Ciência e Natura, v. 37 Part 2 2015, p. 427−433

ciência e natura

ISSN impressa: 0100-8307 ISSN on-line: 2179-460X

A Study on Factors Contributing to Efficiency of Scheduling Algorithms in a Cloud Computing Environment; Overview of Several Algorithms

Minoo Soltanshahi[1]

[1] DEPARTMENT OF COMPUTER ENGINEERING, KERMAN COMPUTER SOFTWARE ENGINEER BRANCH , ISLAMIC AZAD UNIVERSITY, KERMAN IRAN, minoo.soltanshahi@gmail.com

Aliakbar Niknafs[2]

[2] DEPARTMENT OF COMPUTER ENGINEERING, SHAHID BAHONAR UNIVERSITY OF KERMAN, IRAN ,Niknafs@uk.ac.ir

Abstract

Cloud computing is the latest distributed technology providing a rich environment of dynamically shared resources through virtualization, which can fulfill the requirements of users by allocating resources to programs. Any program in a cloud environment is delivered by workflows which are a series of interlinked tasks to accomplish a goal. One of the most important tasks in cloud computing is correct mapping of tasks onto resources. It is essential to schedule processes in distributed systems such as cloud, since it leaves a tremendous impact on the system performance. This is done by scheduling algorithms. Therefore, it is crucial to present and adopt an efficient algorithm in the cloud environment. This article attempted to examine the parameters effective in the efficiency of scheduling algorithms including deadline, cost constraint, balanced loading, power consumption and fault tolerance. Additionally, the performances of several algorithms were briefly discussed.

Keywords: *scheduling algorithm, deadline, cost constraint, balanced loading, fault tolerance*

1 Introduction

Information and communication technologies, including the Internet, which has become a critical component of human life today, are expanding day by day. On the other hand, there is a growing trend in how individuals in a community require advanced technologies so as to accelerate theirs tasks, reduce costs, mutual participation, provide fast and dynamic access to resources etc. Nowadays, the technology capable of fulfilling such requirements is known as cloud computing, which refers to development and deployment of computer technology based on the Internet. It is a method of computing in a space where the IT-related capabilities are supplied as a service(s) to the user (e.g. software as a service, infrastructure as a service, platform as a service and so on), thus allowing the user to gain access to the technology-based services on the Internet, without specialized knowledge about such technologies [6]. Cloud computing is structured like a mass of cloud through which users can access resources anywhere in the world [5]. Therefore, it is essential to take more efficient advanage of the resources, which constitutes a key subject matter numerous researchers are currently working on. One of the factors contributing to efficient performance of clouds involves scheduling algorithms whose task is to propoerly provide mapping on resources. There have been numerous relevant studies where each of these algorithms evaluated certain parameters (e.g. cost, time, fault tolerance, balanced loading, power consumption). However, only a few managed to respond desirably to the cloud environment. Each algorithm entailed its advantages and disadvantages. Hence, the cloud technology requires application of specific methods that can enhance efficiency to a great extent. For that purpose, the environmental circumferences need to be first examined. Then, several optimum solutions can be proposed so as to refine the algorithms.

2 Overview of parameters contributing to scheduling algorithms in a cloud computing environment

When the user delivers a task to the cloud to be completed, it will be processed as a workflow within the environment. The workflows in a cloud computing environment are usually displayed by a Directed Acyclic Graph (DAG) where the tasks and their interconnections are represented by nodes and edges, respectively. This graph may entail a balanced or unbalanced structure. In the former structure, the nodes are equal with sequential dependence, while the nodes in the latter structure are unequal with non-sequential dependencies. In this scenario, the scheduling and execution of tasks would become more complex, which should be accomplished based on the depth of tasks and other measures [1]. The tasks are scheduled within two modes of shared space and shared time. In the former, the allocated space is exclusive dedicated to the same task until it is finished, whereas the latter mode assigns the resources to respective tasks in a shared nonexclusive procedure, continuously available until the tasks are finished [2].

There are a great number of parameters that can affect the efficiency of scheduling algorithms. A cloud computing system can be more optimally utilized through fulfilling the involved criteria, which are discussed from two perspectives: Firstly, the user's perspective such as meeting deadlines and cost constraints. And secondly, the system's perspective such as balanced loading, fault tolerance and power consumption, which shall be described later.

2.1 Deadline

This term represents the time proposed for completion of a workflow usually specified by the user. The deadlines for programs can be divided into two categories: hard and soft deadlines. A soft deadline entails fault tolerance higher than the hard one. If the workflow fails to be completed within the predefined time, it may not encounter any serious problems. However, the workflow in hard deadline should operate properly based on the strict deadline; otherwise the system would be undermined [12].

2.2 Cost constraint

The workflows are typically implemented through different costs. A cost constraint refers to how much it takes a workflow to be properly completed. This is essential from the user's viewpoint and is usually recommended by the user or constitutes one of the Quality of Service (QoS) options. Alternatively, it should be paid according to the Service-Level Agreement (SLA) signed between cloud providers and users.

2.3 Fault Tolerance

Sometimes, workflows are not completed under the previously determined conditions. For instance, a workflow may fail to complete by a certain deadline. Hence, any scheduling algorithm must entail mechanisms in such situations so as to estimate the failure factor of the workflow and prevent its failure. Alternatively in case failure occurrs, the scheduling algorithm should adopt effective strategies within the shortest time at minimal cost so as to compensate and properly complete the workflow.

2.4 Balanced loading

The resources are better utilized inside a cloud computing environment when the tasks are loaded on resources through a balanced procedure. Otherwise, some of the resources may entail a heavy load while others remain idle, which in turn can cause many problems, including: If the input tasks queue of a resource is too populated, there will be longer latency in accomplishment of tasks. In this scenario, the workflow fails or increases the implementation costs. Moreover, if some services remain idle, energy will be wasted. It is therefore crucial to apply certain balanced loading mechanisms in scheduling algorithms that can leave great impact on system's performance.

2.5 Power consumption

Employment of a wrong task scheduling technique can lead to power dissipation in the cloud environment. The system's efficiency can be improved through proposition and application of solutions properly performs accomplishing the workflows at minimal power consumption [3].

3 A review of how several scheduling algorithms function

There are different types of cloud computing environments, including public, private, hybrid, etc. They cover homogeneous and heterogeneous resources, sharing software, hardware, infrastructure and so on, thus responding to the needs of clients through the virtualization technology [4], [5] and [6] .The cloud providers offer their services based on quality usually specified by the clients or service level agreement specifying the quality parameters requested as signed between clients and cloud providers [7]. The efficiency of cloud environments can be enhanced through taking several measures. For instance, it is essential to allocate the resources, concerning which numerous algorithms have so far been proposed. Each algorithm concentrates on specific parameters (e.g. time and cost constraints) entailing certain advantages and disadvantages discussed briefly as follow.

Particle Swarm Optimization (PSO): This algorithm is a global optimization method supporting problems, solutions to which involve a point or level in an N-dimensional space. In PSO, an initial velocity is assigned to the particles and a range of communication channels are considered. Then, the particles begin to travel within the solution space as the results are calculated based on an eligibility criterion after each period. Over time, the particles accelerate toward particles with greater eligibility within the identical communication group. Its main advantage is the high number of swarm particle making the technique sufficiently flexible against the local optimal solution. Furthermore, the PSO is one of the most popular collective intelligence optimization algorithms owing to its simplicity and high-efficiency. However, it comes with several drawbacks such as premature local convergence, loss of useful matrix data of each particle, redundancy, etc. That eventually led to proposition of several types of such algorithm capable of overcoming the disadvantages as much as possible [8]. For instance, the Revised

Discrete Particle Swarm Optimization (RDPSO) was experimented together with PSO and Best Resource Selection (BRS), the results of which demonstrated it was more efficient within the specified makespan and could curtail the costs [18].

The Novel Particle Swarm Optimization (NPSO): The NPSO has implemented the scheduling of resources through the PSO, where the algorithms and functions are similar, while the NPSO can estimate the deadline at a lower cost proportionate to the workflow variations in a dynamic procedure [9].

The DWSGA is a hybrid heuristic method based on the genetic algorithm, accelerating the task completion and distribution of workflows on the resources. It can be employed in workflows with balanced and unbalanced structures. Each task in the DWSGA is prioritized based on its impact on other tasks, according to which the scheduling process is done. Furthermore, this algorithm adopts the Best Fit and Round Robin for allocating resources. It functions more efficiently as compared to the GVNS which is a hybrid genetic algorithm, VNS (neighbor search algorithm) intending to shorten the completion cycle of workflows in spite of lengthy algorithm runtime, and the DCLS focusing on reduction of specified time [1].

The IaaS Cloud Partial Critical Paths (IC-PCP) and IaaS Cloud Partial Critical Paths with Deadline Distribution (IC-PCPD2) are two algorithms developed from Partial Critical Paths (PCP). The PCP algorithm has two phases:

Failure and deadline distribution between workflow tasks

Scheduling of any task on the cheapest service which can be properly completed before the deadline is met.The PCP algorithm is exclusive to static environments with limited resources provided for grades using homogenous bandwidth. It is not applicable to real cloud environments, which is why the two developed versions of it known as IC-PCP and IC-PCPD2 are more appropriate for a cloud environment. The IC-PCP is a single-phase algorithm which, similar to PCP, distributes the deadline between tasks. Moreover, it can provide mapping through finding an existing or new sample capable of implementing the critical path (i.e. longest path from a start node to finish node of a workflow) prior to the deadline, with an exception that IC-PCP schedules each workflow task on an appropriate predefined deadline service rather than assigning a segment of the deadline to the task. In contrast, the IC-PCPD2 is a double-phase algorithm functioning similar to PCP except: Firstly, it adopts the new assignment policy consistent with the new pricing model, and secondly, it tends to employ as much as possible the remaining time of the existing sample in the computing service. In case it does not work out, the algorithm would use a new sample of the service. Finally, both algorithms IC-PCP and IC-PCPD2 entail time complexity O(n^2), employing the pricing model based on payment at specified intervals. According to the results obtained by several experiments, however, it can be argued that IC-PCP can estimate the time and cost constraints more efficiently than IC-PCPD2 does [10].

Additionally, there are three algorithms proposed to improve the time and cost of executing workflows, including Deadline Budget Distribution based cost-time Optimization (DBD-CTO), Dispensation Time-Cost (DTC) and Greedy Cost (GC), among which the results indicated that DTC is most efficient followed by GC and DBD-CTO [11].

The Fault-Tolerant Scheduling Using Spot Instances (FTSUSI) is another algorithm that employs scheduling techniques to enhance the fault tolerance through creating checkpoints specified periodically by the user at certain frequencies in occasions where there is a limited deadline. Moreover, it can be ideal for a dynamic cloud environment owing to the capability to estimate deadlines at minimal cost. The FTSUSI is based on work history, assessing a critical path for each task and calculating the slack time or the difference between deadline and critical path. This serves to attain higher productivity out of slack time of the computing service samples. In addition, this algorithm employs the combination of two pricing models so as to reduce costs:

1. Employment of a specified service sample: In this procedure, the users offer the prices to the service.

2. Payment per use: In this method, the payment is made based on the time service has been used [12].

Fault Tolerant Workflow Scheduling (FTWS): This algorithm similarly serves to enhance the fault tolerance in cases of failure to execute within the specified deadline, functioning through two replication techniques and retransmission of tasks taking into account their priorities. Since replication alone leads to a waste of resources and retransmission alone lengthens the deadline, the FTWS runs a tradeoff between the two factors without resorting to the work history, but rather functioning by allocation of deadline between the critical path tasks [13].

Token-Based Heuristic Algorithm (TBHA): This algorithm intends to load the resources in a balanced procedure based on tokens by passing the shoulder across the factors. Moreover, it can be a desirable solution for large-scale clouds. In this scenario, the factors are independent of any knowledge concerning their neighbors, since they construct knowledge based on the previously received tokens. Hence, there are no redundant communications and checks, which in turn lead to higher efficiency and lower costs. Finally, it provides a quick decision-making routine, where the only drawback is that the tokens are lost [14].

Round Robin: The processing in this algorithm is divided between all the processors. In fact, the workflow is distributed between processors, even though the processing time varies for different processes probably leaving certain heavy-loaded resources idle [15].

Connection Mechanism: This algorithm functions based on counting the number of connections for each service that specifies the loading and accordingly maps the tasks [16].

The balanced loading algorithm with a concentrated node entails a central node in charge of decision -making. Hence, it comes with large overhead as the correction algorithm prevents the overhead by dividing the concentrated node into several smaller nodes and then employs them based on priority for balanced distribution of workflow mainly serving to curtail costs [17].

Power Consumption Optimization Algorithm (PCOA): This algorithm can be used to prevent resources from wasting energy within a cloud computing environment. By searching through the entire scheduling solutions implementable on corresponding service graphs, the PCOA selects algorithms capable of fulfilling the time and cost constraints lower than the service level agreement. Subsequently, it calculates the power consumption for each constraint and eventually selects an algorithm requiring minimum power [3].

4 Conclusion

Regarding the advances in technology and its increasing impact on human life, any utilization should be as optimal as possible. One of such new technologies is the Internet-based cloud computing which has undergone dramatic progress among organizations, businesses for carrying a tremendous number of tasks. This technology can be utilized most efficiently through first evaluating its environment and circumstances, followed by proposing optimal solutions in order to enhance efficiency. One of the most controversial issues in this technology involves the mapping of tasks onto resources, which is accomplished by scheduling algorithms. According to the existing conditions , a scheduling algorithm is supposed to select the best service where each algorithm can make the correct choice based on criteria such as fulfilling the cost and time constraints, building a balanced loading on the system etc. A correct mapping can significantly influence the efficiency of large-scale distributed environments such as cloud spaces. Hence, this paper attempted to examine the key parameters contributing to the efficiency of scheduling algorithms and then review the performance of several scheduling algorithms. Given the above facts, it was found out that accomplishment of a workflow involves various issues, some of which are important from the user's viewpoint that should be taken into account. Moreover, there are issues critical from the cloud provider's perspective, where each scheduling algorithm entails certain advantages and disadvantages. Effort has been made to provide solutions to the cloud-related problems, even though few have proved desirable, i.e. capable of optimally employ the environmental resources according to the requirements made by the user.

References

Yalda Aryan and Arash Ghorbannia Delavar ," a bi-objective workflow application scheduling in cloud computing systems", International Journal on Integrating Technology in Education (IJITE) Vol.3, No.2, June 2014

Mayanka Katyal and Atul Mishra ," A Comparative Study of Load Balancing Algorithms in Cloud Computing Environment" International Journal of Distributed and Cloud Computing Volume 1 Issue 2 December 2013

yonghong luo and shuren zhou ," Power Consumption Optimization Strategy of Cloud Workflow Scheduling Based on SLA" WSEAS TRANSACTIONS on SYSTEMS ,Yonghong Luo, Shuren Zhou, E-ISSN: 2224-2678 , Volume 13, 2014

Sara Qaisar and Kausar Fiaz Khawaja ,"cloud computing: network/security threats and countermeasures" , interdisciplinary journal of contemporary research in business - january 2012 vol 3, no 9

Bahman Rashidi and Mohsen Sharifi and Talieh Jafari , "A Survey on Interoperability in the Cloud Computing Environments" Published Online July 2013 in MECS (http://www.mecs-press.org/) DOI: 10.5815/ijmecs.2013.06.03

Pankaj Sareen , " Cloud Computing: Types, Architecture, Applications, Concerns, Virtualization and Role of IT Governance in Cloud" , International Journal of Advanced Research in Computer Science and Software Engineering ,Volume 3, Issue 3, March 2013

ranjit singh and sarbjeet singh ," score based deadline constrained workflow scheduling algorithm for cloud systems" , international journal on cloud computing: services and architecture (ijccsa) ,vol.3, no.6, december 2013

Kennedy, J. and Eberhart, R. C., "Particle Swarm Optimization", Proceedings of IEEE International Conference on Neural Networks, Piscataway, NJ, pp. 1942-1948, 199

R. Pragaladan and R. Maheswari ," Improve Workflow Scheduling Technique for Novel Particle Swarm Optimization in Cloud Environment " , International Journal of Engineering Research and General Science Volume 2, Issue 5, August-September, 2014 ISSN 2091-2730

Saeid Abrishami and Mahmoud Naghibzadeh and Dick H.J. Epema ," Deadline-constrained workflow scheduling algorithms for Infrastructure as a Service Clouds " , Future Generation Computer Systems 29 (2013) 158–169 , journal homepage: www.elsevier.com/locate/fgcs

Pooja, Naveen Kumari ," Performance Evaluation of Cost-Time Based Workflow Scheduling Algorithms in Cloud Computing " , International Journal of Advanced Research in Computer Science and Software Engineering , Volume 3, Issue 9, September 2013 ISSN: 2277 128X

Deepak Poola and Kotagiri Ramamohanarao and Rajkumar Buyya ," Fault-Tolerant Workflow Scheduling Using Spot Instances on Clouds ",Procedia Computer Science Volume 29, 2014, Pages 523–533 ICCS 2014. 14th International Conference on Computational Science

Jayadivya S K and Jaya Nirmala S and Mary Saira Bhanu S ," Fault tolerant workflow scheduling based on replication and resubmission of tasks in Cloud Computing "Jayadivya S K et al. / International Journal on Computer Science and Engineering (IJCSE), ISSN : 0975-3397 Vol. 4 No. 06 June 2012

Yang Xu, Lei Wu, Liying Guo, Zheng Chen , Lai Yang and Zhongzhi Shi ," An Intelligent Load Balancing Algorithm Towards Efficient Cloud Computing " , AI for Data Center Management and Cloud Computing: Papers from the 2011 AAAI Workshop (WS-11-08)

Zhong Xu and Rong Huang,(2009)"Performance Study of Load Balanacing Algorithms in Distributed Web Server Systems", CS213 Parallel and Distributed Processing Project Report

P.Warstein and H.Situ and Z.Huang(2010), "Load balancing in a cluster computer" In proceeding of the seventh International Conference on Parallel and Distributed Computing, Applications and Technologies IEEE

Parveen Jain and Daya Gupta ,” An Algorithm for Dynamic Load Balancing in Distributed Systems with Multiple Supporting Nodes by Exploiting the Interrupt Service” , International Journal of Recent Trends in Engineering, Vol 1, No. 1, May 200

Zhangjun Wu , Zhiwei Ni , Lichuan Gu and Xiao Liu , “A Revised Discrete Particle Swarm Optimization for Cloud Workflow Scheduling” Published in:Computational Intelligence and Security (CIS), 2010 International Conference on, Page(s):184 – 188, Publisher:IEEE