Optimal Allocation of Virtual Machines in Cloud Computing

Ming-Hua Lin [1], Jung-Fa Tsai [2,*], Yi-Chung Hu [3] and Tzu-Hsuan Su [4]

[1] Department of Urban Industrial Management and Marketing, University of Taipei, Taipei 11153, Taiwan; mhlin@utaipei.edu.tw

[2] Department of Business Management, National Taipei University of Technology, Taipei 10608, Taiwan

[3] Department of Business Administration, Chung Yuan Christian University, Chung-Li 32023, Taiwan; ychu@cycu.edu.tw

[4] Department of Information Technology and Management, Shih Chien University, Taipei 10462, Taiwan; a86478589@gmail.com

*   Correspondence: jftsai@ntut.edu.tw; Tel.: +886-2-2771-2171

check for
updates

Abstract: Virtualization is one of the core technologies used in cloud computing to provide services on demand for end users over the Internet. Most current research allocates virtual machines to physical machines based on CPU utilization. However, for many applications that require communication between services running on different servers, communication costs influence the overall performance. Therefore, this study focuses on the optimal allocation of virtual machines across multiple geographically dispersed data centers, with the objective of minimizing communication costs. The original problem can be constructed as a quadratic assignment problem that is a classical NP-hard combinatorial optimization problem. This study adopts an efficient deterministic optimization approach to reformulate the original problem as a mixed-integer linear program that may be solved to obtain a globally optimal solution. Since the required bandwidth matrix and communication cost matrix are symmetric, the mathematical model of virtual machine placement can be simplified. Several numerical examples drawn from the literature are solved to demonstrate the computational efficiency of the proposed method for determining the optimal virtual machine allocation in cloud computing.

Keywords: virtual machine placement; quadratic assignment problem; globally optimal solution

1. Introduction

Cloud computing has become a popular technology used to provide services on demand for end users over the Internet within the last decade. It enables users to dynamically scale up software, platforms, and hardware infrastructure services. The main advantages for enterprises of adopting cloud-computing services such as Amazon EC2 Services are flexibility, reliability, and cost savings. In addition, the rapid growth of mobile devices has encouraged enterprises to move applications to the Internet. Because of this, more and more data centers are being built to satisfy increasing demand for cloud-computing services.

The expansion of data centers has significantly increased the costs of data center operators. Indeed, the average server utilization varies dramatically over time and physical resources in data centers are seriously underutilized. In order to efficiently use physical resources in data centers, most physical servers in data centers adopt virtualization technology. Visualization enables the sharing of computer hardware by multiplexing virtual machines on the same set of hardware hosts. Based on a service level agreement (SLA), data centers may provide services for use by separate virtual machines that are

located in different physical machines. Each virtual machine maintains a specific level of resources, such as CPU, storage, bandwidth, and memory, to guarantee application performance isolation and security. Since many services only occupy a small portion of physical resources, visualization technology provides flexible resource management and reduces the wastage of resources in data centers [1–3].

Most current research allocates virtual machines to physical machines based on CPU utilization. However, for many applications requiring communication between services, communication costs influence the overall performance. Due to the shared nature of multi-tenant data centers [4], network bandwidth becomes a scarce resource shared among many applications. Generally, data centers are placed all across the world following consideration of reliability, security and user demand. For distributed data centers, communication costs are the major cost. With limited bandwidth links between data centers, bandwidth-intensive applications compete for scarce network resources and execution times become unpredictable. Therefore, the quality of service cannot be guaranteed [5].

This study focuses on the optimal virtual machine placement across multiple geographically dispersed data centers. The aim of the virtual machine placement in this study is to minimize communication costs between network services under quality of service requirements. The original problem can be constructed as a quadratic assignment problem (QAP) that is a classical NP-hard problem. Many heuristic methods have been developed to solve the QAP. However, they cannot be guaranteed to converge on a globally optimal solution, especially for large-scale problems. This study adopts transformation techniques to reformulate the original problem as a mixed-integer linear program that may be solved to obtain a globally optimal solution. Several numerical examples drawn from the literature are solved to indicate computational efficiency by comparing the proposed method with the current method for virtual machine placement problems.

When compared with existing methods, this study has the following merits:

- Global optimality guarantee: Compared with heuristic methods, the proposed approach transforms the virtual machine placement problem into a mixed-integer linear programming problem and is thus guaranteed to reach a global optimum.
- Computational efficiency enhancement: Compared with the existing deterministic method [5], the proposed method adds appropriate constraints to reduce the number of feasible solutions for enhancing computational efficiency.

The rest of this paper is organized as follows. In Section 2, we discuss relevant research on the optimal allocation of virtual machines. Section 3 introduces the proposed method for determining the placement of virtual machines across multiple data centers. Computational results are presented in Section 4. Finally, concluding remarks are made in Section 5.

2. Literature Review

Since use of server virtualization technology is an effective way to improve resource utilization and reduce energy costs in data centers, the virtual machine placement problem has received much attention from both researchers and practitioners. Virtual machine placement is the process of selecting the most suitable physical machine (PM) for running a given virtual machine (VM). Therefore, a VM placement algorithm aims to design the most appropriate mapping of VM to PM. Cloud service providers may use different VM placement approaches. Silva Filho et al. [3] have provided a complete survey on optimizing virtual machine placement and migration in cloud environments. They have discussed the problem formulations, advantages, and shortcomings of relevant research.

Many different goals for VM placement have been considered in previous works, including energy saving, cost reduction, load balancing, reduction of SLA violations, network delays, congestion, and service downtime [3]. According to the goals of placement, VM placement approaches can be generally divided into two types [1]:

- Power-based approach: Determines an energy-efficient VM-PM mapping based on resource utilization [6–8].
- QoS-based approach: Determines a VM-PM mapping using the maximum fulfillment of quality of service requirements [7–9].

According to the type of optimization techniques used to obtain a VM-PM mapping, the VM placement techniques can largely be categorized into six categories [1,10]:

- Heuristic Bin Packing: The VM placement problem is formulated as vector bin packing. VMs are considered to be small items that are tightly packed into the minimum number of bins, each considered a PM. Several heuristic methods are developed to approximate the optimal solution to this packing problem.
- Biology-based optimization: Several bio-inspired optimization techniques such as the ant colony optimization method, the self-adaptive particle swarm optimization method, and genetic algorithms are applied to pack VMs into the smallest number of PMs, given the current workload.
- Linear programming: The VM placement problem is constructed as a linear programming problem which considers a number of constraints derived from practical applications. LP-relaxation-based methods are developed to solve the formulated model.
- Constraint programming: Van et al. [11] have presented a resource management framework, which includes a dynamic utility-based VM provisioning manager and a dynamic VM placement manager, to obtain a suitable VM-PM mapping. Both management tasks are regarded as constraint satisfaction problems. More practical aspects can be taken into consideration by extending the constraints in these problems.
- Stochastic integer programming: Because the future demand of VM for providing network services is uncertain, the stochastic integer programming technique is used to predict a suitable VM-PM mapping.
- Simulated annealing optimization: Liao et al. [12] have proposed a dynamic runtime mapping framework that adopts a simulated annealing optimization algorithm to map VMs onto a small set of PMs in order to minimize power consumption without significant system degradation.

Usmani & Singh [1] and Zheng et al. [10] have reviewed state-of-the-art VM placement and consolidation techniques. The challenge with the VM placement problem is how to find an optimal solution efficiently, especially when the VM placement problem is a kind of NP-hard problem. Most VM placement methods only consider CPU and memory resources and ignore networking limitations. However, communication between network services running on servers located in different data centers makes the limited network bandwidth an important issue for allocating VMs to PMs. Some research [13–15] has investigated networking needs in addition to CPU and memory needs, but most scenarios have only involved a single data center [5,16].

This study discusses the VM placement problem, with reference to Stefanello et al. [5], with consideration of multiple data centers interconnected with limited bandwidth links. We assume there are multiple users connected to different data centers who use cloud-computing services. The latency between every pair of data centers is known. To guarantee application performance for users, the maximum latency between users and the virtual machines assigned for the application must be lower than a specific level. To meet quality of service guarantees for the applications, the minimum bandwidth and maximum latency between each pair of virtual machines assigned for the applications must satisfy requirements. The objective is to derive an optimal VM placement that satisfies performance requirements and minimizes communication costs. The notations used throughout this study, with reference to Stefanello et al. [5], are as follows.

Parameters:

N: set of data centers;
K: set of virtual machines;

U: set of users;

a_i: capacity in number of VMs that data center i can host;

B_{ij}: bandwidth between data centers i and j;

L_{ij}: latency between data centers i and j;

C_{ij}: cost of transferring a unit of data between data centers i and j;

b_{vw}: required bandwidth between VMs v and w;

l_{vw}: required latency between VMs v and w;

t_{vu}: required latency between user u and VM v;

$d(u)$: the data center which hosts user u;

c_{iv}: cost of allocating VM v in data center i;

z: scaling cost factor.

Decision variables:

x_{iv}:$x_{iv} = 1$ if VM v is allocated to data center i, and otherwise $x_{iv} = 0$.

The VM placement problem, with reference to Stefanello et al. [5], can be formulated as follows.
Minimize

$$\sum_{i \in N} \sum_{v \in K} c_{iv} x_{iv} + z \sum_{i \in N} \sum_{j \in N} \sum_{v \in K} \sum_{w \in K} x_{iv} x_{jw} C_{ij} b_{vw}, \tag{1}$$

subject to:

$$\sum_{v \in K} x_{iv} \leq a_i, \forall i \in N, \tag{2}$$

$$\sum_{i \in N} x_{iv} = 1, \forall v \in K, \tag{3}$$

$$\sum_{v \in K} \sum_{w \in K} x_{iv} x_{jw} b_{vw} \leq B_{ij}, \forall i, j \in N, \tag{4}$$

$$\sum_{i \in N} \sum_{j \in N} x_{iv} x_{jw} L_{ij} \leq l_{vw}, \forall v, w \in K, \tag{5}$$

$$\sum_{i \in N} x_{iv} L_{i,d(u)} \leq t_{vu}, \forall u \in U, \forall v \in K, \tag{6}$$

$$x_{iv} \in \{0, 1\}, \forall i \in N, \forall v \in K. \tag{7}$$

The objective function (1) consists of two parts. The first part is the initial cost and the second part is the cost of communication. The VM placement problem considered in this study aims to minimize the cost of allocating VMs to PMs and communication costs resulting from each pair of virtual machines. Constraint (2) indicates that the number of VMs allocated to a data center cannot exceed the capacity of the data center. Constraint (3) ensures that each VM is assigned to exactly one data center. Constraint (4) guarantees that the total bandwidth required by all pairs of VMs allocated to data centers i and j must be less than or equal to the given bandwidth between the data centers i and j. Constraint (5) ensures that the latency between each pair of VMs allocated to data centers i and j should be less than or equal to the given latency between data centers i and j. Constraint (6) indicates that the latency between data centers i and $d(u)$ should be less than or equal to the given latency between VM v and user u if user u is located in data center $d(u)$ and accesses network services on VM v in data center i. Constraint (7) defines the variables domain.

The above model of the VM placement problem is a quadratic assignment problem (QAP) because the objective function and constraints (4) and (5) involve bilinear terms $x_{iv} x_{jw}$ of discrete variables. Since multiple virtual machines may be assigned to a VM, the VM placement problem is, more precisely, a generalized quadratic assignment problem (GQAP), which has been proven to be a definitively NP-hard problem [17]. Additionally, the complexity of the GQAP is significantly higher than that of the QAP [17].

Letting $y_{ivjw} = x_{iv} x_{jw}$, $\forall i, j \in N$, $v, w \in K$, Stefanello et al. [5] proposed a mixed-integer linear mathematical model for the VM placement problem, named the LMVMP (Linear Model for the VM

Placement), with reference to the Frieze and Yadegar [18] method for the QAP and the Lee and Ma [17] method for the GQAP. The LMVMP can be formulated as follows.

LMVMP [5]:

Minimize

$$\sum_{i \in N} \sum_{v \in K} c_{iv} x_{iv} + z \sum_{i \in N} \sum_{j \in N} \sum_{v \in K} \sum_{w \in K} y_{ivjw} C_{ij} b_{vw}, \tag{8}$$

subject to:

$$\sum_{v \in K} x_{iv} \leq a_i, \forall i \in N, \tag{9}$$

$$\sum_{i \in N} x_{iv} = 1, \forall v \in K, \tag{10}$$

$$\sum_{i \in N} y_{ivjw} = x_{jw}, \forall v, w \in K, \; \forall j \in N, \tag{11}$$

$$y_{ivjw} = y_{jwiv}, \forall v, w \in K, \; \forall i, j \in N, \tag{12}$$

$$\sum_{v \in K} \sum_{w \in K} y_{ivjw} b_{vw} \leq B_{ij}, \forall i, j \in N, \tag{13}$$

$$\sum_{i \in N} \sum_{j \in N} y_{ivjw} L_{ij} \leq l_{vw}, \forall v, w \in K, \tag{14}$$

$$\sum_{i \in N} x_{iv} L_{i,d(u)} \leq t_{vu}, \forall u \in U, \; \forall v \in K, \tag{15}$$

$$x_{iv} \in \{0, 1\}, \forall i \in N, \; \forall v \in K, \tag{16}$$

$$0 \leq y_{ivjw} \leq 1, \forall i, j \in N, \; \forall v, w \in K. \tag{17}$$

In the LMVMP model, all bilinear terms $x_{iv} x_{jw}$ are replaced with y_{ivjw}. Constraints (11) and (12) define the relationship between variables x_{iv}, x_{jw} and y_{ivjw}, $\forall i, j \in N, v, w \in K$. However, the number of variables changes from $O(|N||K|)$ to $O(|N|^2 |K|^2)$ [5].

3. Proposed Method

Since the LMVMP model [5] introduces numerous variables, the combination of variables increases rapidly as the number of virtual machines and/or the number of data centers increases. This study improves the LMVMP model [5] by adding three constraints (22), (28), and (29) to reduce the number of feasible solutions for enhancing computational efficiency. The proposed model is described as follows.

Minimize

$$\sum_{i \in N} \sum_{v \in K} c_{iv} x_{iv} + z \sum_{i \in N} \sum_{j \in N} \sum_{v \in K, v \leq (|K|-1)} \sum_{w \in K, w \geq (v+1)} (C_{ij} b_{vw} + C_{ji} b_{wv}) y_{ivjw}, \tag{18}$$

subject to:

$$\sum_{v \in K} x_{iv} \leq a_i, \forall i \in N, \tag{19}$$

$$\sum_{i \in N} x_{iv} = 1, \forall v \in K, \tag{20}$$

$$\sum_{i \in N} y_{ivjw} = x_{jw}, \forall v, w \in K, \; \forall j \in N, \tag{21}$$

$$\sum_{j \in N} y_{ivjw} = x_{iv}, \forall v, w \in K, v < w \leq K, \forall j \in N, \tag{22}$$

$$\sum_{v \in K} \sum_{w \in K} y_{ivjw} b_{vw} \leq B_{ij}, \forall i, j \in N, \tag{23}$$

$$\sum_{i \in N} \sum_{j \in N} y_{ivjw} L_{ij} \leq l_{vw}, \forall v, w \in K, \tag{24}$$

$$\sum_{i \in N} x_{iv} L_{i,d(u)} \leq t_{vu}, \forall u \in U, \; \forall v \in K, \tag{25}$$

$$x_{iv} \in \{0, 1\}, \forall i \in N, \; \forall v \in K, \tag{26}$$

$$0 \leq y_{ivjw} \leq 1, \forall i, j \in N, \; \forall v, w \in K, \tag{27}$$

$$y_{iviv} = x_{iv}, \forall i \in N, \; \forall v \in K, \tag{28}$$

$$y_{ivjv} = 0, \forall i, j \in N, \; j \neq i, \forall v \in K. \tag{29}$$

Since the required bandwidth matrix and communication cost matrix are symmetric, the objective function in the model above considers half of the y_{ivjw} variables. Because each VM is assigned to exactly one data center, in constraint (22)

$$\sum_{j \in N} y_{ivjw} = \left(\sum_{j \in N} x_{jw} \right) x_{iv} = x_{iv}. \tag{30}$$

x_{iv} is a binary variable. Therefore, in constraint (28)

$$y_{iviv} = x_{iv} x_{iv} = x_{iv}. \tag{31}$$

Because each VM can be only assigned to a single data center, x_{iv} and x_{jv} are not equal to one simultaneously if $j \neq i$. In constraint (29),

$$y_{ivjv} = x_{iv} x_{jv} = 0, \quad j \neq i. \tag{32}$$

The fact that there are numerous feasible solutions in the integer programming problem is a key factor to solving performance. The proposed method uses appropriate constraints so that only a fraction of the feasible solutions actually need to be evaluated.

4. Numerical Experiments

The purpose of the numerical experiments in this section is to demonstrate the effectiveness and efficiency of the proposed method in globally solving the VM placement problem. The experiments were conducted on a PC with a 4.3 GHz Intel Core i7-4770K CPU and 16 GB memory. All reformulated models were solved using an optimization solver IBM ILOG CPLEX 12.5 with default settings to evaluate the computational efficiency of optimization methods.

In order to compare the computational efficiency of the proposed method with the LMVMP model [5], the examples drawn from Stefanello et al. [5] were solved. Table 1 lists the experimental results of solving the VM placement problems with the proposed method and the LMVMP model [5]. All instances are encoded with the name of the number of data centers, the number of virtual machines, the number of users and the percentage of overall data center occupation. The CPU time is the average running time of CPLEX to solve each instance over five runs.

Table 1. Experimental results of the LMVMP (Linear Model for the VM Placement) model [5] and the proposed method.

Case No.	Instance	Objective	CPU Time (Seconds)	
			LMVMP	**Proposed Method**
1	005_015_007_070	25,844.02	0.73	0.66
2	005_015_007_090	23,557.30	7.27	3.25
3	005_015_015_070	10,904.78	0.37	0.39
4	005_015_015_090	24,354.96	1.15	0.69
5	005_015_022_070	14,163.60	0.37	0.41
6	005_015_022_090	32,318.02	1.81	1.53
7	005_020_010_070	38,572.62	33.24	41.43
8	005_020_010_090	64,710.80	16.07	12.86
9	005_020_020_070	55,288.76	25.65	31.81
10	005_020_020_090	57,574.90	0.83	0.70
11	005_020_030_070	28,433.34	2.42	1.68
12	005_020_030_090	66,088.70	0.73	0.70
13	005_025_012_070	43,300.76	154.39	151.15
14	005_025_012_090	10,0865.02	N/A *	6877.55
15	005_025_025_070	42,890.40	10.75	13.96
16	005_025_025_090	103.791.96	N/A *	8021.09
17	005_025_037_070	97,335.12	16.24	27.81

* error due to out-of-memory.

As seen in Table 1, CPU time increased as the number of virtual machines increased, but CPU time did not obviously increase as the number of users increased. In most cases, the proposed method required less CPU time to solve the problem. Additionally, out of seventeen cases, two cases (005_025_012_090 and 005_025_025_090) formulated by the LMVMP model [5] were solved by CPLEX and reported an error due to out-of-memory. However, the proposed method could solve all test cases. Compared with the LMVMP model [5], the proposed method that reduces the number of feasible solutions requires less memory when solving the formulated models.

In order to clearly compare the results of the proposed method with those of the LMVMP model [5], CPU times for solving the VM placement problems using the two methods, with 15 VMs and 20 VMs, respectively, are illustrated in Figures 1 and 2. Figure 1 shows that the proposed method requires less CPU time for solving Cases 1, 2, 4, and 6. Figure 2 indicates that the proposed method requires less CPU time for solving Cases 8, 10, 11 and 12. Although the proposed method does not require less CPU time than the LMVMP model [5] in all cases, the proposed method outperforms the LMVMP model [5] for all cases with 90% overall data center occupation, that is, Cases 2, 4, 6, 8, 10, and 12. For the VM placement problems with 5 data centers and 25 VMs, the experimental results are not plotted in a figure because the LMVMP model [5] could not solve Cases 14 and 16 due to out-of-memory error. Hence, the out-of-memory errors shown in Table 1 indicate that the LMVMP model [5] cannot solve Cases 14 and 16, but that the proposed method can solve these two problems in an identical software and hardware environment.
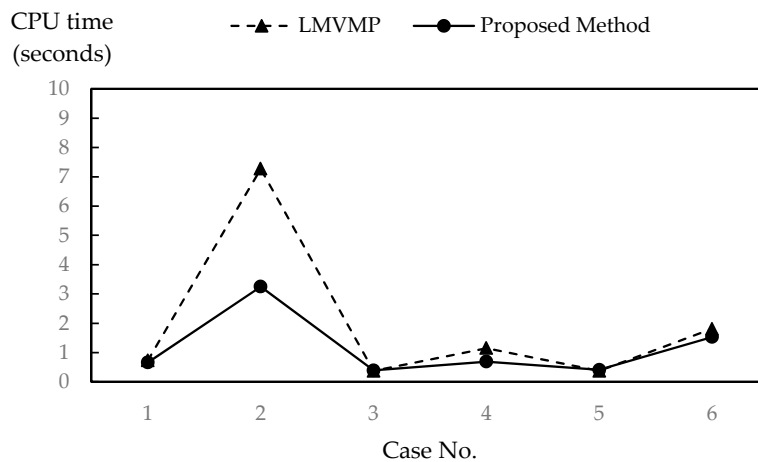

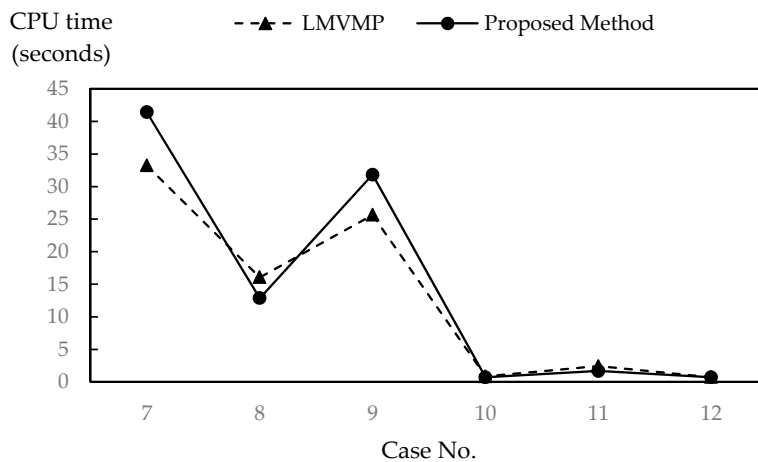**Figure 1.** Comparisons of CPU time for VM placement problems with 5 data centers and 15 VMs.

Figure 2. Comparisons of CPU time for VM placement problems with 5 data centers and 20 VMs.

5. Conclusions

Visualization technology provides an effective solution for flexible resource management within data centers. Virtual machine placement significantly affects resource utilization and quality-of-service guarantees. This study focuses on the optimal placement of virtual machines across multiple geographically dispersed data centers with minimal communication costs. The original problem is reformulated as a mixed-integer linear program which may be solved to reach a globally optimal solution. Experimental results indicate the proposed method can enhance computational efficiency in most cases compared with the current method for determining the optimal allocation of virtual machines across multiple data centers. The obtained solution suggests the optimal virtual machine placement for IT managers to use to reduce the costs of data center operators.

Experimental results indicated that CPU time for solving VM placement problems tends to grow exponentially as the number of virtual machines increases. Although the proposed deterministic approach guarantees the obtaining of a globally optimal VM placement under the specified constraints, its complexity grows rapidly as the number of virtual machines increases. More research will be required to optimize virtual machine allocation in large-scale problems, including, for instance, developing a more efficient linearization method or integrating heuristic algorithms. These issues are important to solving practical VM placement problems and are worthy of future investigation.

Author Contributions: Conceptualization, M.-H.L. and J.-F.T.; methodology, M.-H.L. and J.-F.T.; software, T.-H.S.; validation, M.-H.L. and T.-H.S.; formal analysis, M.-H.L. and J.-F.T.; investigation, M.-H.L., T.-H.S. and Y.-C.H.; data curation, T.-H.S. and Y.-C.H.; writing—original draft preparation, M.-H.L.; writing—review and editing, M.-H.L., J.-F.T. and Y.-C.H.; supervision, M.-H.L.; project administration, M.-H.L.; funding acquisition, M.-H.L. and J.-F.T.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Usmani, Z.; Singh, S. A survey of virtual machine placement techniques in a cloud data center. *Procedia Comput. Sci.* **2016**, *78*, 491–498. [CrossRef]
2. Dong, J.; Jin, X.; Wang, H.; Li, Y.; Zhang, P.; Cheng, S. Energy-saving virtual machine placement in cloud data centers. In Proceedings of the 13th IEEE/ACM International Symposium on Cluster, Cloud, and Grid Computing (CCGrid 2013), Delft, The Netherlands, 13–16 May 2013; pp. 618–624.
3. Silva Filho, M.C.; Monteiro, C.C.; Inácio, P.R.M.; Freire, M.M. Approaches for optimizing virtual machine placement and migration in cloud environments: A survey. *J. Parallel Distrib. Comput.* **2018**, *111*, 222–250. [CrossRef]
4. Greenberg, A.; Hamilton, J.R.; Jain, N.; Kandula, S.; Kim, C.; Lahiri, P.; Sengupta, S. VL2: A scalable and flexible data center network. *ACM SIGCOMM Comput. Commun. Rev.* **2009**, *39*, 51–62. [CrossRef]
5. Stefanello, F.; Buriol, L.S.; Aggarwal, V.; Resende, M.G. A new linear model for placement of virtual machines across geo-separated data centers. *Simpsio Bras. Pesqui. Oper.* **2005**, *47*, 1–11.
6. Song, W.; Xiao, Z.; Chen, Q.; Luo, H. Adaptive resource provisioning for the cloud using online bin packing. *IEEE Trans. Comput.* **2014**, *63*, 2647–2660. [CrossRef]
7. Beloglazov, A.; Buyya, R. Managing overloaded hosts for dynamic consolidation of virtual machines in cloud data centers under quality of service constraints. *IEEE Trans. Parallel Distrib. Syst.* **2013**, *24*, 1366–1379. [CrossRef]
8. Ghobaei-Arani, M.; Rahmanian, A.A.; Shamsi, M.; Rasouli-Kenari, A. A learning-based approach for virtual machine placement in cloud data centers. *Int. J. Commun. Syst.* **2018**, *31*. [CrossRef]
9. Bobroff, N.; Kochut, A.; Beaty, K. Dynamic placement of virtual machines for managing SLA violations. In Proceedings of the 10th IFIP/IEEE International Symposium on Integrated Network Management, Munich, Germany, 21–25 May 2007; pp. 119–128.

10. Zheng, Q.; Li, R.; Li, X.; Shah, N.; Zhang, J.; Tian, F.; Chao, K.M.; Li, J. Virtual machine consolidated placement based on multi-objective biogeography-based optimization. *Future Gener. Comput. Syst.* **2016**, *54*, 95–122. [CrossRef]

11. Van, H.N.; Tran, F.D.; Menaud, J.-M. Performance and power management for cloud infrastructures. In Proceedings of the IEEE 3rd International Conference on Cloud Computing (CLOUD 2010), Miami, FL, USA, 5–10 July 2010; pp. 329–336.

12. Liao, X.; Jin, H.; Liu, H. Towards a green cluster through dynamic remapping of virtual machines. *Future Gener. Comput. Syst.* **2012**, *28*, 469–477. [CrossRef]

13. Guo, C.; Lu, G.; Wang, H.J.; Yang, S.; Kong, C.; Sun, P.; Wu, W.; Zhang, Y. SecondNet: A Data Center Network Virtualization Architecture with Bandwidth Guarantees. In Proceedings of the 6th International Conference (Co-NEXT '10), Philadelphia, PA, USA, 30 November–3 December 2010. Article No. 15.

14. Ballani, H.; Costa, P.; Karagiannis, T.; Rowstron, A. Towards predictable datacenter networks. *ACM SIGCOMM Comput. Commun. Rev.* **2011**, *41*, 242–253. [CrossRef]

15. Xie, D.; Hu, Y.C. The only constant is change: Incorporating time-varying network reservations in data centers. *ACM SIGCOMM Comput. Commun. Rev.* **2012**, *42*, 199–210. [CrossRef]

16. Stefanello, F.; Aggarwal, V.; Buriol, L.S.; Gonçalves, J.F.; Resende, M.G. A biased random-key genetic algorithm for placement of virtual machines across geo-separated data centers. In Proceedings of the 2015 Conference on Genetic and Evolutionary Computation (GECCO'15), Madrid, Spain, 11–15 July 2015; pp. 919–926.

17. Lee, C.G.; Ma, Z. *The Generalized Quadratic Assignment Problem*; Technical Report; Department of Mechanical and Industrial Engineering at the University of Toronto: Toronto, ON, Canada, 2004.

18. Frieze, A.M.; Yadegar, J. On the quadratic assignment problem. *Discret. Appl. Math.* **1983**, *5*, 89–98. [CrossRef]