Future Generation Computer Systems 51 (2015) 1–6

Contents lists available at ScienceDirect

Future Generation Computer Systems

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

Rigorous results on the effectiveness of some heuristics for the consolidation of virtual machines in a cloud data center

Zoltán Ádám Mann

Budapest University of Technology and Economics, Hungary

HIGHLIGHTS

- The MM heuristic is proven to be a 3/2-approximation algorithm.
- The result of the MBFD heuristic can be arbitrarily far from the optimum.
- If MM and MBFD give optimal results, then their interplay is a 2-approximation algorithm.

ARTICLE INFO

Article history:
Received 10 October 2014
Received in revised form
23 February 2015
Accepted 14 April 2015
Available online 24 April 2015

Keywords:
Cloud computing
Virtual machines
VM consolidation
Approximation algorithms
Analysis of algorithms

ABSTRACT

Dynamic consolidation of virtual machines (VMs) in a cloud data center can be used to minimize power consumption. Beloglazov et al. have proposed the MM (Minimization of Migrations) heuristic for selecting the VMs to migrate from under- or over-utilized hosts, as well as the MBFD (Modified Best Fit Decreasing) heuristic for deciding the placement of the migrated VMs. According to their simulation results, these heuristics work very well in practice. In this paper, we investigate what performance guarantees can be rigorously proven for the heuristics. In particular, we establish that MM is optimal with respect to the number of selected VMs of an over-utilized host and it is a 1.5-approximation with respect to the decrease in utilization. On the other hand, we show that the result of MBFD can be arbitrarily far from the optimum. Moreover, we show that even if both MM and MBFD deliver optimal results, their combination does not necessarily result in optimal VM consolidation, but approximation results can be proven under suitable technical conditions. To the best of our knowledge, these are the first rigorously proven results on the effectiveness of also practically useful heuristic algorithms for the VM consolidation problem.

1. Introduction and previous work

In recent years, the increasing adoption of cloud computing has transformed the IT industry [1]. Large, virtualized data centers are serving the ever growing demand for computation, storage, and networking. Because of these trends, the efficient operation of data centers is increasingly important. One of the main concerns is energy consumption, because of both its costs and its environmental impact. According to a recent study, data center energy consumption is the fastest growing part of the energy consumption of the ICT ecosystem; moreover, the initial cost of purchasing the equipment for a data center is already outweighed by the cost of its ongoing electricity consumption [2].

An attractive option for saving energy in data centers is to consolidate the virtual machines (VMs) to the minimal number of physical hosts and switching the unused hosts off or at least to a less power-hungry mode of operation (e.g., sleep mode). However, too aggressive VM consolidation can lead to overloaded hosts with negative effects on the delivered quality of service (QoS), thus potentially violating the service level agreements (SLA) with the customers. Hence, VM consolidation must find the optimal balance between QoS and energy consumption [3,4].

In their recent works, Beloglazov, Buyya and Abawajy proposed a combination of two heuristics for near-optimal VM consolidation [5,6]. The first heuristic, called MM (Minimization of Migrations), selects the VMs that should be migrated from a given host. For this purpose, two thresholds are given: a lower and an upper threshold. If the utilization of a host drops below the lower threshold, then all VMs residing on that host should be removed so that the host can be switched off in order to save energy. If the utilization of the host is higher than the upper threshold, then some of the VMs residing on the host should be removed in order to avoid SLA violations. The MM heuristic selects the minimum number of VMs necessary to decrease the utilization below the upper threshold.

E-mail address: zoltan.mann@gmail.com.

http://dx.doi.org/10.1016/j.future.2015.04.004
0167-739X/© 2015 Elsevier B.V. All rights reserved.

The other heuristic, called MBFD (Modified Best Fit Decreasing), addresses the allocation of VMs to hosts. This can be used for two purposes: (i) to accommodate customer requests for new VMs and (ii) to find a new allocation for the VMs that should be migrated from under- or over-utilized hosts. This problem is similar to the much-studied bin-packing problem, for which simple greedy algorithms like First Fit (FF), First Fit Decreasing (FFD), Best Fit (BF), and Best Fit Decreasing (BFD) perform well and even have rigorously proven worst-case approximation ratios [7–10]. Accordingly, MBFD is also a greedy heuristic that iterates through the list of VMs in decreasing order of load and allocates each VM to the most power-efficient host that has sufficient free capacity to accommodate it.

Beloglazov et al. demonstrated with substantial empirical evidence that MM and MBFD perform well in practice and outperform other competing heuristics [5,6]. In this paper, our aim is to investigate whether any performance guarantees can be established rigorously for these heuristics, either in terms of optimality or approximation ratio.

The novelty of our approach lies in the rigorous analysis of worst-case effectiveness. Most previous works on the optimization of VM provisioning used heuristics and showed their effectiveness by means of simulations or other empirical techniques [11–16]. The drawback of such approaches is that, even if the proposed heuristics yield good results in the specific evaluation environment, there is no guarantee whatsoever that they will work similarly well under other circumstances (e.g., with other types of hosts and VMs, other workload characteristics etc.).

For example, Verma et al. compared four different heuristics using server trace data from a production data center [17]. From their plots it can be seen that there can be huge differences between the quality of the results found by those algorithms: in some cases, the placement delivered by the worst-performing algorithm consumes five times more power than the placement found by the best-performing algorithm. Concerning the number of SLA violations, the differences are sometimes even bigger (an order of magnitude or even more).

Another conclusion that can be drawn from the empirical results of that paper is that heuristics tend to have some critical parameters, the tuning of which may also result in large differences in algorithm effectiveness. For instance, their CBP heuristic has a so-called "correlation cutoff parameter"; different settings of this parameter may lead to power consumption values that are up to a factor of 2.5 apart. This may be a problem if workload characteristics are unknown – as is frequently the case for public Infrastructure-as-a-Service providers – because setting such parameters wrongly can lead to substantial degradation of algorithm effectiveness. Similar conclusions can be drawn from the results presented by Tomás and Tordsson, who showed the effect of data center overbooking on resource utilization and application response time [18]: beyond a – workload-dependent – overbooking threshold, application response time abruptly increases. As a consequence, if the target overbooking rate is wrongly selected, this may lead to severe SLA violations.

For these reasons, we believe that using heuristics without any performance guarantee is very dangerous for VM placement in practice.

There have also been some attempts to solve the VM consolidation problem optimally, by formulating it as a mathematical optimization problem, and solving it using off-the-shelf solvers. Such approaches included integer linear programming [19,20], pseudo-Boolean optimization [21], mixed integer non-linear programming [22] and binary integer programming [23]. With these solutions, the above problems are non-existent since the results are guaranteed to be optimal. However, all of these approaches suffer from a scalability problem that renders them unusable in practice: the runtime becomes prohibitively large for instances of even moderate size.

On the other hand, exact solutions also shed some light on the effectiveness of heuristics. A perfect illustration is given by Ribas et al. [21]. They compare a pseudo-Boolean formulation using two exact pseudo-Boolean solvers (SAT4j and Bsolo) with two heuristics (Round-Robin and First-Fit). The bigger benchmarks can be solved only by the heuristics, because the pseudo-Boolean solvers time out. However, on instances that are within the reach of the exact methods, it can be observed that the heuristics' results are sometimes very far from the optimum: in extreme cases, the cost of the result of the First-Fit heuristic is three times as high as the optimum; for Round-Robin, this ratio is even worse.

Therefore, we believe that none of the approaches presented so far in the literature are completely satisfactory: they are either fast but unreliable heuristics without any guarantee on their effectiveness, or they are exact methods that deliver optimal results but require exorbitantly long runtimes. In this paper, we suggest a new way which seems to be a good compromise: to investigate whether formal performance guarantees can be proven for practically usable heuristics. This way we can have the best of both worlds: fast execution *and* formal guarantees that the heuristics will be fairly effective even for unknown workloads or for suboptimal parameter values. We make a first step in this direction by analyzing a pair of heuristics that have been empirically found to be useful. Our results demonstrate that it is indeed possible to prove bounds on the effectiveness of practical heuristics, thus guaranteeing that they will work well in any situation. Furthermore, our analysis also makes the conditions explicit under which these results hold, thus pinpointing the limitations of the heuristics and giving insight for future work on the design of improved algorithms.

For some related problems, there has been some work on approximation algorithms. Breitgand and Epstein presented a 2-approximation algorithm for the stochastic bin packing problem under the assumption of independent normally distributed random variables [24]. Alicherry and Lakshman derived some approximation algorithms and inapproximability results for the problem of minimizing the cost of communication among VMs [25,26]. Breitgand et al. devised algorithms for profit optimization in a federated cloud and proved that, under certain conditions, the algorithm for one of the sub-problems, which is a greedy LP-rounding procedure, ensures 2-approximation [27]. However, none of these algorithms have been proven to work well in practice. Our approach is different: we analyze algorithms that we know are practically useful.

Section 2 of the paper is devoted to the MM heuristic, Section 3 to the MBFD heuristic. In both cases, we first describe the algorithms themselves, and then analyze their effectiveness. Section 4 is about the interplay of the two heuristics. Finally, Section 5 concludes the paper.

2. Analysis of the MM heuristic

We are given a host with capacity $C > 0$. There are k VMs currently allocated to this host with utilizations $0 < v_1, v_2, \ldots, v_k$. Obviously,

$$S := \sum_{i=1}^{k} v_i \le C$$

must hold. The host is considered overloaded if the total utilization is higher than a given threshold, defined as a percentage τ of the total capacity ($0 < \tau < 1$). That is, the host is overloaded if $S > \tau C$.

Let $V = \{1, 2, \ldots, k\}$ denote the set of VMs currently allocated to the overloaded host. The objective is to select a subset of V with