GMDH based on Genetic Algorithmand MMT Policy for Energy Efficient Dynamic Consolidation of Virtual Machines

C.L. Peng, Y. Li, Z.Q. Wang, S.D. Du
School of Electronic Science and Engineering
Nanjing University, China

***Abstract*—Dynamic consolidation of virtual machines (VMs) is an effective way to improve the utilization of resources and energy efficiency in grid computing. We propose a novel load balancing approach that combines the Group Method of Data Handling (GMDH) based on Geneticalgorithmfor host overload prediction and the Minimum Migration Time (MMT) policy for VM selection. The GA-GMDH algorithm could predict the actual host load in each consecutive future time interval. We evaluate our method using the workload traces of Google Cluster data. Our proposed algorithms significantly reduce energy consumption, while ensuring a high level of adherence to the Service Level Agreements (SLAs).**

Keywords-grid computing;virtualization;dynamic consolidation; host overload prediction; group method; minimum migration time; energy efficiency

I. INTRODUCTION

The focus of this work is on energy and performance efficient resource management strategies that can be applied in a virtualized data center by a Cloud provider (e.g. Google App Engine). Effective host overload prediction is conducive to dynamic resource provisioning [1], virtual machine migration [2], server consolidation and energy management. Therefore, accurate host overload prediction is essential for load balancing. In this paper, we propose an effective host overload prediction method with comparatively less prediction errors and acceptable prediction interval length. The main idea of our approach is to use GMDH[3]method based on genetic algorithm for host overload prediction and apply Minimum Migration Time (MMT) policy [4] to the VM selection stage. We evaluate the proposed algorithms by extensive simulation using the Cloud Sim [5] toolkit and one month's worth of accounting records from the Google Cluster data.

Our main contributions are three-fold:

1. We introduce a GA-GMDH algorithm, which predict the actual host load for a future time interval rather than the mean load only.

2. We combine the GA-GMDH and MMT approaches for energy efficient dynamic consolidation of VMs in the context of Cloud Computing.

3. An extensive simulation-based evaluation and performance analysis of the proposed algorithms.

The remainder of the paper is organized as follows. In Section 2 we discuss the related work. In Sections 3 we present a thorough analysis of the VM consolidation problem. We propose our adaptive heuristics algorithms in Section 4, continuing with an evaluation and analysis of the obtained experiment results in Section 5. We discuss future research directions and conclude the paper in Section 6.

II. RELATED WORK

Many efforts [6][7][8] have been made in host load prediction in Grids or HPC systems. C. Dabrowski et al. [6] perform the host overload prediction by leveraging the Markov model via a simulated environment. S. Akioka, et al. [7] combine the Markov model and seasonal analysis to predict the host load for one-step ahead in a computational Grid. Y. Wu et al. [8] use hybrid model for multi-step ahead host overload prediction, which combines the Auto Regressive (AR) model and Kalman filter.

To predict the host load in the Cloud, B. Guenter [9] proposed a simple linear prediction scheme which predicts the host load for the next time. Q. Zhang [10] used the Auto-Regressive Integrated Moving Average (ARIMA) model to predict the host load. In [9], the ARIMA model could predict the load over a time window H by iterated the one step prediction. In [11], D. Yang et al. proposed a multi-step-ahead prediction method for CPU load.

S. Di et al. [12] firstly use the Bayesian model to predict the host load in the Cloud. Srikantaiah et al. [13] have studied the problem of request scheduling for multi-tier web applications in virtualized heterogeneous systems to minimize energy consumption, while meeting performance requirements.

III. THE VM CONSOLIDATION PROBLEM

VM consolidation is the key problem that IaaS provider or data center operators often face. They need develop appropriate resource management and scheduling strategies to meet SLAs, improve load balancing capability and reduce energy consumption. Before the VM selection stage, we need know which host is overloaded. Then the next step is to select particular VMs to migrate from this host.

We define that there are n homogeneous hosts, and the capacity of each host is A_h . Although VMs experience variable workloads, the maximum CPU capacity that can be allocated to a VMis A_v . Therefore, the maximum number of VMs allocated to a host when they demand their maximum CPU capacity is $m = \frac{A_h}{A_v}$The total number of VMs is nm. VMs can be migrated between hosts using live migration with a migration time t_m. Obviously, SLA violation occurs when the total demand for the CPU performance exceeds the available CPU capacity A_h. The cost of power is C_p, and the cost of SLA violation per unit of time is C_v. Without loss of generality, we can define $C_p = 1$ and $C_v = s$, where $s \in R^+$. We assume that when a host is idle, i.e., there are no allocated VMs, it is switched off and consumes no power, or switched to the sleep mode with negligible power consumption. We call non-idle hosts active. The total cost C is defined as follows:

$$C = \sum_{t=t_0}^{T} (C_p \sum_{i=0}^{n} a_{ti} + C_v \sum_{j=0}^{n} v_{tj}) \quad (1)$$

where t_0 is the initial time; *T* is the total time; $a_{ti} \in \{0,1\}$ indicating whether the host *i* is active at the time *t*; $v_{tj} \in \{0,1\}$ indicating whether the host *j* is experiencing an SLA violation at the time *t*. The problem is to determine what time, which VMs and where should be migrated to minimize the total cost *C*.

IV. THE ALGORITHMS FOR VM CONSOLIDATION

We split the problem of dynamic VM consolidation into four parts: (1) determining when a host is considered as being overloaded to migrate of one or more VMs from this host; (2) determining when a host is considered as being under loaded to migrate all VMs from this host and switch the host to the sleep mode; (3) selection of VMs that should be migrated from an overloaded host; and (4) finding a new placement of the VMs selected for migration from either the overloaded or under loaded hosts.

A. The Overview of GA-GMDH

The GMDH network is a feed-forward network that can be represented as a set of neurons, of which different pairs in each layer are connected through a quadratic polynomial and thereby produce new neurons in the next layer. The coefficients of the neuron are estimated using the Least Squares Method. The most popular base function used in GMDH is the gradually complicated Kolmogorov-Gabor polynomial:

$$\hat{y} = a_0 + \sum_{i=1}^{n} a_i x_i + \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij} x_i x_j + \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n} a_{ijk} x_i x_j x_k + \cdots \quad (2)$$

where n is the number of the data in the dataset; $A = (a_0; a_1; a_2; \ldots)$ and $X = (x_i; x_j; x_k; \ldots)$ are the vectors of the coefficients and input variables of the multi-input single-output system; and $\hat{y}$ is the output of an individual host. However, in the GMDH algorithm, the infinite Kolmogorov-Gabor polynomial is estimated by a cascade of a second order polynomials using only pairs of variables in the form of

$$\hat{y} = a_0 + a_1 x_i + a_2 x_j + a_3 x_i x_j + a_4 x_i^2 + a_5 x_j^2 \quad (3)$$

The basic form of the GMDH algorithm has several limitations, e.g., each host can only have two input variables, and the neurons in each layer are only connected to the host in its adjacent layer. Therefore, we choose GA-GMDH to remove these restrictions, as each neuron in GA-GMDH can have a different number of input variables as well as a different order of polynomial.

The representation of the GA-GMDH network should contain the number of input variables for each neuron, the best type of polynomial for each neuron, and which input variables should be chosen for each neuron. Therefore, the chromosome for each individual should contain three subchromosomes. Each subchromosome in our algorithm is represented as a string of integer digits.

B. VM Selection

Once the system get the predicted load, it has been decided that a host is *overloaded* or *under loaded*, the next step is to select particular VMs to migrate from this host. In this section we propose two policies for VM selection.

1) The minimum migration time policy: The Minimum Migration Time (MMT) policy migrates a VM *v* that requires the minimum time to complete a migration relatively to the other VMs allocated to the host. The migration time is estimated as the amount of RAM utilized by the VM divided by the spare network bandwidth available for the host *j*. Let $\mathbf{V_j}$ be a set of VMs currently allocated to the host *j*. The MMT policy finds a VM *v* that satisfies conditions formalized in

$$v \in V_j \mid \forall a \in V_j, \frac{RAM_u(v)}{NET_j} \le \frac{RAM_u(a)}{NET_j} \quad (4)$$

where $RAM_u(a)$ is the amount of RAM currently utilized by the VM *a*; and NET_j is the sparenetwork bandwidth available for the host *j*.

2) The random selection policy (RSP): The Random Selection Policy (RSP) selects a VM to be migrated according to a uniformly distributed discrete random variable $X \stackrel{\text{def}}{=} \boldsymbol{U}(\mathbf{0}, |\boldsymbol{V_j}|)$, whose values index a set of VMs $\boldsymbol{V_j}$ allocated to a host *j*.

V. PERFORMANCE EVALUATION

A. Experiment Setup

We use CloudSim toolkit [5] as the simulation platform. We have simulated a data center that comprises 1000 heterogeneous hosts. In order to compare the efficiency of the algorithms we use three metrics to evaluate their performance. The first metric is the total energy consumption(EC). The second metric is the level of SLA violations(SLAV). The last one is the number of VM migrations.

B. Host Overload Prediction

The accurate prediction of host load in a Cloud computing data center is very important to improve resource utilization, lower data center costs and ensure the job performance. We quantified the performance of actual load prediction with mean squared error(MSE).

$$MSE = \frac{1}{H}\sum_{i=1}^{H}(A_i - F_i)^2 \quad (5)$$

Where H is the step length, A_i and F_i are the actual value and forecast value.

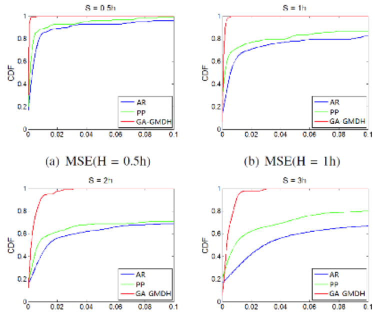

FIGURE I. MSE OF ACTUAL LOAD PREDICTION.

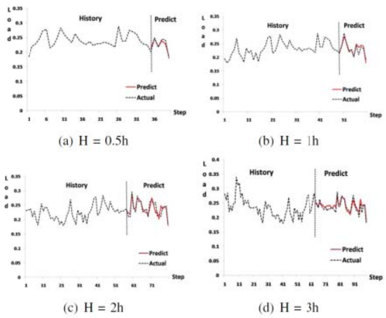

FIGURE II. ACTUAL LOAD PREDICTION.

In Figure 1, we compare our method with the AR method and the Pattern Prediction (PP) method proposed by Yang [11].The average MSE of our method in 3h ahead prediction is 0.0046, which is much lower than the other two methods. Figure 2shows the load prediction results of two different type of hosts in the Google data center.

C. Simulation Results

To make a simulation-based evaluation applicable, it is important to conduct experiments using workload traces from a real system. For our experiments we have used data coming from the cluster workload traces of Google datacenters. The interval of utilization measurements is 5 minutes. We have randomly chosen record of 1600 tasks running on 1000 hosts of 29 days from the workload traces collected from May 2011 [14]. During the simulations, each VM is randomly assigned a workload trace from one of the VMs from the corresponding day. In the simulations we do not limit the VM consolidation by the memory bounds, as this would constrain the consolidation, whereas the objective of the experiments is to stress the consolidation algorithms.

The average results of 10 data centers of the combinations of each host load detection algorithm and the MMT policy are shown in Table I.

TABLE I. AVERRAGE RESULTS.

Algorithms	**Energy (KWH)**	**SLA violation(%)**	**VM migration($\times 10^3$)**
LR-RSP	84.94	4.38	17.98
LR-MMTP	83.82	4.32	17.33
GMDH-RSP	83.54	4.30	16.77
GMDH-MMTP	81.93	4.26	13.37

We have simulated all combinations of the host load detection algorithms(local regression(LR), and GMDH) and VM selection policies(MMTP, RSP).The results produced by the selected algorithms are shown in Figure 3.

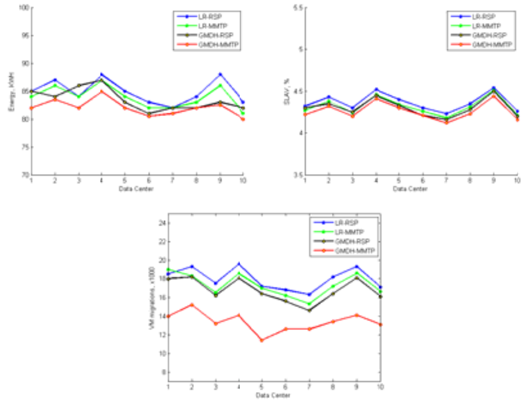

FIGURE III. ENERGY CONSUMTIONSLAVIOLATIONSVM MIGRATIONS.

From the observed simulation results, we can see that the combination of GA-GMDH with MMTP algorithms outperform others.

VI. CONCLUSION

We propose to combine GA-GMDHalgorithmand MMT policy for optimal online deterministic algorithms for these problems. The results of the experiments have shown that the proposed GA-GMDH prediction algorithm combined with the MMT selection policy significantly outperforms other VM consolidation algorithms in regard to the MSE metric due to a lower value in a long time interval and a substantially reduced level of SLA violations and the number of VM migrations.

ACKNOWLEDGMENT

This work is supported in part by the National Natural Foundation of China under Grant No. 61271231 and the scientific and technical supporting programs of Jiangsu province, china, No.BE2011169.

REFERENCES

[1] B. Urgaonkar, P. Shenoy, A. Chandra and P. Goyal, Dynamic Provisioning of Multi-tier Internet Applications, Proc of ICAC,2005.

[2] S. Osman, D. Subhraveti, G. su and J. Nieh, The Design and Implementation of Zap: A System for Migrating Computing Environments, Proc of OSDI, 2002.

[3] A. G. Ivakhnenko, Polynomial theory of complex systems, IEEE Trans. Syst. Man Cyber., vol. SMC-1, no. 4, pp. 364-378, 1971.

[4] Beloglazov A., Buyya R., Optimal Online Deterministic Algorithms and Adaptive Heuristics for Energy and Performance Efficient Dynamic Consolidation of Virtual Machines in Cloud Data Centers, Concurrency Computat.: Pract. Exper. 2012; 24:1397–1420

[5] Calheiros RN, Ranjan R, Beloglazov A, Rose CAFD, Buyya R. CloudSim: a toolkit for modeling and simulationof Cloud computing environments and evaluation of resource provisioning algorithms. Software: Practice andExperience 2011; 41(1):23–50.

[6] C. Dabrowski and F.Hunt, Using markov chain analysis tostudy dynamic behavior in large-scale grid systems, SeventhAustralasian Symposium on Grid Computing and Research,ser. CRPT, vol. 99. Wellington, New Zealand:ACS, pp. 29-40,2009.

[7] S. Akioka and Y. Muraoka, Extended forecast of cpu andnetwork load on computational grid, in Proceedings of the2004 IEEE International Symposium on Cluster Computeringand Grid(CCGrid'2006), pp. 321-326, 2006.

[8] Y. Wu, Y. Yuan, G. Yang and W. Zheng, Load PredictionUsing Hybrid Model for Computational Grid, Proceedings of8th IEEE/ACM International Conference on Grid Computing,2007.

[9] B. Guenter, N. Jain, and C. Williams, Managing Cost, Performance,and Reliability Tradeoffs for Energy-Aware ServerProvisioning, INFOCOM, pp. 1332-1340, 2011.

[10] Q. Zhang, M. F. Zhani, S. Zhang, Q. Zhu, R. Boutaba, and J.L. Hellerstein, Dynamic Energy-Aware Capacity Provisioningfor Cloud Computing Environments, Proceedings of the 9thinternational conference on Autonomic computing, 2012.

[11] D. Yang, J. Cao, C. Yu, and J. Xiao, A Multi-step-aheadCPU Load Prediction Approach in Distributed System, SencondInternational Conference on Cloud Computing GreenComputing, 2012

[12] S. DI, D. Kondo and W. Cirne, Host overload prediction in aGoogle Compute Cloud with a Bayesian Model,Procceedingsof the International Conference on High Performance Computing,Networking, Storage and Analysis, 2012

[13] H. S. Kim, R. Eykholt, J.D. Salas, Nonlinear dynamics,delay tiems and embedding windows, Physica D: NonlinearPhenomena, 1999.

[14] https://code.google.com/p/googleclusterdata/wiki/ClusterData2011_1