Future Generation Computer Systems 70 (2017) 4–16

Contents lists available at ScienceDirect

Future Generation Computer Systems

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

Power efficient server consolidation for Cloud data center[☆]

Somnath Mazumdar, Marco Pranzo[*]

Dipartimento di Ingegneria dell'Informazione e delle Scienze Matematiche Università di Siena, Siena, Italy

HIGHLIGHTS

- Propose a MILP mathematical formulation for the Server Consolidation Problem.
- Show that it can find optimal/near optimal allocation of VMs in short computing times.
- Truncated runs of the model improve over Best-Fit up to 15%.
- Use a LB provided by the MILP to compute the absolute theoretical improvement.

ARTICLE INFO

Article history:
Received 8 February 2016
Received in revised form
16 November 2016
Accepted 18 December 2016
Available online 23 December 2016

Keywords:
Cloud
Optimisation
Server consolidation
Virtual Machine allocation
Integer linear programming

ABSTRACT

Cloud computing has become an essential part of the global digital economy due to its extensibility, flexibility and reduced costs of operations. Nowadays, data centers (DCs) contain thousands of different machines running a huge number of diverse applications over an extended period. Resource management in Cloud is an open issue since an efficient resource allocation can reduce the infrastructure running cost. In this paper, we propose a snapshot-based solution for server consolidation problem from Cloud infrastructure provider (CIP) perspective. Our proposed mathematical formulation aims at reducing power cost by employing efficient server consolidation, and also considering the issues such as (i) mapping incoming and failing virtual machines (VMs), (ii) reducing a total number of VM migrations and (iii) consolidating running server workloads. We also compare the performance of our proposed model to the well-known Best Fit heuristics and its extension to include server consolidation via VM migration denoted as Best Fit with Consolidation (BFC). Our proposed mathematical formulation allows us to measure the solution quality in absolute terms, and it can also be applicable in practice. In our simulations, we show that relevant improvements (from 6% to 15%) over the widely adopted Best Fit algorithm achieved in a reasonable computing time.

1. Introduction

Cloud computing is a resource delegation model via web services by which scalable and elastically priced computation capabilities are delivered to users. The reasons behind the success of Cloud are to sell inexpensive, scalable, energy efficient [1,2] and reliable computing resources without requiring users to host them on premise. To guarantee the quality of service (QoS) to users, CIPs try to fulfil service level objectives (SLO).

A major challenge for CIPs is to respect SLO while maintaining their operational profits. DCs pay 40%–50% of their total expenses as power bills [3] mainly for running servers on the 24 × 7 basis. In fact, the power consumption of a server increases linearly as the workload increases [4]. However, idle servers can consume about 60% of their peak power [5,4,6]. Hence, idle servers should be reduced, and more workloads should be packed into running servers. An efficient server consolidation policy can significantly increase the resource utilisation (reducing both under-utilisation and over-utilisation) [7,8] which can also improve the workload imbalance while minimising the power consumption [9,10].

A *workload* is a collection of user tasks running in a server, and each workload consumes computing resources (processor, memory, I/Os) at a variable rate over the time. VMs are the primary computing blocks in Cloud. VMs allow servers to run efficiently different workloads belonging to different users and also allow to achieve economy of scale. VM management is a dynamic problem as at times new VMs arrive into the system, and similarly, old VMs get terminated. All these considerations make *VM allocation problem* (VMAP) a complex and critical allocation problem. On the other side, *server consolidation problem* (SCP)

[☆] Both authors contributed equally.

[*] Corresponding author.
E-mail addresses: mazumdar@dii.unisi.it (S. Mazumdar), pranzo@dii.unisi.it (M. Pranzo).

http://dx.doi.org/10.1016/j.future.2016.12.022

extends the VMAP to take into account server consolidation via VM migrations to improve some indicators (such as reducing power cost, server loads). Although VM migration provides the flexibility of workload management [11], it has a cost and suffers from degraded performance [12,13].

The SCP and the VMAP problems are highly dynamic in nature [14,15] and are not easy to solve at optimality. The way to address such dynamic problems is to resort to *online optimisation algorithms* [16], i.e., algorithms that take immediate decisions (assigning or migrating VMs) at a given time without knowing the future events (like incoming new VMs or terminating old VMs). An alternative approach to tackling such dynamic problems is to formulate them as offline problems, where all the events happening in a time period T define a problem instance (in our case a *snapshot*). Finally, decisions are taken by the optimisation algorithm at the end of the time period (T). Since VM creation can take several minutes, a snapshot-based approach can be used in (near) real-time by the system managers. In fact, lists of incoming VMs are the input to find an optimal or sub-optimal mapping for the given snapshot. The servers generated traffic in DCs stays within a rack [17] and also most of the network traffic is restricted to within a rack [18]. For large DCs where server counts are huge such snapshot-based approach could also run in parallel.

From CIP perspective, two of the main assumptions in our study are (i) we do not know how long a VM will run, and (ii) we do not know what will be its load profile on the resources over the time. Hence, in our approach VMs are provisioned based on the *worst case* scenario (i.e., the case in which VM consumes all the allocated resources for all the time). In this paper, we formulate a (near) real-time SCP as a mixed integer linear programming (MILP) problem. We compare our approach with widely used heuristics such as Best Fit [19] and a modified Best Fit (BFC) to take into account VM migrations. We use a commercial MILP solver with different time limits to solve the mathematical model and compare the results with the optimal solution or the bound on the solution. The rationale of our approach is to use the lower bound computed by the MILP solver to calculate the relative error. Thus, assessing in absolute terms the quality of the solutions provided by different algorithms. Therefore, the main contributions of our work are:

- To show our snapshot based formulation can find the optimal or near-optimal allocation of VMs for a variety of instances in reasonable computation times.
- To show, by comparing with the lower bound provided by the MILP solver, there exists room for improvement over the solutions provided by the widely adopted Best Fit heuristic.

The paper is organised as follows. Section 2 reviews the literature on the topic, Section 3 formally introduces the VMAP and the SCP while in the following section, we introduce the mathematical formulation for SCP. In Section 5, we present and discuss the results of our thorough simulation campaign. Finally, conclusion and future research conclude the paper.

2. Related work

Workload management in Cloud has been studied extensively with the objective of reducing energy consumption. To this aim, several approaches have been developed ranging from software based techniques (as VM migrations, server consolidations) to hardware based solutions (such as Dynamic Voltage Frequency Scaling). In this paper, we focus on software-oriented optimisations and, in particular, on effective provisioning and scheduling techniques for VM management. The rationale is that optimising resource usage by these techniques can help to minimise energy consumption by efficiently handling workloads within given constraints (e.g., QoS or power budget constraints). We refer readers to [20] for a survey of resource allocation in distributed systems and [21] for a detailed study on energy-efficient resource allocation in Cloud. In this section, we briefly review those works which are mainly aiming to perform efficiently server consolidation using workload placements and migrations and in turn reduce power consumptions.

A wide range of approaches have been proposed to tackle the problem of server consolidation ranging from constraint satisfaction problem [22,23], control theory [24], game theory [25–27], genetic algorithms [28,29], queuing model [30,31], multiple-criteria decision analysis [32] to some well-regarded heuristics based solutions ([33–37] among others). However, in the following paragraphs, we briefly review solution techniques that are relevant to our work such as *Any Fit* algorithms, mathematical programming formulations, and some heuristics based approaches.

Any Fit algorithms such as Best Fit and First Fit are well studied for achieving near-optimal solutions for Bin-packing problems. Both the heuristics are used to find solutions to the workload assignments and server consolidation. Beloglazov et al. [38] apply a modified Best Fit decreasing algorithm for VM allocation and also proposed a mechanism for minimising VM migration triggered by adaptive utilisation thresholds. Here, VMs are mapped based on their energy efficiency to avoid SLA violation. However, this approach does not include the use of memory and networks. Similarly, Sercon [39] is a server consolidation algorithm (inheriting some features from both First Fit and Best Fit) which tries to minimise the number of servers and the VM migrations. However, power cost is not considered for migration, since only CPU and memory utilisation are considered. Verma et al. [40] use a First Fit decreasing based algorithm as application placement controller to maximise performance while minimising the overall power consumption of all servers. Similar to our work, it executes in a time window and uses a linear power model. In this work, neither RAM nor Network utilisation has been considered for taking decisions. In another work [41], First Fit approximation based algorithm combined with forecasting technique is used to control VM migrations while reducing the number of running servers and SLA violations. However, this approach is mainly focused on the CPU utilisation.

Several mathematical programming based approaches have also been developed in the literature. In [42] the authors present a case study of assignment algorithms using integer programming based and also genetic algorithm based for allocating enterprise applications to servers. Both algorithms consolidate servers after analysing applications resource demand (mainly CPUs). Similarly, a MILP has been proposed in [43] for inter-and-intra DC VM placement strategy for jointly optimising the energy versus network delay. The proposed approach minimises the transport energy and the DC power consumption by periodically running the MILP solver by virtualizing the backbone topology and allocating resources in DCs. For efficient resource allocation only CPU utilisation has been considered and, in their experiments, authors set the VM hosting capacity of each server to five VMs. Whereas in [44], binary integer programming has also been employed for reducing total execution cost for deploying batch workloads in hybrid Cloud with strict deadlines. The cost component consists of data traffic cost, and computational cost, overall time slots in the schedule. Goudarzi et al. [45] propose an algorithm based on convex optimisation method and dynamic programming for solving SLA based semi-static VM placement problem. The proposed approach aims at minimising total operational cost (energy cost and migration costs) while penalising response time constraint violation. Stochastic integer programming has also been used to formulate the VM management to minimise the Cloud consumer's total provisioning cost [46]. In this paper, a Benders decomposition and sample average approximation algorithms