A Grouping Genetic Algorithm for Virtual Machine Placement in Cloud Computing

Hong Chen(✉)

State Grid Info-Telecom Great Power Science and Technology Co., Ltd., Beijing, China
hongchengrid@sina.com

Abstract. Virtual machine placement is a process of mapping virtual machines to physical machines. The optimal placement is important for improving power efficiency in a cloud computing environment. In this paper, we exploit a grouping genetic algorithm to solve the virtual machine placement problem. The goal is to efficiently obtain a set of non-dominated solutions that minimize power consumption. The proposed algorithm is tested with some instances from the related literatures. The experimental results show that the proposed algorithm is more efficient and effective than the other related algorithms.

Keywords: Virtual machine placement · Grouping genetic algorithm · Power consumption

1 Introduction

In recent year, cloud computing has become a popular computing paradigm for hosting and delivering services over the Internet [1]. To the consumer, the cloud appears to be infinite, and the consumer can purchase as much or as little computing power as they need. From a provider's perspective, the key issue is to maximize profits by minimizing the operational costs. In this regard, power management in cloud data centers is becoming a crucial issue since it dominates the operational costs. Moreover, power consumption in large-scale computer systems like clouds also raises many other serious issues including carbon dioxide and system reliability.

The emergence of cloud computing has made a tremendous impact on the information technology (IT) industry over the past few years, where large companies such as Amazon, Google, Salesforce, IBM, Microsoft, and Oracle have begun to establish new data centers for hosting cloud computing applications in various locations around the world to provide redundancy and ensure reliability in case of site failures. There are a number of key technologies that make cloud computing possible. One of the most importance is virtualization. Virtualization provides a promising approach through which hardware resources on one or more machines can be divided through partial or complete machine simulation, time-sharing, hardware and software partitioning into multiple execution environments, each of which can act as a complete system. Virtualization enables dynamic sharing of physical resources in cloud computing environments, allowing multiple applications to run in different performance-isolated platforms called

S. Wang and A. Zhou (Eds.): CollaborateCom 2016, LNICST 201, pp. 468–473, 2017.
DOI: 10.1007/978-3-319-59288-6_43

virtual machines (VMs) in a single physical machine (PM). This technology also enables on-demand or utility computing-a just-in time resource provisioning model in which computing resources such as CPU, memory, and disk space are made available to applications only as needed and not allocated statically based on the peak workload demand [2]. Through virtualization, a cloud provider can ensure the quality of service (QoS) delivered to the users while achieving a high server utilization and energy efficiency.

VM placement is a process of mapping VMs to PMs. As virtualization is a core technology of cloud computing, the problem of VM placement has become a hot topic recently. This VM placement is an important approach for improving power efficiency and resource utilization in cloud infrastructures. Several research works [3, 4] addressed the importance of placing VMs appropriately. Vogels [5] quoted the benefit of packing VMs efficiently in server consolidation. The proxy placement [6] and object placement/replacement [7] for transparent data replication bear some resemblance to the issues we face since they all attempt to exploit the flexibility available in determining proper placement. The following are some of the approaches that have been used to solve the VM placement problem. Chaisiri et al. [8] presented a nice algorithm for optimal placement of VMs on PMs. The goal is that the number of used nodes is minimum. They provided approaches based on linear and quadratic programming. Mi et al. [9] proposed a GA based approach, namely GABA, to adaptively self-reconfigure the VMs in cloud data centers consisting of heterogeneous nodes. GABA can efficiently decide the optimal physical locations of VMs according to time-varying requirements and the dynamic environmental conditions. Hermenier et al. [10] proposed the Entropy resource manager for homogeneous clusters, which performs dynamic consolidation based on constraint programming and takes into account both the problem of allocating the VMs to the available nodes and the problem of how to migrate the VMs to these nodes.

The scenario considered is a virtualized cloud data center that provides a shared hosting infrastructure to customers, who need resources as cloud services on a virtualized platform. Each cloud service runs inside of its own VM which can be provisioned and managed on-demand. The data center manager must respond to various on-demand resource requests by determining where VMs are placed and how the resources are allocated to them. This is a time-consuming complex task that cannot be performed by human operators in a timely fashion in increasingly larger data centers. In this paper, we exploit a grouping genetic algorithm (GGA) to solve the VM placement problem. The goal is to efficiently obtain a set of non-dominated solutions that minimize power consumption. The proposed algorithm is tested with some instances from the related literatures. The experimental results show that the proposed algorithm is more efficient and effective than the other related algorithms.

The remainder of this paper is organized as follows. In Sect. 2, we introduce a VMP optimization problem, and present a simple procedure to perform VM placement in a virtualized cloud environment. The computational results on benchmark problems are given in Sect. 3. We conclude this paper in Sect. 4.

2 VMP Optimization

In this section, we will exploit GGA to solve the VMP discrete optimization problem.

2.1 Power Consumption Model

The power consumption of PMs in cloud data centers depends on the comprehensive utilization of the CPU, memory, disk storage and network interfaces. Some studies [11] have shown that on average an idle PM consumes approximately 70% of the power consumed when it is fully utilized [12] and a linearly relationship between the power consumption and CPU utilization. Therefore, we defined the power consumption model of the *j-th* PM as follows.

$$P(u_j^{CPU}) = k * P_{max} + (1 - k) * P_{max} * u_j^{CPU} \tag{1}$$

where P_{max} is the maximum power consumed when the *j-th* PM is fully utilized; k is the fraction of power consumed by an idle PM; and u_j^{CPU} is the CPU utilization, which is a function of the time: $u_j^{CPU}(t)$ ($u_j^{CPU}(t) \in [0, 1]$). Therefore, the total power consumption by the *j-th* PM (E_j) can be modeled as as follows.

$$E_j = \int_{t_0}^{t_1} P(u_j^{CPU}(t))dt \tag{2}$$

where $P(u_j^{CPU}(t))$ is the power consumption of this PM at time t [13].

Next, an optimization formulation is proposed to minimize the power consumption.

2.2 Optimization Formulation

In this section, the optimization objective of the VM placement optimization problem to minimize the overall power consumption. The multi-objective optimization problem can therefore be formulated as:

$$\text{Minimize:} \qquad \sum\nolimits_{j=1}^{N} \sum\nolimits_{i=1}^{M} E_j x_{ij} \tag{3}$$

Such that

$$\sum\nolimits_{j=1}^{N} x_{ij} = 1,\ x_{ij} = 0\ \ or\ \ 1 \tag{4}$$

$$\sum\nolimits_{i=1}^{M} R_i^{mem} x_{ij} < T_j^{mem} \cap \sum\nolimits_{i=1}^{M} R_i^{cpu} x_{ij} < T_j^{cpu} \cap \sum\nolimits_{i=1}^{M} R_i^{bw} x_{ij} < T_j^{bw} \tag{5}$$

$$y_j, x_{ij} \in \{0, 1\} \quad i = [1, 2, \ldots, M], j = [1, 2, \ldots N] \tag{6}$$

where N is the number of PMs in the cloud data center; M is the number of VMs in the cloud data center; Eqs. (4) and (6) define the range of the variables y_j and x_{ij} and show that a VM can only be placed on one PM, such that $x_{ij} = 1$ if *i-th* VM is run on the *j-th* PM, and $x_{ij} = 0$ otherwise; Eq. (5) shows that the sum of the resource requirements for VMs must be less than the PM's idle resource capacity.

Next, we will exploit an adaptive heuristic algorithm based on the improved GGA to solve the optimization problems.

2.3 Grouping Genetic Algorithm

Since a classic GA performs poorly on grouping problems (e.g., bin-packing) which is to group a set of items into a collection of mutually disjoint subsets, the GA is heavily modified into the GGA and to suit the structure of grouping problems [15]. In the GGA, a special encoding scheme is exploited to make the relevant structure of grouping correspond to genes in chromosomes. Meanwhile, crossover and mutation operators are redefined to suit the structure of chromosomes.

Encoding: Since the objective function is defined over groups rather than isolated objects, the encoding in GGA is group oriented.

Crossover: In order to produce the offspring out of two parents, this operator inherit as much as possible of useful information from both parents. Specially, it randomly selects a portion of the first parent (i.e., some of the VM groups) and injects it into the second parent. In this way, some VMs could be contained by the groups eliminated in the second parent, this is because that they could appear twice in the solution. Therefore, GGA exploits a local heuristic (e.g., first-fit) to reinsert these missing VMs.

Mutation: Since the operator is also group oriented, it randomly select and eliminate a few VM groups. Finally, it inserts the VMs in those groups back in a random order using first-fit heuristic algorithm.

3 Experiment

In this section, we compare it with four other approaches in terms of the overall power consumption. We use CloudSim to simulate our experimental environment. In our experiment [3], we simulate a cloud data center comprising 1024 heterogeneous PMs and 2000 heterogeneous VMs. To assess the performance of our proposed algorithm (VMPGGA), we compare our algorithm with four other heuristic algorithms: Random First-fit (RFF), First-fit (FF), Best-fit algorithm (BF), and MBFD [13].

As depicted in the Fig. 1, the experiment aims at estimating the overall power consumption incurred due to the PMs used and the VMs hosted by these PMs. The experimental results indicate that our proposed algorithm (VMPGGA) has the least amount of the overall power consumption compared to other four related algorithms. This is because that our algorithm exploits the improved GGA-based approximation algorithm to search the near-optimal PMs for these VMs allocated. Therefore, when these VMs are allocated to some near-optimal PMs, the overall power consumption is minimum. Other related algorithms do not adopt heuristic algorithm. RFF, FF, and BF have similar higher overall power consumption, since these algorithms do not consider the power consumption during the allocation of these VMs.

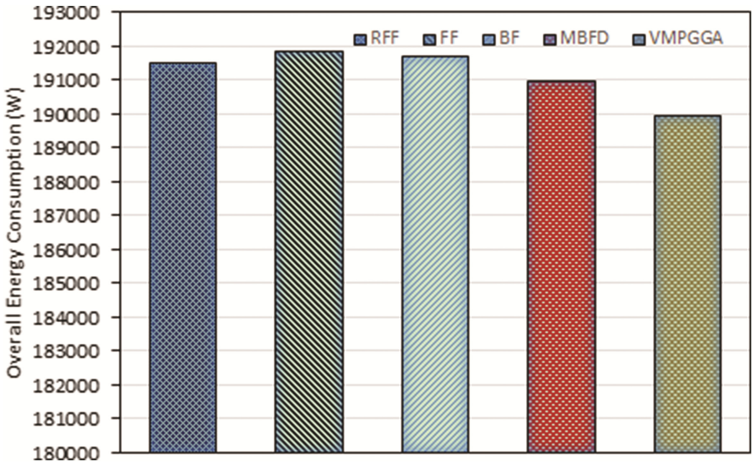

Fig. 1. Comparison of overall power consumption

4 Conclusions

In this paper, the problem of VM placement is formulated as an optimization problem aiming to optimize possibly the objective. A modified GA is exploited to effectively deal with the potential large solution space for large-scale data centers. The simulation-based experimental results showed the superior performance of the proposed approach compared with well-known bin-packing algorithms. Future work focus on implement our approach for green cloud data center [12–14].

References

1. Wang, S., Zhou, A., Hsu, C., Xiao, X., Yang, F.: Provision of data-intensive services through Energy- and QoS-aware virtual machine placement in national cloud data centers. IEEE Trans. Emerg. Top. Comput. **4**(2), 290–300 (2016)
2. Wang, S., Sun, Q., Zou, H., Yang, F.: Towards an accurate evaluation of quality of cloud service in service-oriented cloud computing. J. Intell. Manuf. **25**(2), 283–291 (2014)
3. Liu, J., Wang, S., Zhou, A., Kumar, S.A.P., Yang, F., Buyya, R.: Using proactive fault-tolerance approach to enhance cloud service reliability. IEEE Trans. Cloud Comput. PP(99), 1–1 (2016). doi:10.1109/TCC.2016.2567392
4. Grit, L., Irwin, D., Yumerefendi, A., Chase, J.: Virtual machine hosting for networked clusters: Building the foundations for autonomic orchestration. In: Proceedings of the 2nd International Workshop on Virtualization Technology in Distributed Computing, p. 7 (2006)
5. Vogels, W.: Beyond server consolidation. Queue **6**(1), 20–26 (2008)
6. Keqiu, L., Hong, S.: Optimal proxy placement for coordinated en-route transcoding proxy caching. IEICE Trans. Inf. Syst. **87**(12), 2689–2696 (2004)

7. Li, K., Shen, H., Chin, F.Y., Zheng, S.Q.: Optimal methods for coordinated enroute web caching for tree networks. ACM Trans. Internet Technol. **5**(3), 480–507 (2005)
8. Chaisiri, S., Lee, B.-S., Niyato, D.: Optimal virtual machine placement across multiple cloud providers. In: IEEE Asia-Pacific Services Computing Conference, pp. 103–110 (2009)
9. Mi, H., Wang, H., Yin, G., Zhou, Y., Shi, D., Yuan, L.: Online self-reconfiguration with performance guarantee for energy-efficient large-scale cloud computing data centers. In: IEEE International Conference on Services Computing, pp. 514–521 (2010)
10. Hermenier, F., Lorca, X., Menaud, J.-M., Muller, G., Lawall, J.: Entropy: A consolidation manager for clusters. In: Proceedings of the 2009 ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environments, pp. 41–50 (2009)
11. Fan, X., Weber, W.-D., Barroso, L.A.: Power provisioning for a warehouse-sized computer. ACM SIGARCH Comput. Archit. News **35**(2), 13–23 (2007)
12. Beloglazov, A., Buyya, R.: Adaptive threshold-based approach for energy-efficient consolidation of virtual machines in cloud data centers. In: Proceedings of the 8th International Workshop on Middleware for Grids, Clouds and e-Science (2010)
13. Wang, S., Zhou, A., Hsu, C., Xiao, X., Yang, F.: Provision of data-intensive services through energy- and QoS-aware virtual machine placement in national cloud data centers. IEEE Trans. Emerg. Top. Comput. **4**(2), 290–300 (2016)
14. Wang, S., Zhou, A., Yang, F., Chang, R.: Towards network-aware service composition in the cloud. IEEE Trans. Cloud Comput. doi:10.1109/TCC.2016.2603504
15. Xu, J., Fortes, J.A.: Multi-objective virtual machine placement in virtualized data center environments. In: Proceedings IEEE/ACM International Conference on Green Computing and Communications (GreenCom 2010), pp. 179–188 (2010)