UDC 004.7:004942(045)

Vitaliy Litvinov[1]
Karyna Matsueva[2]

RESOURCE-EFFICIENT ALLOCATION HEURISTICS FOR MANAGEMENT OF DATA CENTERS FOR CLOUD COMPUTING

[1]Institute of Problems of Mathematical Machines and Systems, National Academy of Sciences of Ukraine
Glushkov avenue 42, Kyiv, Ukraine
[2]National Aviation University
Kosmonavta Komarova avenue 1, 03680, Kyiv, Ukraine
E-mails: [1]vlitvin@ukrsoft.ua; [2]karyna_matsueva@bigmir.net

Abstract. *Survey of research in resource-efficient computing and architectural principles forresource-efficient management of Clouds are offered in this article. Resource-efficient resource allocation policies and scheduling algorithms considering QoS expectations and power usage characteristics of the devices are defined.*

Keywords: cloud computing; dynamic consolidation virtualization; energy efficiency; resource management.

1. Introduction

Cloud computing can be classified as a new paradigm for the dynamic provisioning of computing services supported by state-of-the-art data centers that usually employ Virtual Machine (VM) technologies for consolidation and environment isolation purposes [1].

Cloud computing model has immense potential as it offers significant cost savings and demonstrates high potential for the improvement of energy efficiency under dynamic workload scenarios.

Nowadays, high performance has been the sole concern in data center deployments, and this demand has been fulfilled without paying much attention to energy consumption.

However, an average datacenter consumes as much energy as 25,000 households.

As energy costs are increasing while availability dwindles, there is need to shift the focus from optimizing data center resource management for pure performance to optimizing them for energy efficiency, while maintaining high service level performance.

Lowering the energy usage of data centers is a challenging and complex issue because computing applications and data are growing so quickly that increasingly larger servers and disks are needed to process them fast enough within the required time period.

Green Cloud computing is envisioned to achieve not only the efficient processing and utilization of a computing infrastructure, but also to minimize energy consumption [2].

This is essential for ensuring that the future growth of Cloud computing is sustainable.

Otherwise, Cloud computing with increasingly pervasive frontend client devices interacting with back-end data centers will cause an enormous escalation of the energy usage.

To address this problem and drive Green Cloud computing, data center resources need to be managed in an energy-efficient manner.

2. The analysis of researches

One of the first works, in which power management has been applied at the data center level, has been done by E. Pinheiro R. Bianchini, E.V. Carera, T. Heath [7].

In this work the authors have proposed a technique for minimization of power consumption in a heterogeneous cluster of computing nodes serving multiple web-applications.

The main technique applied to minimize power consumption is concentrating the workload to the minimum of physical nodes and switching idle nodes off.

J.S. Chase, D.C. Anderson, P.N. Thakar, A.M. Vahdat, R.P. Doyle [3] have considered the problem of energy-efficient management of homogeneous resources in Internet hosting centers.

The main challenge is to determine the resource demand of each application at its current request load level and to allocate resources in the most efficient way.

A. Verma, P. Ahuja, A. Neogi [10] have formulated the problem of power-aware dynamic placement of applications in virtualized heterogeneous systems as continuous optimization: at each time frame the placement of VMs is optimized to minimize power consumption and maximize performance.

In contrast to the discussed studies, we propose efficient heuristics for dynamic adaption of VM allocation at run-time according to the current utilization of resources applying live migration, switching idle nodes to the sleep mode, and thus minimizing energy consumption.

The proposed approach can effectively handle strict SLAs, heterogeneous infrastructure and heterogeneous VMs.

The algorithms do not depend on a particular type of workload and do not require any knowledge about applications running in VMs. Recently, a number of research works have been done on the thermal efficient resource management in data centers [8].

The studies have shown that the software-driven thermal management and temperature-aware workload placement bring additional energy savings.

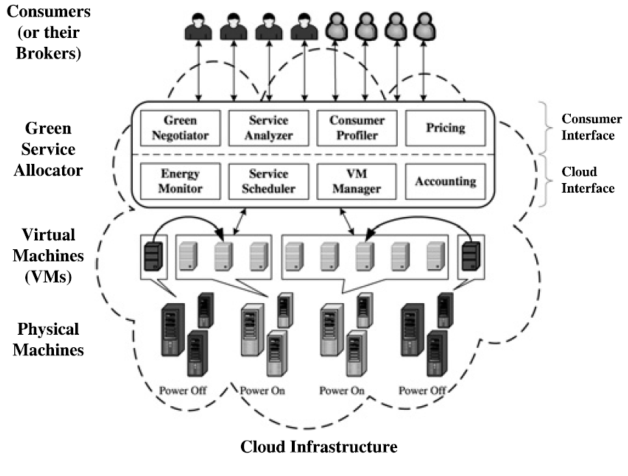
However, the problem of thermal management in the context of virtualized data centers has not beeninvestigated.

3. Green Cloud architecture

Clouds aim to drive the design of the next generation data centers by architecting them as networks of virtual services (hardware, database, user-interface, application logic) so that users can access and deploy applications from anywhere in the world on demand at competitive costs depending on their QoS requirements [6].

The high-level system architecture

Figure shows the high-level architecture for supporting energy-efficient service allocation in a Green Cloud computing infrastructure.



The high-level system architecture

There are basically four main entities involved:

1. Consumers/Brokers: Cloud consumers or their brokers submit service requests from anywhere in the world to the Cloud. It is important to notice that there can be a difference between Cloud consumers and users of deployed services.

2.Green Service Allocator: Acts as the interface between the Cloud infrastructure and consumers.

3. VMs: Multiple VMs can be dynamically started and stopped on a single physical machine according to incoming requests, hence providing the flexibility of configuring various partitions of resources on the same physical machine to different requirements of service requests.

By dynamically migrating VMs across physical machines, workloads can be consolidated and unused resources can be switched to a low-power mode, turned off or configured to operate at low-performance levels (e.g. using DVFS) in order to save energy.

4. Physical Machines: The underlying physical computing servers provide the hardware infrastructure for creating virtualized resources to meet service demands.

4. Power model

Power consumption by computing nodes in data centers is mostly determined by the CPU, memory, disk storage and network interfaces. In comparison to other system resources, the CPU consumes the main part of energy, and hence in this work we focus on managing its power consumption and efficient usage.

Recent studies [4, 5] have shown that the application of DVFS on the CPU results in almost linear power-to-frequency relationship for a server.

The reason lies in the limited number of states that can be set to the frequency and voltage of the CPU and the fact that DVFS is not applied to other system components apart from the CPU.

Moreover, these studies have shown that on average an idle server consumes approximately 70% of the power consumed by the server running at the full CPU speed.

This fact justifies the technique of switching idle servers to the sleep mode to reduce the total power consumption.

Therefore, in this work we use the power model defined in

$$P(u) = kP_{\max} \cdot u,$$

where u – the CPU utilization;

$P_{\max}$ – the maximum power consumed when the server is fully utilized;

k – the fraction of power consumed by the idle server (i.e. 70 %).

For our experiments $P_{\max}$ is set to 250 W, which is a usual value for modern servers.

The utilization of the CPU may change over time due to the workload variability.

Thus, the CPU utilization is a function of time and is represented as $u(t)$.

Therefore, the total energy consumption by a physical node (E) can be defined as an integral of the power consumption function over a period of time as shown in

$$E = \int_{t_0}^{t_1} P(u(t))dt.$$

5. Energy-aware allocation of data center resources

Recent developments in virtualization have resulted in its proliferation across data centers.

By supporting the movement of VMs between physical nodes, it enables dynamic migration of VMs according to the performance requirements.

When VMs do not use all the provided resources, they can be logically resized and consolidated to the minimum number of physical nodes, while idle nodes can be switched to the sleep mode to eliminate the idle power consumption and reduce the total energy consumption by the data center.

Currently, resource allocation in a Cloud data center aims to provide high performance while meeting SLAs, without focusing on allocating VMs to minimize energy consumption.

To explore both performance and energy efficiency, three crucial issues must be addressed.

First, excessive power cycling of a server could reduce its reliability.

Second, turning resources off in a dynamic environment is risky from the QoS perspective.

Due to the variability of the workload and aggressive consolidation, some VMs may not obtain required resources under peak load, and fail to meet the desired QoS.

Third, ensuring SLAs brings challenges to accurate application performance management in virtualized environments.

All these issues require effective consolidation policies that can minimize energy consumption without compromising the user-specified QoS requirements.

6. Placement of virtual machines

The problem of VM allocation can be divided in two: the first part is the admission of new requests for VM provisioning and placing the VMs on hosts, whereas the second part is the optimization of the current VM allocation.

The first part can be seen as a bin packing problem with variable bin sizes and prices.

To solve it we apply a modification of the Best Fit Decreasing (BFD) algorithm that is shown to use no more than 11/9 *OPT+ 1 bins (where OPT is the number of bins given by the optimal solution) [9].

In our modification, the Modified Best Fit Decreasing algorithms, we sort all VMs in decreasing order of their current CPU utilizations, and allocate each VM to a host that provides the least increase of power consumption due to this allocation.

This allows leveraging the heterogeneity of resources by choosing the most power-efficient nodes first. The pseudo-code for the algorithm is presented in Algorithm 1.

The complexity of the allocation part of the algorithm is n*m, where n is the number of VMs that have to be allocated and m is the number of hosts.

Algorithm 1: Modified BFD

```
Input: hostList, vmList Output: allocation of VMs

vmList.sortDecreasingUtilization()
foreach vm in vmList do
    minPower←MAX
    allocatedHost←NULL
    foreach host in hostList do
        f host has enough resource for vm then
            power←estimatePower(host, vm)
            if power <minPower then
                allocatedHost←host
                minPower←power
    if allocatedHost □= NULL then
        allocate vm to allocatedHost
return allocation
```

7. Selection of virtual machines

The optimization of the current VM allocation is carried out in two steps: at the first step we select VMs that need to be migrated, at the second step the chosen VMs are placed on the hosts using the MBFD algorithm.

To determine when and which VMs should be migrated, we introduce three double-threshold VM selection policies.

The basic idea is to set upper and lower utilization thresholds for hosts and keep the total utilization of the CPU by all the VMs allocated to the host between these thresholds.

If the CPU utilization of a host falls below the lower threshold, all VMs have to be migrated from this host and the host has to be switched to the sleep mode in order to eliminate the idle power consumption.

If the utilization exceeds the upper threshold, some VMs have to be migrated from the host to reduce the utilization.

The aim is to preserve free resources in order to prevent SLA violations due to the consolidation in cases when the utilization by VMs increases.

The difference between the old and new placements forms a set of VMs that have to be reallocated.

The new placement is achieved using live igration of VMs.

In the following sections we discuss the proposed VM selection policies.

8. The minimization of migrations policy

The Minimization of Migrations (MM) policy selects the minimum number of VMs needed to migrate from a host to lower the CPU utilization below the upper utilization threshold if the upper threshold is violated. Let V_j be a set of VMs currently allocated to the host j.

Then $P(V_j)$ is the power set of V_j.

The MM policy finds a set $R \in P(V_j)$ defined in

$$R = \begin{cases} \{S \,|\, S \in (V_j), & u_j - \Sigma_{v \in S} u_a < T_u; \\ |S| \to \min\}, & if u_j > T_u; \\ V_j, & if u_j < T_l; \\ \varnothing, & otherwise \end{cases}$$

where u_j – the current CPU utilization of the host j;

$u_a(v)$ – the fraction of the CPU utilization allocated to the VMv;

T_u – the upper utilization threshold;

T_l – the lowerutilization threshold.

The pseudo-code for the MM algorithm for the over-utilization case is presented in Algorithm 2.

The algorithm sorts the list of VMs in the decreasing order of the CPU utilization.

Algorithm 2: Minimization of Migrations

```
Input: hostList Output: migrationList
foreach h in hostList do
    vmList←h.getVmList()
    vmList.sortDecreasingUtilization()
    hUtil←h.getUtil()
    bestFitUtil←MAX
    while hUtil >THRESH_UP do
        foreach vm in vmList do
            if vm.getUtil() >hUtil − THRESH_UP then
                t←vm.getUtil() − hUtil + THRESH_UP
                if t <bestFitUtil then
                    bestFitUtil←t
                    bestFitVm←vm
            else
                if bestFitUtil = MAX then
                    bestFitVm←vm
                break
        hUtil←hUtil − bestFitVm.getUtil()
        migrationList.add(bestFitVm)
        vmList.remove(bestFitVm)
    if hUtil <THRESH_LOW then
        migrationList.add(h.getVmList())
        vmList.remove(h.getVmList())
return migrationList
```

The highest potential growth policy. When the upper threshold is violated, the Highest Potential Growth (HPG) policy migrates VMs that have the lowest usage of the CPU relatively to the CPU capacity defined by the VM parameters in order to minimize the potential increase of the host's utilization and prevent an SLA violation, as formalized in

$$R = \begin{cases} \{S \,|\, S \in P(V_j), & u_j - \Sigma_{v \in S} u_a(v) < T_u; \\ \Sigma_{v \cup S} \dfrac{u_a(v)}{u_r(v)} \to \min\}, & u_j > T_u; \\ V_j, & if u_j < T_l; \\ \varnothing, & otherwise \end{cases}$$

where $u_r(v)$ – the fraction of the CPU capacity initially requested for the VMv and defined as the VM's parameter.

We do not provide the pseudo-code for the HPG algorithm, as it is similar to the MM algorithm presented earlier.

The random choice policy. The Random Choice policy relies on a random selection of a number of VMs needed to decrease the CPU utilization by a host below the upper utilization threshold.

According to a uniformly distributed discrete random variable (X), whose values index subsets of V_j, the policy selects a set $R \in P(V_j)$, as shown in

$$R = \begin{cases} \{S \,|\, S \in P(V_j), & u_j - \Sigma_{v \in S} u_a(v) < T_u; \\ X \overset{def}{=} U(0, |P(V_j)| - 1)\}, & u_j > T_u; \\ V_j, & if u_j < T_l; \\ \varnothing, & otherwise \end{cases}$$

where X – a uniformly distributed discrete random variable used to select a subset of Vj .

9. Conclusions

Virtualization technology in the environment of cloud computing largely relies on the possibility of moving virtual machines between physical hosts using live or offline migration.

This provides a method of dynamic consolidation of virtual machines to minimize physical nodes according to current resource requirements. In the case of part-load units, they may be excluded or move into resource-mode (eg, sleep, hibernate) to reduce overall energy consumption in data centers .

In this paper we propose algorithms which use this method, and therefore in conjunction confirmed its effectiveness.

However, there are many open issues that must be addressed in order to fully exploit the potential of energy savings in the data center with cloud computing.

References

[1] *Barham, P.; Dragovic, B.;Fraser,K.;Hand, S.; Harris,T.; Ho, A.;Neugebauer, R.; Pratt, I.*; *Warfield, A.* Xen and the art of virtualization. Proceedings of the 19th ACM Symposium on Operating Systems Principles. SOSP 2003. New York. 2003. P. 177.

[2] *Buyya, R;. Beloglazov, A.; Abawajy, J.* Energy-efficient management of datacenter resources for cloud computing: a vision, architectural elements, andopen challenges. Proceedings of the 2010 International Conference on Parallel and Distributed Processing Techniques and Applications. PDPTA 2010. Las Vegas. 2010. P. 50-56.

[3] *Chase,J.S;.Anderson, D.C.; Thakar, P.N.; Vahdat, A.M.; Doyle, R.P.* Managing energy and server resources in hosting centers. Proceedings of the 18th ACM Symposium on Operating Systems Principles. ACM. New York. 2001. P. 103–116.

[4] *Gandhi, A.*; *Harchol-Balter, M.. Das, R.; Lefurgy, C.* Optimal power allocation in server farms. Proceedings of the 11th International Joint Conference on Measurement and Modeling of Computer Systems. ACM. New York. 2009. P. 157–168.

[5] *Kusic, D.; Kephart, J.O.; Hanson, J.E.; Kandasamy, N.; Jiang, G.* Power and performance management of virtualized computing environments vialookahead control. Cluster Computing. 2009. Vol. 12, N 1. P. 1–15.

[6] *Moore, J.; Chase, J.; Ranganathan, P.; Sharma, R.* Making scheduling ‘‘cool’’:temperature-aware workload placement in data centers. Proceedings of the Annual Conference on USENIX Annual Technical Conference. Anaheim. 2005. P. 19–25.

[7]. *Pinheiro, E.; Bianchini, R.;. Carrera, E.V.; Heath, T.* Load balancing and unbalancingfor power and performancee in cluster-based systems. Proceedings of theWorkshop on Compilers and Operating Systems for Low Power.2001. P. 182–195.

[8] *Sharma, R.K.; Bash, C.E.; Patel, C.D.; Friedrich, R.J.; Chase, J.S.* Balance ofpower: dynamic thermal management for internet data centers . IEEE Internet Computing. 2005. P. 42–49.

[9] *Yue, M.* A simple proof of the inequality FFD (L)<11/9 OPT (L)+ 1,for all l forthe FFD bin-packing algorithm . Acta Mathematicae Applicatae Sinica (English Series). 1991. Vol. 7, N4. P.321–331.

[10] *Verma, A.; Ahuja, P.; Neogi, A.* pMapper: power and migration cost aware application placement in virtualized systems, in: Proceedings of the 9th ACM/IFIP/USENIX International Conference on Middleware. Springer. 2008. P. 243–264.

Received 3 April 2014.

В.В. Литвинов[1], К.А. Мацуєва[2]. Ресурсозберігаючий розподіл навантаження для ефективного управління центром обробки даних із хмарними обчисленнями

[1]Інститут проблем математичних машин і систем Національної академії наук України, просп. Глушкова, 42, Київ, Україна
[2]Національний авіаційний університет, просп. Космонавта Комарова, 1, Київ, Україна,03680
E-mails: [1]vlitvin@ukrsoft.ua; [2]karyna_matsueva@bigmir.net

Визначено архітектурні рамки і принципи енергозберігаючих хмарних обчислень. Розглянуто алгоритми розподілу для енергоефективного управління в Cloud-обчислювальних середовищах. Показано ресурсозберігаючі можливості центрів обробки даних для надання евристики розподілу клієнтських додатків, щоб підвищити ефективність енергоспоживання центру обробки даних і визначити узгоджену якість обслуговування QoS. Подано огляд досліджень із ресурсоефективних обчислень. Запропоновано архітектурні принципи енергозберігаючого управління хмарами, політика розподілу ресурсоефективних можливостей і алгоритми планування з урахуванням очікування QoS, особливості характеристик використання пристроїв, науково-дослідні завдання, використовуючи які можна отримати істотні вигоди для постачальників і споживачів ресурсів.

Ключові слова: віртуалізація; динамічна консолідація; енергоефективність; управління ресурсами; хмарні обчислення.

В.В. Литвинов[1], К.А. Мацуева[2]. Ресурсосберегающее распределение нагрузки для эффективного управления центрами обработки даних с облачными вычислениями

[1]Институт проблем математических машин и систем Национальной академии наук Украины, просп. Глушкова, 42, Киев, Украина
[2]Национальный авиационный университет, просп. Космонавта Комарова, 1, Киев, Украина, 03680
E-mails: [1]vlitvin@ukrsoft.ua; [2]karyna_matsueva@bigmir.net

Определены архитектурные рамки и принципы энергосберегающих облачных вычислений. Рассмотрены алгоритмы распределения для энергоэффективного управления в Cloud-вычислительных средах. Показаны ресурсосберегающие возможности центров обработки данных для предоставления эвристики распределения клиентских приложений, чтобы повысить эффективность энергопотребления центра обработки данных и определить согласованное качество обслуживания QoS. Приведен обзор исследований ресурсоэффективних вычислений. Предложены архитектурные принципы энергосберегающего управления облаками, политика распределения ресурсоэффективних возможностей и алгоритмы планирования с учетом ожидания QoS, особенности характеристик использования устройств, научно-исследовательские задачи, используя которые можно получить существенные выгоды для поставщиков и потребителей ресурсов.

Ключевые слова: виртуализация; динамическая консолидация; облачные вычисления; управление ресурсами; энергоэффективность.

Litvinov Vitaliy (1940)**.** Doctor of Engineering. Professor.
Head of the Department Programming Engineering, Chernihiv National University of Technology, Chernihiv, Ukraine.
Education: National Technical University of Ukraine "Kyiv Polytechnic Institute", Degree in Electric Technics, Kyiv, Ukraine (1963).
Research area: methods for modeling the behavior of complex systems and cryptography, methods of management of technology systems.
Publications: 200.
E-mail: vlitvin@ukrsoft.ua

Matsueva Karyna (1989)**.** Postraduate student. Assistant.
Department of Computerized Control Systems, National Aviation University, Kyiv, Ukraine.
Education: National Aviation University , Degree in Computer Systems and Networks, Kyiv, Ukraine (2012).
Research area: distributed computations, the concept of sustainable development, methods for modeling the behavior of complex systems.
Publications: 22.
E-mail: karyna_matsueva@bigmir.net