Precise VM Placement Algorithm Supported by Data Analytic Service

Dapeng Dong and John Herbert
Mobile and Internet Systems Laboratory, Department of Computer Science, University College Cork, Cork, Ireland

Keywords: Cloud Computing, VM Placement, Forecast, Decision Support.

Abstract: The popularity and commercial use of cloud computing has prompted an increased concern among cloud service providers for both energy efficiency and quality of service. One of the key techniques used for the efficient use of cloud server resources is virtual machine placement. This work introduces a precise VM placement algorithm for power conservation and SLA violation prevention. The mathematical model of the algorithm is supported by a sophisticated data analytic system implemented as a service. The precision of the algorithm is achieved by allowing each individual VM to build, on demand, its own data model over an appropriate time horizon. Thus the data model can reflect the characteristics of resource usage of the VM accurately. The algorithm can communicate synchronously or asynchronously with the data analytic service which is deployed as a cloud-based solution. In the experiments, several advanced data modelling and use forecasting techniques were evaluated. Results from simulation-based experiments show that the VM placement algorithm (supported by the data analytic service) can effectively reduce power consumption, the number of VM migrations, and prevent SLA violation; it also compares favourably with other heuristic algorithms.

1 INTRODUCTION

Cloud computing has gained hugely in popularity in recent years. As the utility computing paradigm requires massively server deployment, one of the main concerns for a cloud service provider is the operational cost, especially the cost of power consumption. Research indicates that servers in many organizations typically run at less than 30% of their full capacity (Barroso and Holzle, 2007) (Sargeant, 2010). Thus it is possible to reduce power consumption of the hardware by means of allocating more Virtual Machines (VMs) to less hosts. VM placement is one of the key techniques used for this purpose, and is extensively studied. The basic principle of VM placement is to allocate as many VMs on a physical server as possible, while satisfying various constraints specified as part of the system requirements. Previous work (prompted by business strategy or user preference) has focused on improving VM performance and availability(Jayasinghe et al., 2011), scalability (Jiang et al., 2012) (Biran et al., 2012) (Meng et al., 2010), energy conservation (Verma et al., 2008), SLA (Service Level Agreement) violation prevention (Beloglazov and Buyya, 2012), VM live migration cost (Clark et al., 2005) (Liu et al., 2011), or a combination (Goudarzi et al., 2012) (Xu and Fortes, 2010). In this work, both power conservation and SLA violation prevention are considered.

The commercial use of cloud computing continues to expand. An SLA is one of the main ways to deal legally with QoS guarantees, and, as such, is of concern to both consumers and service providers. Theoretically, QoS (Quality of Service) can be guaranteed through appropriate resource provisioning via prediction. Due to the dynamic and heterogeneous nature of cloud services, predictions are often inaccurate. An improved prediction accuracy is achieved in this work by allowing each individual VM to build (on demand) its own data model over an appropriate time horizon. Thus the data model can reflect the characteristics of resource usage of the individual VM accurately. The mathematical modelling of the algorithm is supported by a sophisticated data analytic system implemented as a service. More specifically, the R open source data analytic framework (R, 2012) is employed as decision support and a modelling engine. The R Decision Support System (rDSS) is designed and deployed as a cloud-based solution, providing services to the VM placement algorithm.

The rDSS system architecture along with its use in a simulation environment are described in later sections. The system is used to evaluate the proposed precise VM placement algorithm. The ex-

Dong D. and Herbert J..
Precise VM Placement Algorithm Supported by Data Analytic Service.
DOI: 10.5220/0004371904630468
In *Proceedings of the 3rd International Conference on Cloud Computing and Services Science* (CLOSER-2013), pages 463-468
ISBN: 978-989-8565-52-5

perimental results show that our forecast-based approach, supported by the data analytic service, saves $0.03 \sim 1.14kW/h$ in power consumption with up to 70% fewer VM migrations and a low rate of SLA violation (0.05% on average) over a six hour period when compared to a non-forecast based power aware best fit decreasing algorithm.

2 RELATED WORK

VM placement falls into the field of multi-objective optimization, and it is often formulated as a Bin Packing or a Constraint Programming problem. Chen et al. (Chen et al., 2011) proposed an Effective Sizing guided VM placement algorithm. The proposed Effective Sizings were calculated by computing least workload correlations with other VMs on the target host. As VM consolidation is often an NP-complete task, many researchers employ heuristic algorithms in order to provide optimal solutions in a timely fashion. A Power Aware Best Fit Decreasing (PABFD) heuristic was proposed in the study of Beloglazov et al.(Beloglazov and Buyya, 2012). More advanced heuristic algorithms were also employed by researchers. such as Genetic Algorithms (GAs) as used by Xu and Fortes (Xu and Fortes, 2010). Although GAs may provide better solutions than simple heuristic algorithms, but it's not able to provide optimal solutions in a timely fashion.

Recent research found that network communication consumes a considerable portion of energy in cloud data centres (Meng et al., 2010)(Biran et al., 2012). This is not surprising if one considers moving a number of VMs with memory footprint ranging from a few hundred MB to tens of GB. Using VM migration as a technique for VM consolidation can therefore cause both network performance and VM performance to degrade significantly (Liu et al., 2011)(Clark et al., 2005). Meng et al. (Meng et al., 2010) present a traffic-aware algorithm for VM placement. The authors formulate VM placement as a Quadratic Assignment Problem and make their solution aware of network topologies and network traffic patterns. Cloud service providers rely on forecasting for resource provisioning. Beloglazov at el.(Beloglazov and Buyya, 2012) illustrate some simple statistical and curve fitting techniques in their study. More advanced techniques have also been chosen by other researchers. In the study by Kusic at el. (Kusic et al., 2008), the authors used a trained Kalman filter to produce estimates of the number of workload requests and to forecast the future state of the system.

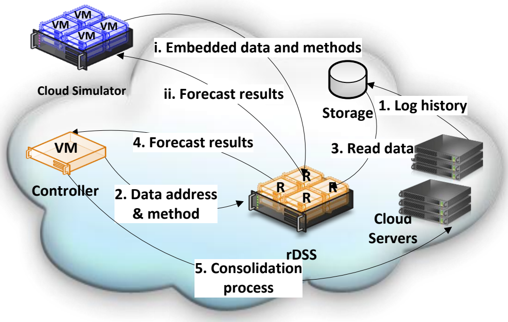

Figure 1: rDSS system architecture.

3 SYSTEM ARCHITECTURE

The rDSS system is a cloud-based solution. Resource utilization information from each VM and host is logged to a centralised cloud location (Figure 1). The data collection process can be done by a hypervisor or third party software. Data analytic servers that employ the R framework as an engine, are pre-packaged Linux images that can run on VMs. The number of data analytic servers can be scaled up on demand since it's cloud-based. A group of host machines are controlled by a Controller. The Controller has three responsibilities. Firstly, it segments historical data to a specified length for each VM or host; this segmented data will be used for building forecast models. Secondly, the Controller passes the address of data and the specific modelling algorithm to the rDSS. Information is sent programmatically by calling program functions which have embedded R language clauses. Communication can also be done asynchronously via queues. In this case, information will be sent/received as messages through queues. The asynchronous communication approach is particularly useful when the data set is large. The returned results can also be received synchronously or asynchronously. Finally, based on the forecast model and prediction results returned from rDSS, the Controller carries out a consolidation process.

4 PROBLEM FORMULATION

Given a set H of hosts, a set V of virtual machines in the cloud data centre and power consumption models for each host, the objective is to decide how to rearrange V on H such that the total power consumption in the data centre is minimized, and the SLA violation rate is kept as low as possible. All $v \in V$ requirements r_i, must be satisfied by the targeting host; all $v \in V$ predicted CPU requirements $\{\hat{r}_{i(t+n)}\}$ must be satisfied by the target host in order to minimize SLA violations; each h has a resource capacity limit C. The total power consumption is the sum of power p_{ij} consumed by CPUs of each VM i on host j, plus a fixed power f_j consumed by the other components of host j. Let $h_j = 1$ represent choosing host j to be switched on, and 0 otherwise. Also, let $v_{ij} = 1$ represent the assignment of VM i to host j, and 0 otherwise. The mathematical model is outlined as follows:

$$
\begin{array}{llll}
\min & \sum_{i\in V}\sum_{j\in H} p_{ij}v_{ij} + \sum_{j\in H} f_j h_j & & \\
\text{s.t.} & \sum_{i\in V} r_i v_{ij} \leqslant Ch_j & & \forall j \in H \\
& \sum_{i\in V} \{\hat{r}_{i(t+n)}\} v_{ij} \leqslant Ch_j & & \forall j \in H \\
& \sum_{j\in H} v_{ij} = 1 & & \forall i \in V \\
& v_{ij} \leqslant h_j & & \forall i \in V, j \in H \\
& v_{ij}, \quad h_j \in \{0,1\} & & \forall i \in V, j \in H
\end{array}
$$

One of the main causes of SLA violation is that the requested resources (from VMs) can not be satisfied by the resource providers (hosts). SLA violation minimization is done in two parts. In the first part, forecast models are built for each VM based on a certain length of historical data. The forecast model is then used to predict the future CPU requirements for each VM. The SLA is controlled as follows. Let the matrix $A \in \mathbb{R}^{+m*n}$ denote the total m VMs in the cloud; a list is associated with each VM which contains n step-ahead forecast values, the number of steps is adjusted according to the consolidation process frequency; $\hat{r}_{i(t+n)}$ denotes the predicted CPU requirement for VM i at time $t+n$. For all VMs that have been placed and/or going to be placed on host j construct a matrix A', $A'_j \subseteq A$, $i \leqslant m$.

$$
A'_j = \begin{bmatrix} \hat{r}_{0(t+1)} & \hat{r}_{0(t+2)} & \cdots & \hat{r}_{0(t+n)} \\ \vdots & \vdots & \ddots & \vdots \\ \hat{r}_{i(t+1)} & \hat{r}_{i(t+2)} & \cdots & \hat{r}_{i(t+n)} \end{bmatrix}
$$

Such that

$$
SLA(A'_j) \models \{\alpha \sum_{k=0}^{i} \hat{r}_{k(t+n)} \leqslant Ch_j, \quad \forall n\} \qquad (1)
$$

Forecasting often contains errors. The second part of SLA protection is to reserve a certain amount of resources on each host to tolerate forecast errors and accommodate sudden bursts in CPU requests indicated by α in condition 1. The resource reservation strategy is outlined as follows. If a host has less resources to offer than the required buffers, it will be seen as over utilized, then one or more VMs will be selected to migrate to other host(s). The result of VM(s) migrating to the target host(s) must not violate the conditions 1. VM migration results in VM performance degradation, extra load on the network burden, and energy cost (Liu et al., 2011)(Clark et al., 2005). The conclusion of (Beloglazov and Buyya, 2012) is that smaller VM migration time produces better results; and (Liu et al., 2011) asserts that VM live migration time is mainly determined by memory size, memory dirtying rate, and network bandwidth. Based on these conclusions, the principle of smallest memory size first is used in the selection of VMs for migration. For simplicity, it is assumed that the memory dirtying rate and network bandwidth are constants. In this work, we reserve resources on each host statically.

As the proposed VM placement algorithm relies heavily on the forecast results, an accurate forecast model is at its foundation. As VMs are continuously running in the cloud, the CPU utilization of each VM at the sampling times generates a time series. It should be noted that due to the heterogeneity of workloads, the time series of VMs often exhibit different properties, and we need an adaptive way of building forecast models without prior knowledge of the types of workloads. Employing the powerful R framework as decision support system, allows us to produce forecasts based on advanced data modelling techniques.

A prerequisite for the proposed algorithm is of establishing connections between VMs and rDSS servers. Once the connections are established, VMs remain connected during their lifetime and the algorithm maintains a map of VMs to connections. The map is used as an input to the algorithm. Every VM join/leave event will correspond to a map refreshment action. The VM placement process follows the same principle as the PABFD algorithm (Beloglazov and Buyya, 2012). The differences are that any successful placement needs to satisfy both hardware requirements (such as memory and storage) and SLA requirements; over utilized hosts are determined by examining the condition $SLA(A'_{host})$. Given the power consumption model of each host, the reason for choosing PABFD is that it allows VMs to be placed on more power efficient hosts in a heterogeneous environment.

5 EVALUATION

The simulated environment is IaaS (Infrastructure as a Service). It consists of 80 HP ProLiant ML110 G4, G5 hosts, randomly selected to build a heterogeneous cloud environment. Power consumption models of hosts were collected from (Spe, 2008). Mem-

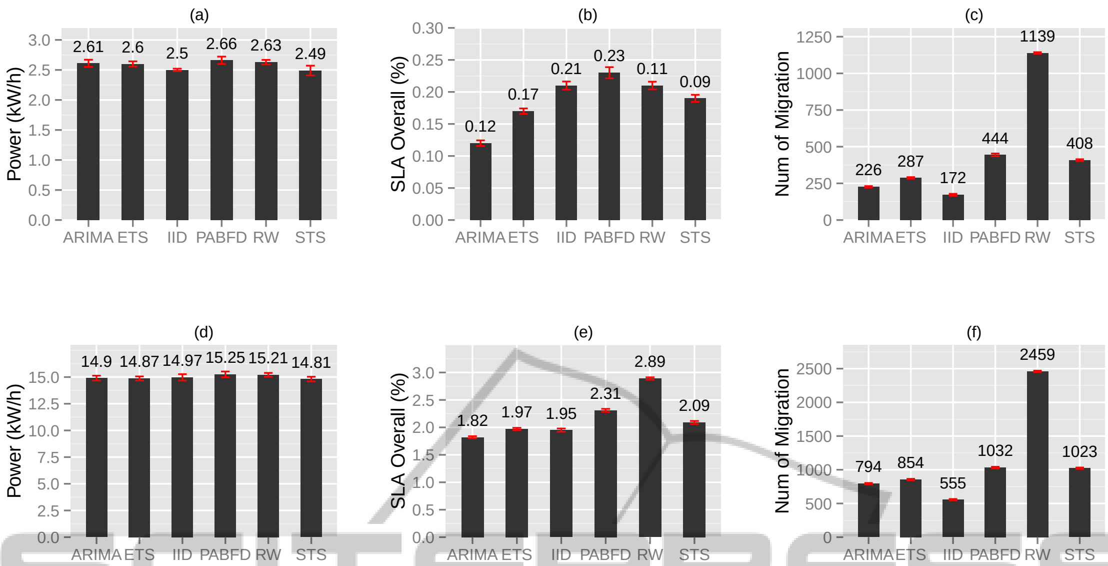

Figure 2: Experiment results from real-world server workloads and random workloads.

ory assignment for VMs are uniformly distributed in the range $256MB \sim 1GB$. A set of mixed, real-world server workloads were collected from (Beloglazov and Buyya, 2012). They were given to each VM during simulation. We use three sets of experiments to evaluate the One-step ahead Forecast-based Power Aware Best Fit Decreasing(F1PABFD) algorithm. All sets of experiments use a configuration of one hundred VMs and eighty hosts. The experiments simulated a cloud environment continuously operational for six hours, with the consolidation frequency set to five minutes.

Forecast models were built for each VM based on two hours of historical data. We also compared our algorithm with the PABFD (Local Regression and Minimum Migration Time) heuristics approaches (Beloglazov and Buyya, 2012). Figure 2(a)(b) and (c) show results from the first set of experiment. A set of mixed, real-world server workloads were given to each VM during simulation. The workloads are directly mapped to the CPU utilization of each VM. They reflect 10.74% CPU utilization of each VM on average. Figure 2(a),(b),(c) show comparison of the total power consumption, number of VM migrations, and overall SLA violation for each of the algorithms respectively. It can be observed that our F1PABFD algorithm is able to save $0.03 \sim 0.17$ electricity unit (kW/h) over the six hours operation compared to PABFD. Among the algorithms, F1PABFD-IDD produced the lowest power consumption. However, our forecast-based algorithm has a significantly reduced number of VM migrations - up to 39% of that achieved by F1PABFD-IDD. Based on accurate forecasting, our algorithm effectively prevented SLA violations by reducing overall SLA violation to the best result of 0.12%, compared with 0.23% given by non-forecast PABFD.

In the second set of experiments, we evaluated the robustness of our algorithms. We performed exactly the same experiments but with random workloads. The generated random workloads reflect approximately 53.2% CPU utilization of each VM on average. In Figure 2(d), the power consumption trends are similar to that produced in Figure 2(a). Because of the obvious reason that the workloads were heavier, the power consumption is higher. With random and heavier workloads, our algorithms start saving more energy – $0.03 \sim 0.44$ kW/h energy saving for six hours operation compared with Figure 2(a). We further prove this in the third set of experiments. In Figure 2(f), we still observe a significant drop in the number of VM migrations. In Figure 2(e) we see a significant increase of SLA violations. There are three possible reasons. 1) It's caused by randomness, because the randomness has a general negative effect on our forecast-based algorithm; 2) It's caused by the increased average CPU utilization on each VM; 3) or both. The third set of experiments aims to answer this question. For these more detailed experiments, F1PABFD-IID, -STS, and -ARIMA algorithms were selected for further evaluation, eliminating F1PABFD-RW due to having the highest number of VM migration and F1PABFD-ETS due to having lowest performance (Figure 4).

In the third set of experiments, we evaluated how the weight of workloads effect our algorithms. We generated eight sets of workloads artificially based on the real-world server workloads used in the first ex-

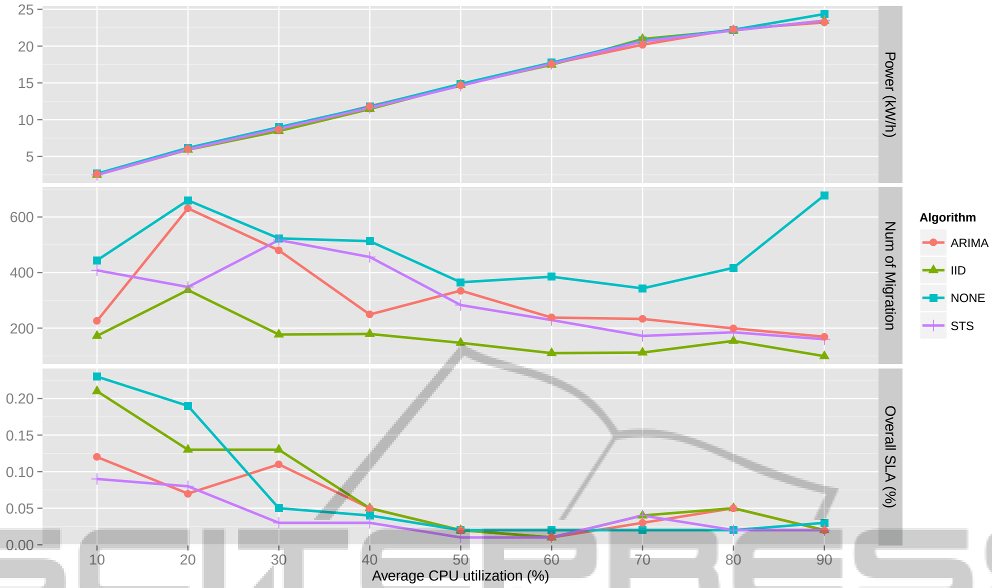

Figure 3: Experiment results from real-world workloads.

periment. To preserve the characteristics of the original workloads as much as possible, such as trends and periodicity, etc, we added a constant to each workload value; and gradually increased the weight of the average workloads from 10.74% (original) up to 90%. In Figure 3, we calculated that all F1PABFD-based algorithms consume 0.03 $\sim$ 1.14kW/h less power than the non-forecast PABFD. A significant power drop is observed at the average weight of workloads of 90%. The reason is as follows. When the average workload (average CPU utilization of each VM) is reaching full capacity, the changes in the CPU utilization curves become more smooth. Our forecast results become more accurate; VM placement decisions based on the forecast results can be more accurate, and fewer active hosts are required. Consequently power consumption is reduced. The more accurate forecasting also results in lower SLA violations. This can be observed in the end portion of Figure 3. Another observation is that SLA violation becomes higher when average workloads become lower. This is because when CPU utilization of each VM is lower, more VMs will be assigned to a host. When a host becomes more compact, it increases uncertainty of resource requirements. An improvement can be made by dynamically reserving more resources on each host according to the level of the average workloads. This is planned for future implementation. We also noticed that SLA violation didn't increase with the increasing weight of workloads. Therefore, we concluded that the observed significant increase in SLA violation in Figure 2 (e) was caused by the randomness of the workload. Overall, our F1PABFD algorithm performs much better than

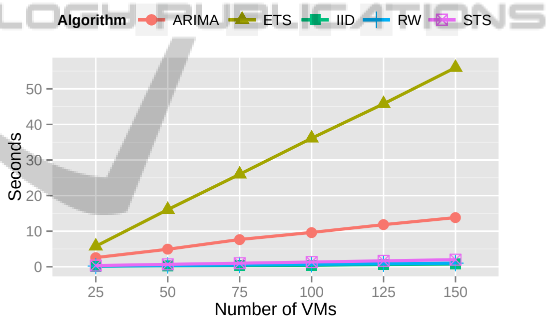

Figure 4: Algorithm performance.

the original PABFD, and especially in reducing the number of VM migrations. In the middle facet of Figure 3, we observe that F1PABFD-IID, -STS, -ARIMA algorithms can reduce the number of VM migrations by 3/1.5/1.5 times on average respectively, compared with PABFD. Among them, F1PABFD-STS, -ARIMA performed consistently well; F1PABFD-IID was best at reducing VM migrations. Considering that F1PABFD-ARIMA is much slower (Figure 4) than -STS and -IID, F1PABFD-STS and -IID are recommended. Depending on the status of the network for the cloud, F1PABFD-IID may be preferred over -STS. Therefore, there is an opportunity for researching dynamic algorithm switching based on network status.

6 CONCLUSIONS

This paper has described a forecast-based VM placement algorithm with power aware best fit decreasing heuristic algorithms for cloud server consolidation. Our main concerns are power consumption and SLA violation. We designed and deployed an architecture using cloud-based R servers as the decision support system. Forecast models were built for each VM, and predictions made as to their future CPU resource requirements. The simulation-based experiments analysed the proposed algorithms with respect to their consistency, robustness, and performance. The results demonstrate that the proposed approach, compared with other heuristics, can significantly reduce power consumption, the number of VM migrations, and number of SLA violations. The analytical model has not taken load-balancing into consideration, and this will be addressed in future work.

ACKNOWLEDGEMENTS

This work is supported by the Telecommunications Graduate Initiative (TGI) program which is funded by the Higher Education Authority under the Programme for Research in Third-Level Institutions (PTRLI) Cycle 5 and co-founded under the European Regional Development Fund (ERDF).

REFERENCES

(2000 - 2012). R porject. http://www.r-project.org.

(2008). Spec. http://www.test.org/doe/.

Barroso, L. and Holzle, U. (2007). The case for energy-proportional computing. *Computer*, 40(12):33 –37.

Beloglazov, A. and Buyya, R. (2012). Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers. *Concurrency and Computation: Practice and Experience*, 24(13):1397–1420.

Biran, O., Corradi, A., Fanelli, M., Foschini, L., Nus, A., Raz, D., and Silvera, E. (2012). A stable network-aware vm placement for cloud systems. In *Proceedings of the 2012 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (ccgrid 2012)*, pages 498 –506.

Chen, M., Zhang, H., Su, Y.-Y., Wang, X., Jiang, G., and Yoshihira, K. (2011). Effective vm sizing in virtualized data centers. In *Integrated Network Management (IM), 2011 IFIP/IEEE International Symposium on*, pages 594 –601.

Clark, C., Fraser, K., Hand, S., Hansen, J. G., Jul, E., Limpach, C., Pratt, I., and Warfield, A. (2005). Live migration of virtual machines. In *Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation - Volume 2*, NSDI'05, pages 273–286.

Goudarzi, H., Ghasemazar, M., and Pedram, M. (2012). Sla-based optimization of power and migration cost in cloud computing. In *Cluster, Cloud and Grid Computing (CCGrid), 2012 12th IEEE/ACM International Symposium on*, pages 172 –179.

Jayasinghe, D., Pu, C., Eilam, T., Steinder, M., Whally, I., and Snible, E. (2011). Improving performance and availability of services hosted on iaas clouds with structural constraint-aware virtual machine placement. In *Services Computing (SCC), 2011 IEEE International Conference on*, pages 72 –79.

Jiang, J., Lan, T., Ha, S., Chen, M., and Chiang, M. (2012). Joint vm placement and routing for data center traffic engineering. In *INFOCOM, 2012 Proceedings IEEE*, pages 2876 –2880.

Kusic, D., Kephart, J., Hanson, J., Kandasamy, N., and Jiang, G. (2008). Power and performance management of virtualized computing environments via lookahead control. In *Autonomic Computing, 2008. ICAC '08. International Conference*, pages 3 –12.

Liu, H., Xu, C.-Z., Jin, H., Gong, J., and Liao, X. (2011). Performance and energy modeling for live migration of virtual machines. In *Proceedings of the 20th international symposium on High performance distributed computing*, HPDC '11, pages 171–182. ACM.

Meng, X., Pappas, V., and Zhang, L. (2010). Improving the scalability of data center networks with traffic-aware virtual machine placement. In *INFOCOM, 2010 Proceedings IEEE*, pages 1 –9.

Sargeant, P. (2010). Data centre transformation: How mature is your it? Gartner. Inc.

Verma, A., Ahuja, P., and Neogi, A. (2008). pmapper: power and migration cost aware application placement in virtualized systems. In *Proceedings of the 9th ACM/IFIP/USENIX International Conference on Middleware*, Middleware '08, pages 243–264.

Xu, J. and Fortes, J. (2010). Multi-objective virtual machine placement in virtualized data center environments. In *Green Computing and Communications (GreenCom), 2010 IEEE/ACM Int'l Conference on Int'l Conference on Cyber, Physical and Social Computing (CPSCom)*, pages 179 –188.