specifications.	60
Table 4.3	Public cloud specifications	60
Table 4.4	Packages of SOAP	62
Table 6.1	Comparisons between SPSH and algorithms on data intensive task	85
Table 6.2	Performance comparisons between N-core to N-core processing.	86
Table 6.3	Performance comparisons between single node N-core to two nodes N-core processing	87
Table 6.4	Performance comparisons between two nodes N-core to three nodes N-core processing	88

List of Algorithms

Algorithm 1	Dynamic Task Splitting Allocator (DTSA).....	83
Algorithm 2	PPSH Phase 1—Mapping Phase.....	83
Algorithm 3	PPSH Phase 2—Shuffling Phase.....	84
Algorithm 4	PPSH Phase 3—Eliminating Phase.....	84
Algorithm 5	PPSH Phase 4—Finishing Phase	84
Algorithm 6	Scalable Parallel Scheduling Heuristic (SPSH).....	84