Future Generation Computer Systems 82 (2018) 29–40

Contents lists available at ScienceDirect

Future Generation Computer Systems

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

Multi criteria biased randomized method for resource allocation in distributed systems: Application in a volunteer computing system

Javier Panadero [a,*], Jesica de Armas [b], Xavier Serra [a], Joan Manuel Marquès [a]

[a] IN3 - Computer Science Department, Open University of Catalonia, Barcelona, Spain
[b] Department of Economics and Business, Universitat Pompeu Fabra, Barcelona, Spain

HIGHLIGHTS

- We propose Multi Criteria Biased Randomized (MCBR), a novel selection method for large-scale systems composed of unreliable nodes.
- MCBR allows to select the most suitable nodes in an efficient and fast way, ensuring a minimum QoS to the users.
- MCBR is based on a Lexicographic Ordering (LO) multicriteria optimization strategy.

ARTICLE INFO

Article history:
Received 14 July 2017
Received in revised form 24 October 2017
Accepted 23 November 2017
Available online 18 December 2017

Keywords:
Distributed computing
Volunteer systems
User assignment
Allocation methods
Resource provisioning

ABSTRACT

Volunteer computing is a type of distributed computing in which a part or all the resources (processing power and storage) necessary to run the system are donated by users. In other words, participants contribute their idle computing resources to help running the system. Due to the fact that the nodes which compose the system are provided by a large number of users instead of a single (or a few) institution, a main drawback of volunteer computing is the unreliability of these nodes. For this reason, the selection of nodes to be involved in each task becomes a key issue. In this paper, we propose the Multi Criteria Biased Randomized (MCBR) method, a novel selection method for large-scale systems that use unreliable nodes. MCBR method is based on a multicriteria optimization strategy. We evaluated the method in a microblogging social network formed by a large number of microservices hosted in nodes voluntarily contributed by their participants. Simulation results show that our proposal is able to select nodes in a fast and efficient manner while requiring low computational power.

1. Introduction

Volunteer Computing (VC) [1] systems are large-scale heterogeneous distributed systems where resources (nodes) are donated by volunteers. Public contributors share a part of their idle computational resources to execute computationally expensive applications.

This kind of computation has become increasingly popular due to the fact that it provides a scalable, elastic, practical, and low cost platform to increase the computational and storage demands of many applications. However, the nodes are provided by users in a voluntary way, which means that they may suffer from a lack of reliability, since they are usually non-dedicated and dynamic. Therefore, the system must be able to tolerate both sudden connections and disconnections of nodes. An efficient mechanism to select which nodes will run a job or store some data is of high importance for two main reasons: (a) it is necessary to guarantee the fulfillment of the task or the availability of the data; and (b) it is recommended to minimize the quantity of nodes required for it.

Regarding to this second aspect, it is important to minimize the number of replicas involved to provide the service, specially from the storage point of view: each time a node fails a new node must be selected and all data must be replicated into it. In a VC environment with not enough highly available nodes, the selection mechanism should be able to combine nodes with different availability levels to guarantee that the system provides a good quality of service (QoS). In addition, this mechanism should be fast in order to quickly react to changes in the system.

In this paper we propose the *Multi Criteria Biased Randomized* (MCBR) method, a novel selection strategy for large-scale systems composed of unreliable nodes. MCBR allows to select the most suitable nodes in an efficient and fast way, ensuring a minimum QoS to the users. The proposed method is based on ideas of the Lexicographic Ordering (LO) multicriteria optimization strategy [2]. Thus, MCBR is a hierarchical method in which the intrinsic properties of the nodes are categorized according to different priority

[*] Corresponding author.
E-mail addresses: jpanaderom@uoc.edu (J. Panadero), jesica.dearmas@upf.edu (J. de Armas), xserraa@uoc.edu (X. Serra), jmarquesp@uoc.edu (J.M. Marquès).

https://doi.org/10.1016/j.future.2017.11.039
0167-739X/© 2017 Elsevier B.V. All rights reserved.

levels. Then, a sequence of decisions is made following the previous established priority order. Biased randomization techniques [3] are used to distribute and balance the load of the nodes. The proposed method provides high quality solutions in a very fast way, since it does not require costly computations in runtime. Moreover, due to the flexibility of this method to prioritize the properties of the nodes, it can be applied to a wide range of large-scale distributed systems other than VC, as could be P2P or Grid Systems.

We tested MCBR by simulating a real large-scale social network called Garlanet [4], that stores all data in computers voluntarily contributed by its participants. More precisely, for each user, Garlanet deploys a set of replicated microservices (in the voluntarily contributed nodes) that are in charge of guaranteeing the availability of the data. The MCBR method is used to select which node will allocate each replica of each microservice.

To validate and quantify the quality of MCBR, we have developed a metaheuristic [5]. Metaheuristic algorithms are widely recognized as efficient approaches for many optimization problems. They focus on exploring the search space to obtain optimal or quasi-optimal solutions in a reasonably short time. The metaheuristic developed in this work allows us to compare the evolution experimented by the system when applying MCBR in real time, with the results of a near optimal selection of nodes obtained with it. The experimental validation proves that the MCBR method provides high quality solutions, ensuring the minimum QoS and avoiding the excess of data movement. This last point is crucial when selecting a solving method for this kind of systems.

The remainder of this paper is structured as follows: Section 2 presents a literature review on similar approaches. Section 3 is devoted to describe the proposed MCBR method. Then, Section 4 presents the prediction quality model needed by the MCBR method. In Section 5, the metaheuristic used to compare our results is described. Section 6 presents a complete set of experiments and analyses the results. Finally, Section 7 concludes this work and proposes possible future research lines.

2. Related work

Several recent works in the literature have focus their attention on the selection of resources in distributed large-scale systems based on heterogeneous and non-dedicated components, due to the importance of making an efficient use of the resources in them. Thus, next sections are devoted to go through the main works about it. As mentioned in each section, none of these works solves the particular problem at hand.

2.1. Resource allocation in VC systems

Since the efficient resource allocation is a key factor in VC systems, several authors have worked on this research line. Estrada et al. [6] propose a distributed evolutionary genetic algorithm to design scheduling policies in VC, which maximize the throughput of the system. The proposed algorithm automatically generates scheduling policies that increase throughput across a variety of different VC projects, in contrast to the manually-designed policies, which are limited to increasing throughput for single projects. The algorithm is based on searching over a wide space of possible scheduling policies, using a small subset of IF-THEN-ELSE rules, which are used to generate the most suitable policies.

Ghafarian et al. [7,8] focus on proposing a method to schedule scientific and data intensive workflows, to enhance the utilization of VC systems. The proposed method increases the percentage of workflows that meet the deadline, satisfying the QoS constraints in terms of the deadline, minimum CPU speed, and minimum RAM or hard disk requirements. The proposed workflow scheduling system partitions a workflow into sub-workflows, to minimize data dependencies among the sub-workflows.

Sebastio et al. [9] propose a distributed framework to allocate tasks in large-scale Volunteer Clouds platforms, according to different scheduling policies. The framework takes into account five different policies, which attempt to maximize the number of executed tasks and minimizing the time at which the execution ends, both for the entire task set and for each task in the set. Each policy is formalized as a mathematical optimization problem with constraints, which is solved in a distributed fashion. In order to solve the problem in a distributed way, the framework uses the Alternating Direction Method of Multipliers (ADMM) [10] to decompose the optimization problem. Then, it is distributed and independently solved by the volunteer nodes. Besides the throughput, another important point to consider by users of Volunteer Clouds platforms is the money budget. Guler et al. [11] propose various heuristics to distribute jobs, while maximizing the throughput done by the users, without violating established money budget constraints. The heuristics are based on the price of electricity consumed by the peers, considering its temporal variation during the time, and the CPU time used.

These previous approaches are focused on maximizing the throughput of the VC system under some constraints, taking into account the types of jobs/tasks to execute in the system previously. Unlike these works, our method is focus on the resources selection in dynamical real time environments, trying to quickly react to changes in the system, e.g., sudden disconnections or the arrival of new users to the system.

2.2. Resource allocation in distributed social networks and applications

Due to the increasing popularity of social networks, other works have focused on the assignment of resources in Online Distributed Social Networks, which run over large-scale distributed systems. Thuan et al. [12] propose three heuristic algorithms for solving the client–server assignment problem in online social network applications. The algorithms are based on the user communication patterns. The authors objective is to find an approximately optimal client–server assignment that results in small total communication load, while maintaining a certain level of load balance.

Zhang et al. [13] propose three heuristics to assign clients to servers in continuous Distributed Interactive Applications (DIA) [14]. The heuristics are focused on reducing the network latency for maximizing the interactivity under consistency and fairness requirements. They are based on analyzing the minimum achievable interaction time for DIA's to preserve consistency and provide fairness among clients. Zheng et al. [15] add a complementary study to the previous work. Authors present two efficient server placement algorithms for hosting continuous DIA's. These algorithms are addressed to find optimum locations of servers in the network, with the goal of optimizing the interactivity performance, while maintaining the consistency and fairness of DIA's. The proposed algorithms take into account the interaction between clients, considering their path in the network and the latency, to produce near-optimal server placements.

Hiroshi et al. [16] present a heuristic algorithm via relaxed convex optimization, that takes a given communication pattern among the clients, providing an approximately optimal client–server assignment for a pre-specified trade-off between load balance and communication. This heuristic can be used in distributed applications such as Instant Messaging Systems (IMS).

The proposed methods in these works are based on profiling the user behaviors (i.e. obtaining information about the user communication patterns), to find optimal client–server assignments in large-scale distributed systems. However, the MCBR method does not need to gather user behavior to make optimal assignments. All the information needed is obtained from the nodes that compose the distributed system.