Future Generation Computer Systems 51 (2015) 61–71

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Resource-aware hybrid scheduling algorithm in heterogeneous distributed computing

Mihaela-Andreea Vasile [a], Florin Pop [a,*], Radu-Ioan Tutueanu [a], Valentin Cristea [a], Joanna Kołodziej [b]

[a] *Faculty of Automatic Control and Computers, University Politehnica of Bucharest, Romania*

[b] *Institute of Computer Science, Cracow University of Technology, Poland*

HIGHLIGHTS

- We proposed a hybrid approach for tasks scheduling in Heterogeneous Distributed Computing.
- The proposed algorithm considers hierarchical clustering of the available resources into groups.
- We considered different scheduling strategies for independent tasks and scheduling for DAG scheduling.
- We analyze the performance of our proposed algorithm through simulation by using and extending CloudSim.

ARTICLE INFO

Article history:
Received 4 August 2014
Received in revised form
29 October 2014
Accepted 27 November 2014
Available online 9 December 2014

Keywords:
Resource provisioning
Task scheduling
Clustering
Heterogeneous distributed computing
Multimedia content-driven applications

ABSTRACT

Today, almost everyone is connected to the Internet and uses different Cloud solutions to store, deliver and process data. Cloud computing assembles large networks of virtualized services such as hardware and software resources. The new era in which ICT penetrated almost all domains (healthcare, aged-care, social assistance, surveillance, education, etc.) creates the need of new multimedia content-driven applications. These applications generate huge amount of data, require gathering, processing and then aggregation in a fault-tolerant, reliable and secure heterogeneous distributed system created by a mixture of Cloud systems (public/private), mobile devices networks, desktop-based clusters, etc. In this context dynamic resource provisioning for Big Data application scheduling became a challenge in modern systems. We proposed a resource-aware hybrid scheduling algorithm for different types of application: batch jobs and workflows. The proposed algorithm considers hierarchical clustering of the available resources into groups in the allocation phase. Task execution is performed in two phases: in the first, tasks are assigned to groups of resources and in the second phase, a classical scheduling algorithm is used for each group of resources. The proposed algorithm is suitable for Heterogeneous Distributed Computing, especially for modern High-Performance Computing (HPC) systems in which applications are modeled with various requirements (both IO and computational intensive), with accent on data from multimedia applications. We evaluate their performance in a realistic setting of CloudSim tool with respect to load-balancing, cost savings, dependency assurance for workflows and computational efficiency, and investigate the computing methods of these performance metrics at runtime.

1. Introduction

We are surrounded by multimedia devices, whether they are sources (smart-phones, cameras, drones) or devices that display multimedia information (computers, tablets, smart-phones, TVs), and also by new multimedia content-driven applications, needed in various domains (scientific research, education, healthcare, etc.). A massive amount of heterogeneous data (video, audio, photos), organized in different formats, is created and requires gathering, processing and then aggregation in a fault-tolerant, reliable and secure heterogeneous distributed system. Different Cloud solutions are used to store, deliver and process this data. Heterogeneous distributed systems (HDC) may be created as a mixture of Cloud systems (public or private), mobile devices networks,

* Corresponding author.
E-mail addresses: mihaela.vasile@cti.pub.ro (M.-A. Vasile), florin.pop@cs.pub.ro (F. Pop), radu.tutueanu@cti.pub.ro (R.-I. Tutueanu), valentin.cristea@cs.pub.ro (V. Cristea), jokolodziej@pk.edu.pl (J. Kołodziej).

http://dx.doi.org/10.1016/j.future.2014.11.019
0167-739X/

desktop-based clusters, etc., and are able to handle the current challenges in managing large heterogeneous data sets in a limited amount of time [1]. In this context, dynamic resource provisioning for Big Data application scheduling became a challenge in modern High-Performance Computing (HPC) systems [2,3]. Unfortunately, most existing scheduling algorithms for tasks with dependencies in HDC systems do not consider reliability requirements of interdependent tasks and many times the execution fails because the tasks are allocated to unsuitable resources [4].

Modern HPC systems should be able to face variable demand loads in an efficient way when referring to the resources utilization [5] (use the minimum number of resources) or SLA achievement. A resource provisioning mechanism is an important element that should be used to manage utilization of available resources and to detect if there has been reached peak demand at a certain moment and the system should be extended by requesting additional resources (from a private Cloud for example), or if the system could be shrunk because there are just few requests and check if the QoS requirements of current applications will be still met [6].

Scheduling is a key problem in the context of Cloud computing, virtualization and Big Data because scheduling techniques also have to evolve along with HPC systems. Classic schedulers have been built for batch, homogeneous environments, having a static internal behavior with no or very few changes in the resources structure. The emergent modern systems are defined by highly heterogeneous environments with variable structure—new resources may be added if needed and removed when the work load decreases (on demand provisioning). The distributed computations as a divisible load scheduling problem for MapReduce is another new approach [7]. In this approach, a divisible load model of the computation, and two load partitioning algorithms are proposed. Novel scheduling approaches have to monitor the system structure and adjust the planning according to the real-time situation. This will also provide increased fault-tolerance because changes are always expected (either planned or not).

A workflow [8,9] describes the automation of a process, be it a business, scientific or general process and the set of rules (dependencies). In other words, workflows represent sets of elementary tasks and their dependencies, required for more complex purposes. So, the workflows are described by a DAG having nodes for each task and edges for each dependency. According with [10] the workflow execution should respond dynamically to interference of real-time monitoring and real-time execution, to support the experimentation process in HPC and in Big Data platforms. So, efficient scheduling of concurrent workflows becomes an important issue in HDC environments [11].

In this paper we present an extension of HySARC^2 scheduling algorithm for dependent tasks scheduling. The algorithm was introduced in [12] as a scheduling algorithm that improves workload on the resources available into the Cloud and satisfies tasks requirements. The HySARC^2 algorithm has three parts: (i) Analyze the available resources and group them into clusters (resource aware algorithm); (ii) Provision different groups of similar tasks to different clusters of resources; and (iii) Schedule the tasks in each cluster of resources. The initial version of HySARC^2 was applied for Bag-of-Tasks applications such as data mining algorithms or Monte Carlo simulations, having both IO and computational intensive phases. We extend the HySARC^2 applicability to applications modeled with workflows. We considered Modified Critical Path or Earliest Time First algorithms in our hybrid approach. The proposed scheduling algorithm allows a more efficient and exact structure of resources, because the tasks are classified before the scheduling phase and assigned to suitable groups of resources.

Our contributions in this paper can be summarized as follows:

- First, we proposed a hybrid approach for tasks scheduling in HDC considering both tasks and resources clustering. We acknowledge that there is no solution that could fit all kinds of tasks models. Therefore, our scheduling strategy is based on using different scheduling strategies, selected by taking into consideration both the heterogeneity of computing resources, and application tasks and/or flows. The effectiveness of the utilization of the computing resources is determined by the efficiency of the allocation strategy of the resources.
- We provided a model for adaptive and dynamic clustering considering the abstract modeling of HDC resources. The "distance" computed in this model highlights the similarity between resources, so we can group the resources in clusters and then use the formed clusters in HySARC^2 algorithm.
- We extended CloudSim [13] to consider this approach and we integrate four scheduling strategies: two for independent tasks (Earliest Deadline First and Shortest Job First) and two for DAG scheduling (Modified Critical Path and Earliest Time First).

The structure of the paper is the following. In Section 2 we analyze some existing scheduling solutions in the related work. Section 3 describes in detail the proposed approach: architecture, clustering and scheduling algorithms and the extensions for workflows. We analyze the total execution time and scalability in Section 4 as experimental methodology. Section 5 presents the integration in real Cloud platforms of our solution and in the last part, Section 6, are presented the conclusions and future research work.

2. Related work

The goal of HySARC^2 is to improve scheduling in a given Cloud environment, using adequate resources to satisfy both user requirements and service provider's interests, achieved by clustering and resource labeling. Therefore, topics of interest for this paper are clustering, resource provisioning, hybrid scheduling and algorithms for scheduling different types of tasks.

Cloud service providers are interested in optimizing available resources, in order to being able to satisfy as many user requirements as possible and as a result improving the profit. Efficient energy management is a challenging research issue in resource management in Cloud [14,15]. The HySARC^2 algorithm aims at efficient resource utilization: tasks assigned to suitable resources, having as effect energy saving because inadequate resources could be put in a hibernate state, in the limits of the Service Level Agreement (SLA) [16,17]. We will describe several solutions that take into consideration the resource allocation.

A Resource Aware Scheduling Algorithm which leverages two existing task scheduling algorithms, Min–min and Max–min, is described in [18]. Both algorithms use an estimation of tasks completion time and resource execution time. The presented algorithm alternates the two algorithms depending on input tasks.

An important feature for scheduling algorithms is to have a dynamic behavior according to real environment evolution. Such an algorithm is described in [19]. The algorithm is suitable for arbitrary constraints tasks, their dependencies may be organized as a graph, having as nodes the tasks and as edges the constraints. It consists in two parts: an initial scheduling phase and a rescheduling phase, in which tasks are separated in entry tasks and inner tasks, based on dependency of failing tasks. Depending on the type the node that fails, there may be used different scheduling algorithms: Highest Level First with Estimated Times, Modified Critical Path or Earliest Time First.

In [20] an algorithm having good results on the compromise cost-execution time is presented. The tests showed that the cost