2013 IEEE Eighth International Conference on Networking, Architecture and Storage

A New Approach to Multi-objective Virtual Machine Placement in Virtualized Data Center

Sinong Wang[1], Huaxi Gu[1,2], Gang Wu[1]
1. State Key Laboratory of ISN, Xidian University, Xi'an, China
2. Science and Technology on Information Transmission and Dissemination in Communications Laboratary
Email: wangsinong@stu.xidian.edu.cn

***Abstract*—In this paper, a virtual machine placement model to maximize resource utilization, balance multi-dimensional resources use and minimize communication traffic simultaneously within the data center is proposed. The multi-objective problem is simplified by employing average valued inequality and positional constraints. The improved genetic algorithm with local heuristic method and elitism strategy is developed to solve the problem. The simulation results show that performance gains in all aspects can be achieved by the proposed model and algorithm compared to the existing algorithms.**

***Index Terms*—virtual networks; virtual machine placement; genetic algorithm; local heuristic;**

I. Introduction

With the spectrum of application paradigm for the data center services continuing to extend, efficient resource management within the data center becomes more important. One of the key technologies to this problem is virtualization [1], which improves the resource utilizations, scalabilities, flexibilities and availabilities of applications. Based on the virtualization technology, individual physical machines(PMs) can be carved into multiple virtual machines(VMs), each of which grantees its required resource by slicing a portion of resources from PM [2]. As a result, infrastructure provider just need to regard these VMs as basic deployment and management unit instead of PMs to efficiently manage the resources within the data center. However, a core challenge at the infrastructure level is the placement of the VMs on PMs.

Several solutions concentrated on this problem have been suggested in literature[3-15]. Different goals are proposed to optimize, such as maximizing the data center resource utilization [3-7], minimizing the communication traffic [8-10], balancing the multi-dimensional resources use [4][11] and reducing the power consumption [12][13].

To the best of our knowledge, few works[11][15] take above demands into account simultaneously to achieve combining profits. [11] introduces a mutlti-objective VM deployment and reconfiguration optimization framework,which aims to maximize resource utilization and balance multi-dimensional utilization. And it adopts GGA [16] to solve the model. [15] propose a two-level control system to manage the mappings of workloads to VMs and VMs to physical resources. It formulates the VM placement problem as multi-objective optimization problem of minimizing power consumption, total resource wastage and thermal dissipation costs. However,it is unlikely that a solution is optimal for one demand, while optimal for another [8].

In this paper, a single-objective model is proposed to achieve combining profits of three objectives: maximizing resource utilization, minimizing the traffic load and balancing the multi-dimensional utilization. The main contributions of this work can be summarized as:

- Single-objective and nonlinear characteristics of the model are achieved by two techniques: expressing objectives of maximizing resource utilization and balancing resource use in one formula by *Inequality of arithmetic and harmonic means*. transforming minimizing communication traffic into constraint condition by introducing positional constraint.
- An efficient algorithm called *IGA*(improved genetic algorithm) that have unique crossover operator and mutation operator are designed in view of this situation.

The rest of the paper is organized as follows. The overview of the framework is presented in the Section II, and the details of initial VM deployment and runtime VM deployment are given in the Section III. Experiments and evaluations are shown in Section IV. Finally, we give a conclusion and introduce the future work in Section V.

II. VM Placement Model

We consider a data center consisting of N heterogeneous servers, each of which has three types of physical resources capacity such as CPU cycles, memory size and bandwidth, denoted by a vector:

$$R_n = (Cpu_n, Mem_n, Bnd_n), n = 0, 1, \cdots, N-1. \quad (1)$$

Let the number of the applications hosted by the data center be denoted by M. Considering that some applications may span over a set of VMs, such as Map-Reduce computational applications and multi-tier Web services. For any $0 \le i \le M-1$, application i requires w_i virtual machines. And for any $0 \le i \le M-1$, $0 \le j \le w_i - 1$, VM_{ij} represents the the corresponding virtual machine serving application i, each VM_{ij} demands for CPU cycles, memory sizes, and bandwidth to handle one request, also donated by a resource vector:

$$D_{ij} = (Cpu_{ij}, Mem_{ij}, Bnd_{ij}) \quad (2)$$

978-0-7695-5034-3/13 $26.00 © 2013 IEEE
DOI 10.1109/NAS.2013.54

A. VM Placement

For each application i, A set of physical machines need to be found to host the w_i virtual machines. Let the binary variable X_{ij}^n represent the location of VM_{ij}, which is equal to 1 if the j_{th} VM of application i is placed on the physical machine n and equal to 0, otherwise. The solution space for X is characterized by

$$X = \{X_{ij}^n \mid X_{ij}^n \in \{0,1\}, \forall i, j\}$$

Subject to:

$$\sum_{i=1}^{N} X_{ij}^n = 1, \forall i, j \tag{3}$$

$$\sum_{i=1}^{N} \sum_{j=1}^{w_i} X_{ij}^n \cdot D_{ij} \leq R_n, \forall n \tag{4}$$

$$max\{distance(VM_{ij_1}, VM_{ij_2})\} \leq d_i, \forall i \tag{5}$$

The equation (3) suggests that each VM is assigned to exactly only one PM. And the inequalities (4) represent the resource constraint in each PMs.

Furthermore, the positional constraints (5) to minimize the traffic load within the data center are introduced. The $distance(VM_{ij_1}, VM_{ij_2})$ is the number of switches on the routing path from VM_{ij_1} to VM_{ij_2}, and the d_i is the max number of switches on the routing path between any VM_{ij} in application i. Commonly, VMs with high inter traffic on the neighboring PM should be consolidated, because less switches are needed, thus, reducing link load for the entire data center. Based on this consideration, d_i can be exploited as a barrier in the process of VM placement, which is the max number of switches on the routing path between any two virtual machines for application i. For those applications with a large amount of inter-VM bandwidth, d_i can be smaller. As shown in Fig 1, different applications are assigned to the different d_i due to the different size of traffic generated. The larger d_i, the larger solution space for X, and the larger traffic generated within the data center. Therefore, there will be a equilibrium between resource utilization and communication traffic, which can be obtained by adjusting d_i.

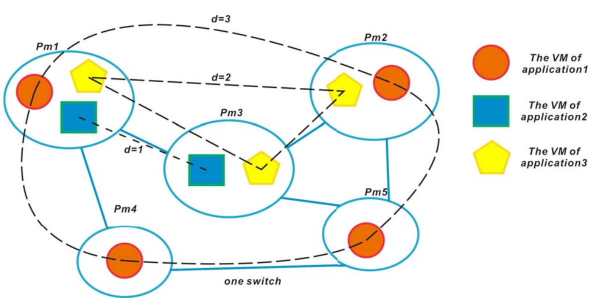

Fig. 1. A feasible placement in a date center. Application 1 has 4 VMs and lowest inter-VM bandwidth, thus $d_i = 3$; application 2 is a multi-tier Web services which has 2 VMs and highest inter-VM bandwidth, thus $d_i = 1$; application 3 has 3 VMs and medium inter-VM bandwidth, thus $d_i = 2$.

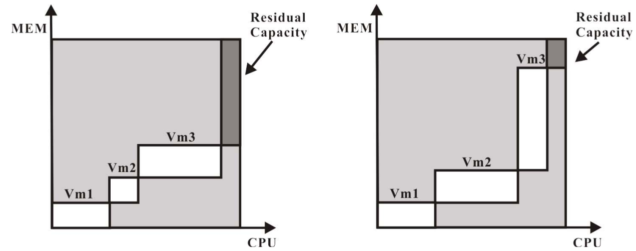

Fig. 2. Illustrating the different resources utilization in cpu and memory.

B. Objective Function

Let us concentrate on this problem and attempt to define a suitable function to optimize.

There is a fact that unbalanced use of the multi-dimensional resources can not only reduce holistic resource utilization significantly, but also initiate some anomalies[3]. As defined in [4], *the balance of the multi-dimensional resources utilization means the resource in each dimension is utilized proportionally to the total amount*. If the infrastructure provider place VMs regardless of this characteristic, some resources may run out while others are idle. As shown in Fig2, the former PM hosting two cpu-demand VMs leads to the cpu utilization is much higher than other resource utilization, while the latter is a balanced placement. It can be estimated that the holistic resource utilization of the latter is higher than the former.

In this circumstance, it can be observed that resources with a high utilization will inhibit utilizing resources in other dimensions. Therefore, a simple rule can be proposed: If some resources utilization ratio is much higher, we can reduce their contribution to the holistic resource utilization by cutting down their weights in the objective function. Then traditional constant weights w_{ij} can be changed into into function $w_{ij}(r_{i1}, r_{i2}, r_{i3})$, and a simple form is to inverse each resource utilization, that is:

$$w_{ij}(r_{i1}, r_{i2}, r_{i3}) = \frac{1/r_{ij}}{\sum_{j=1}^{3} r_{ij}}, i = 0, 1, \cdots N-1, j = 1, 2, 3 \tag{6}$$

where r_{i1}, r_{i2}, r_{i3} donate the cpu, memory size and bandwidth resource utilization ratio of server i.

According to the definition of the above weights, objective function f about holistic resource utilization can be obtained via a simple derivation:

$$\begin{aligned} f &= \frac{1}{N'} \sum_{i=1}^{N'} (w_{i1} r_{i1} + w_{i2} r_{i2} + w_{i3} r_{i3}) \\ &= \frac{1}{N'} \sum_{i=1}^{N'} \Big(\frac{1/r_{i1}}{\sum_{j=1}^{3} 1/r_{ij}} r_{i1} + \frac{1/r_{i2}}{\sum_{j=1}^{3} 1/r_{ij}} r_{i2} + \\ &\qquad \frac{1/r_{i3}}{\sum_{j=1}^{3} 1/r_{ij}} r_{i3} \Big) \\ &= \frac{1}{N'} \sum_{i=1}^{N'} \frac{3}{1/r_{i1} + 1/r_{i2} + 1/r_{i3}} \end{aligned} \tag{7}$$

where N' donates the the number of PMs being used.

The following theorem characterizes the proposed objective function's progressiveness in maximizing and balancing each dimensional resource utilization ratio compared to the traditional function mathematically.

Theorem1. *The objective function (7) is smaller than the traditional objective function only by equalizing the three dimensional resource utilization.*

$$\frac{1}{N'}\sum_{i=1}^{N'}\frac{3}{1/r_{i1}+1/r_{i2}+1/r_{i3}} \leq \frac{1}{N'}\sum_{i=1}^{N'}\frac{r_{i1}+r_{i2}+r_{i3}}{3} \tag{8}$$

where the right equation represents the traditional objective function whose weights are stationary and equalized. This inequality can be proved by Cauchy-Schwarz Inequality:

$$(\frac{1}{r_{i1}}+\frac{1}{r_{i2}}+\frac{1}{r_{i3}})(r_{i1}+r_{i2}+r_{i3}) \geq (\frac{\sqrt{r_{i1}}}{\sqrt{r_{i1}}}+\frac{\sqrt{r_{i2}}}{\sqrt{r_{i2}}}+\frac{\sqrt{r_{i3}}}{\sqrt{r_{i3}}})^2 = 9$$

$$\frac{3}{1/r_{i1}+1/r_{i2}+1/r_{i3}} \leq \frac{r_{i1}+r_{i2}+r_{i3}}{3}$$

Then we can get

$$\frac{1}{N'}\sum_{i=1}^{N'}\frac{3}{1/r_{i1}+1/r_{i2}+1/r_{i3}} \leq \frac{1}{N'}\sum_{i=1}^{N'}\frac{r_{i1}+r_{i2}+r_{i3}}{3}$$

The inequality (8) shows the harmonic mean is smaller than the arithmetic mean in the average valued inequality, Equal if only if $r_{i1} = r_{i2} = r_{i3}$ for each i.

The proposed objective function is up close to the traditional objective function and equal if only if the three dimensional utilization ratio equal.

Based on the theorem 1, it can be concluded that if a solution is optimal in the traditional function, it will drop in our proposed function because of the unbalance use of three dimensional resources. In other words, it still contains some searching space to optimize, which is the solution space from unbalance to balance.

At last, a constant k is introduced to express our concentration on the well-filled 'elite' PMs in comparison to the less filled ones, which also change the algorithm's convergence and the most appropriate k will be obtained in the section IV.

III. Proposed Genetic Algorithm

Since the VM placement model is to maximize objective function, our proposed objective function can be regarded as fitness function directly.

As defined in our model, a binary variable X_{ij}^n is choosed to represent the position of each VM, which facilitates the calculation of objective function while complicates the process of the crossover and mutation. So its dimension can be cut down by the following rule:

$$f_{ij} = \begin{cases} position(X_{ij}^n), & 0 \leq i \leq N-1, 1 \leq j \leq w_i \\ 0, & 0 \leq i \leq N-1, w_i \leq j \leq L \end{cases} \tag{9}$$

since different applications require different number of VMs, the second dimension of matrix F might be uneven. To guarantee its uniformity, we choose $L = max(w_i)$ as the second dimension, and set the rest to 0. Then matrix F can be regard as the chromosome structure for individual who represents a feasible placement.

Optimization operators include the crossover operator, the mutation operator and the selection operator. Bad operators work against algorithm's own process towards destruction of the good schemata [16]. Thus, we should design an operator that has the properties of order-preserving in the iteration of algorithm.

Crossover Operator: It can be found that traditional cross cannot guarantee the efficient optimization result, but it can get the result in short time if it is combined with the local heuristic method. The process of our designed crossover is listed as follow:

- **step1:** Two individuals F_{a_1}, F_{a_2} are chosen randomly in the initial population.
- **step2:** Choosing an element f_{ij} randomly from parental chromosome F_{a_1}.
- **step3:** Calculating the range of alternative elements in the F_{a_2} according to the positional constraint (5)
- **step4:** Calculating the residual resources both in the F_{a_1}, F_{a_2} after each swap.
- **step5:** Choosing the vm-exchange way that generates the smallest residual resources and treat the new individuals F_{a_1}, F_{a_2} as the offspring.

Mutation Operator: In genetic algorithm, Mutation Operator can assure the diversity of the population to avoid the algorithm accesses premature convergence. However, mutation rate should not be too high to avoid mutation interference. And our Mutation Operators perform large chromosome structural variation by using crossover operators iteratively.

Selection Operator: In order to guarantee the properties of order-preserving in the iteration of algorithm. The elitism strategy is adopted instead of traditional roulette wheel selection(RWS). Only the fittest individuals can keep in the next generation.

Since genetic algorithm is sort of stochastic algorithm, it cannot be predicted when it will get the max fitness. A simple method is to terminate the algorithm after a fixed number of generations. And in the process of evolution, a terminal condition is also appended that if the fitness D-value between offspring and parental generation is less than predefined threshold, algorithm should be terminated.

Fig. 3. PSEUDOCODE FOR THE IMPROVED GENTIC ALGORITHM

```
Initialize parental population PVmFlag by mentocarlo
//evolution process of genetic algorithm
For   i = 1 to MaxGeneration
      X = Encoding(PVmFlag)
      //cross operator
      For   j = 1 to PopulationNumber*CrossRate
            Choose two individuals a1 and a2
            (b1,b2) = Cross(a1,a2)
            Caculate the new fitness function
      End   For
      //mutation operator
      For   j = 1 to PopulationNumber*MutationRate
            Choose an individual a randomly
            b = Mutation(a)
            Caculate the new fitness function g(b)
      End   For
      Construct offspring OVmFlag
      PVmFlag = Seclect (PVmFlag,OVmFlag)
      If (NewBest - OldBest < 1e-5)   Break   End If
End For
```

IV. Evaluation

In this section, the effectiveness of our proposed VM placement model and algorithm on each aspect are evaluated, and simulation setups and results are presented.

For simulation, model parameters are generated based on the real world parameters.The number of servers is set to 100, each of which is assumed to have the different amount of resources:$U(1000, 4000)MIPS$ cpu cycles, $U(500, 2000)GB$ memory size and $U(800, 1500)Mbps$ bandwidth. We assume each application requires $\{$1,2,3,4,5$\}$ VMs randomly and the typology limitation follows the uniform distribution from one PM to the overall PMs, and each VMs require a basic resources: $N(2, 0.25)$ cpu cycles, $N(4, 1)GB$ memory size and $U(300, 800)Kbps$ bandwidth. The number of new applications is set from

We define the convergence coefficient $c(k)$ to estimate the efficiency of algorithm under different k, which will increase when fitness function increases, decrease when the cost of algorithm increases. It can be defined:

$$c(k) = \frac{maxFitness^{\frac{1}{k}}}{maxGeneration} \tag{10}$$

As shown in Table I, it can be found that $k = 1.5$ gives out good result.and make the searching more robust. Larger or smaller value of k seems to lead to a premature convergence of the algorithm and trap in local optimum.

TABLE I
DIFFRENT k AND $c(k)$

k	0.5	1	1.5	2	2.5	3
$c(k)$	0.0012	0.0015	0.0017	0.0015	0.0013	0.0012

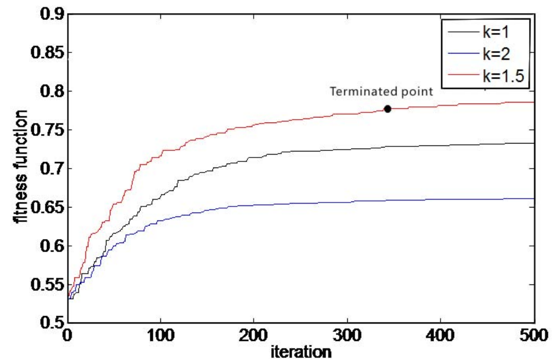

Fig. 4. Illustrating the different fitness function in the process of iteration

As shown in Fig5, we compare the resource utilization ratio, minimal cosine value [11], total traffic [8] as well as the fitness value for each of the algorithms *GREEDY, GA, BA, CLUSTER* and our proposed *IGA* under consideration.

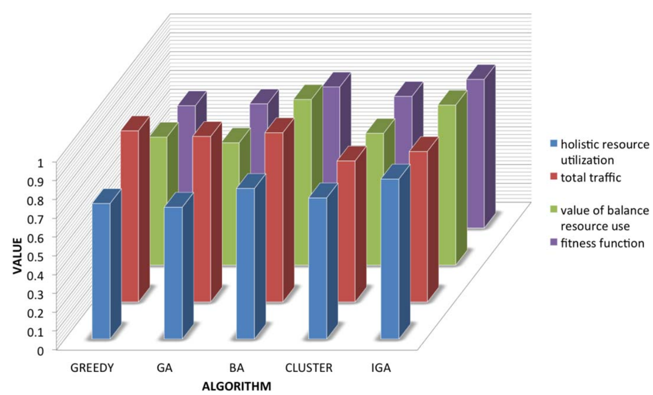

Fig. 5. Performance comparisons of five placement algorithms in four aspects.The resource utilization ratio, minimal cosine value and the fitness value is the real value, while the total traffic is divided by 2000 to show more clearly

Our proposed algorithm yields the maximum resource utilization because we construct a single-objective model to optimize resource utilization while other model construct more than one objective and they just get the Pareto Optimum, and our model enlarges the solution space compared to the other model emphasizing that a set of VMs serving the same application should be located in the same PM. The minimal cosine value of our algorithm is also high because the placement tries to equalize the three dimensional resources utilization to make the objective function approach its upper bound. Among them, *IGA* also produce a low communication traffic within the data center.

Since positional constraints of our VM placement model is constructed based on the characteristic of typology structure, it is necessary to estimate the value of objective function in different typology structure such as Tree, Fat-Tree, VL2 and BCube.

As shown in Fig 6, the value of fitness function in BCube is much higher than the other typological structure because for a data center of fixed amount of PMs, BCube network requires the min levels of intermediate switches [9], which weakens the positional constraints and enlarges the searching space.

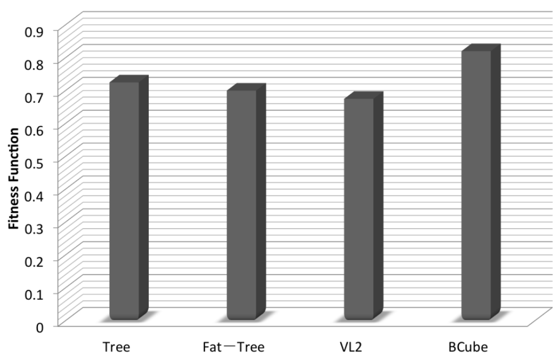

Fig. 6. Different fitness function under different typological structure

V. Conclusion and Future Work

In this paper, a VM placement model to maximize resource utilization, balance multi-dimensional resources use and minimize communication traffic is described, and an improving genetic algorithm is designed to solve this problem. The algorithm adapts genetic operators with local heuristic method and Elitism strategy into iteration in finding the best placement. Simulation results show that our proposed model and algorithm can maximize the resource utilization, balance multi-dimensional resources utilization and minimize communication traffic compared to the GREEDY, GA, BA and CLUSTER.

For future work, we investigate the positional constraints in depth, because reasonable set of positional conditions is so complicated that requires more scientific mathematical method. Furthermore, we plan to design a VM deployment and reconfiguration framework, because our proposed techniques speed up the solving speed, which can be exploited in runtime VM deployment.

VI. Acknowledgements

This work is supported by the National Innovative Experimental Projects for University Students of China NO.20120701010; the National Science Foundation of China Grant No.61070046; the 111 Project Grant No.B08038; the fund from Science and Technology on Information Transmission and Dissemination in Communication Networks Laboratory NO.ITD-U12002.

References

[1] J. E. Smith, and R. Nair, "The Architecture of Virtual Machines," Computer, 38(5):32-38, 2005.
[2] Jenn-Wei Lin, and Chien-Hung Chen, "Interference-Aware Virtual Machine Placement in Cloud Computing Systems", International Conference on Computer & Information Science, 2012, pp. 598-603
[3] S.chaisiri, B.-S.Lee, and D. Niyato, "Optimal Virtual Machine Placement across Mutilple Cloud Providers", IEEE Asia-Pacific Services Comput. Conf. 2009, pp.103-110.
[4] M. Mishra and A. Sahoo, "On Theory of VM Placement: Anomalies in Existing Methodologies and Their Mitigation Using a Novel Vector Based Approach", 4th International Conference on Cloud Computing, 2011.
[5] D. Gmach, J. Rolia, L. Cherkasova, "Satisfying Service Level Objectives in a Self-Managing Resource Pool," IEEE International Conference on Self-Adaptive and Self-Organizing Systems, September, 2009.
[6] X. Liu, X. Zhu, S. Singhal, and M. Arlitt, "Adaptive entitlement control of resource partitions on shared servers," International Symposium on Integrated Network Management, 2005.
[7] B. Urgaonkar, P. Shenoy, A. Chandra, and P. Goyal, "Dynamic Provisioning of Multi-tier Internet Applications," International Conference on Autonomic Computing, 2005.
[8] Bolei Zhang, Zhuzhong, QianWei Huang, Xin Li and Sanglu Lu, "Minimizing Communication traffic in Data Centers with Power-aware VM Placement", Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, pp. 280-285, 2012.
[9] X. Meng, V. Pappas, and Li. Zhang. "Improving the Scalability of Data Center Networks with Traffic-aware Virtual Machine Placement", INFOCOM, 2010.
[10] J. D. Sonnek, J. B. S. G. Greensky, R. Reutiman, and A. Chandra, "Starling: Minimizing communication overhead in virtualized computing platforms using decentralized affinity-aware migration," in ICPP, 2010, pp. 228-237.
[11] Wei Chen, Xiaoqiang Qiao, Jun Wei, Tao Huang, "A Profit-aware Virtual Machine Deployment Optimization Framework for Cloud Platform Providers", Fifth IEEE International Conference on Cloud Computing.
[12] Hadi Goudarzi and Massoud Pedram, "Energy-Efficient Virtual Machine Replication and Placement in a Cloud Computing System", IEEE Fifth International Conference on Cloud Computing, 2012, pp.750-757
[13] Wang Xiaoli, Liu Zhanghui, "An energy-aware VMs placement algorithm in Cloud Computing environment", International Conference on Intelligent System Design and Engineering Application, 2012
[14] N.Bobroff, A. Kochut, and K. Beaty, "Dynamic placement of virtual machines for managing SLA violations", 10th IFIP/IEEE International Symposium on, 2007, PP, 119-128
[15] Jing Xu and Jos A. B. Fortes, "Multi-objective Virtual Machine Placement in Virtualized Data Center Environments", IEEE/ACM International Conference on Green Computing and Communications, 2012
[16] E.Falkenauer and ADelchambre, "A Genetic Algorithm for Bin Pcking and Line Balancing", IEEE International Conference on Robotics and Automation, 1992