A comparison of resource allocation process in grid and cloud technologies

Huda Hallawi [1], Jörn Mehnen [2] and Hongmei He[3]

[1] Computer Science Department, University of Kerbala, Iraq
[2] Digital Manufacturing Department, University of Strathclyde, Glasgow, UK
[3] Manufacturing Department, Cranfield University, Bedford, UK

huda.f@uokerbala.edu.iq

Abstract. Grid Computing and Cloud Computing are two different technologies that have emerged to validate the long-held dream of computing as utilities which led to an important revolution in IT industry. These technologies came with several challenges in terms of middleware, programming model, resources management and business models. These challenges are seriously considered by Distributed System research. Resources allocation is a key challenge in both technologies as it causes the possible resource wastage and service degradation. This paper is addressing a comprehensive study of the resources allocation processes in both technologies. It provides the researchers with an in-depth understanding of all resources allocation related aspects and associative challenges, including: load balancing, performance, energy consumption, scheduling algorithms, resources consolidation and migration. The comparison also contributes an informal definition of the Cloud resource allocation process. Resources in the Cloud are being shared by all users in a time and space sharing manner, in contrast to dedicated resources that governed by a queuing system in Grid resource management. Cloud Resource allocation suffers from extra challenges abbreviated by achieving good load balancing and making right consolidation decision.

Keywords. Cloud Resources Allocation; Grid Resources Allocation; Resources Consolidation; Load Balancing; Energy Consumption; Scheduling Strategies, Performances.

1. Introduction

Nowadays, Cloud and Grid computing are widely used in solving scientific problems; those technologies are based on providing computing services on demand just like conventional power and water National Grids; both technologies were developed with the goal of creating a scalable and powerful virtual computer out of a large collection of homogenous or heterogeneous systems that share various combinations of resources [1, 2, 3].

Resources allocation, also known as resource management is one of the major fields in both technologies, since it controls the way that resources and services are made available to use by entities like users, applications, or services, and to make sure of the efficient utilization of computing resources and to optimize the performance of the submitted tasks [4,5]. For instance, a grid manger in one geographical area may have limited access over the system components or there may be a discrepancy of resource availability in the highly distributed cyber physical systems. This discrepancy may lead to severe increase in the energy consumption during the execution of grid applications on one side, and to the realization of tasks submitted by gird users on the other side [6,7].

In Cloud Computing, scheduling and resource allocation is a widely considered problem due to its crucial role in the whole of Cloud paradigm. Generally, it controls a number of conflicting objectives; computing resources must be well-managed to prevent overloading and waste of bandwidth, processing unit, memory, etc. This waste relates directly to significant financial loss for large Cloud service providers with regards to energy, operational cost as well as dissatisfaction of the Cloud service user [6].

This paper addresses the comparison between the Resources Allocation processes in Grid and Cloud technologies while characterizing the differences between both technologies in detail. The paper is structured as follows: Grid and Cloud models are defined in Section II. The processes of resources allocation and scheduling for both Grid Computing and Cloud Computing are reviewed in Section III. The common associated problems in both models are discussed in Section IV, and finally the conclusion comes in Section V.

2. Background

2.1. Grid technology

Based on the idea of electricity gird the Grid Computing (GC) introduced to the world with the aim of integrating the grid with existing computing technologies like web and virtual reality computing to serve some complex scientific problems [8,9]. Grid computing is one of the many computer scientific disciplines which can be defined as a large-scale and multidisciplinary infrastructure based on computational and data intensive platform for solving large-scale scientific problems. GC is known with its outstanding characteristics such as reliability, security, dependability and coordination [2,8,10].

Generally, a Grid system is seen as a multi-layer architecture with a hierarchical management system that consists of two or three levels depending on the privilege access to system data, system services, and resources.

Buyya et al. in [11] defines the grid components in four layers as below:

- Grid fabric layer, that is composed of the grid resources, services, and local resource administration systems.
- Grid core middleware, which is in charge of services related to security, management authority, scheduling, and remote tasks submission.
- Grid user layer that comprises a set of end-users, service providers, and system administrators.
- Grid applications layer.

These layers present fundamental Grid architecture; both of middleware layer and user layer have effective impacts in the multi-criterion scheduling and resource management. The most critical characteristics of GC can be summarized as in [1,8,10,12]: the heterogeneity and the multiple administration controls. The computing resources and internetworking connections are almost heterogeneous, whereas the conventional Grid consists of different institutions, each of which has its own policy to control its resources, and hence, it provides a large scale distributed platform.

2.2. Cloud Technology

Cloud Computing is known as the cutting edge of distributed system that initially comprises of Cluster Computing and GC. Cloud computing has emerged as a computing model aimed at providing resources as a service according to pay-per-use paradigm [4]. It has been defined by (NIST) as "a model for enabling convenient, on demand network access to a shared pool of configurable computing resources (e.g. network servers, storage, applications and services) that can be rapidly provisioned or released with minimal management effort or services provider interaction. This Cloud model promotes availability." [13]. The sole principle of Cloud computing is that data is not stored in locally but in the data centers via the Internet. The key attributes which distinguish Cloud Computing from the other traditional computing can be described as in [6], [14], [15]:

- Underlying infrastructure and software is abstracted and offered as a service.
- Build on scalable and flexible infrastructure.
- Offers an on-demand service provisioning and quality of service (QoS) guarantees.
- Pay for use of computing resources without up-front commitment by Cloud users.
- Shared and multitenant.
- Accessible over the Internet by any device.

Due to all these attributes there has been a rapid growth in the demands for Cloud services. This brings more challenges for Cloud providers to provide resources to the Cloud subscribers, hence, emerging new paradigm "Federated Cloud", where many providers are collaborated to fulfil clients' requests [13], [16]. This collaboration can be seen as a large scale Cloud environment.

3. Resources Allocation in Grid And Cloud Technology

Grid and Cloud resources allocation and scheduling deals with a large number of processes (tasks) racing to obtain resources (CPU, memory, storage, network) in order to complete their tasks. This requires an optimal mapping between the competing processes and the underlying computing resources. Moreover, scheduling and resource allocation in Grid Computing is considered as an NP-complete optimization problem [17], [18], The complexity of this problem depends on the number of the objectives to be minimized such as task inter-relations, makespan, resources utilization and energy consumption ,while Cloud resources allocation problem is approved as NP-hard in a strong sense [19,20].

3.1. Resources Allocation in Grid Technology

In Grid Resource Allocation, a local resource manager (LRM) manages the computing resources for a Grid site which contains a pool of resources. When users submit their jobs, these jobs will be considered as batch jobs. Each job needs to be executed by leasing some of the free resources for some time specified by the user. The hosted resources could not be shared with other jobs for that time figure 1 presents a basic batch scheduler in Grid. Resource Allocation can be classified into three phases [15,17,21]:-

Resources discovery represents the first phase, where the Grid scheduler should continuously monitor the status of the participated resources to update its resource availability database to know which resources are free to serve the upcoming requests. This phase is a challenge in its own rights and needs a specific algorithm to cover it. Many algorithms have been proposed to deal with this phase, for example "Scalable Grid Resource Discovery through Distributed Search" has been introduced by Butt et al. [22] and "semantic based scalable decentralized grid resources discovery framework" was proposed by Hassan et al. [23].

The second phase is resources selection, where Grids are usually composed of heterogeneous resources which are completely varied in their number of cores, computation speed and memory size. Therefore, the resource selection phase is focusing on selecting the best resource among the available ones to serve the incoming request. Each request specifies the amount of computing power it needs and for how long.

The third phase is focusing on resources usage; after allocating a suitable resource for the incoming task, the task will then be sent to the selected resource over a wide area network. The resource will be reserved to serve this task for the specified amount of time and it could not be used to host another request during this allocated period.

Resources selection is the critical phase in the whole process. For example, if the incoming requested task needs 3 cores CPU for 3 hours and it was hosted in 4 CPU cores node, one un-needed CPU core would run for 3 hours which affects the total Resource Utilization [24]. This leads to lose much of computation power and energy. , where and represent the number of used processors and total processors on resource respectively.

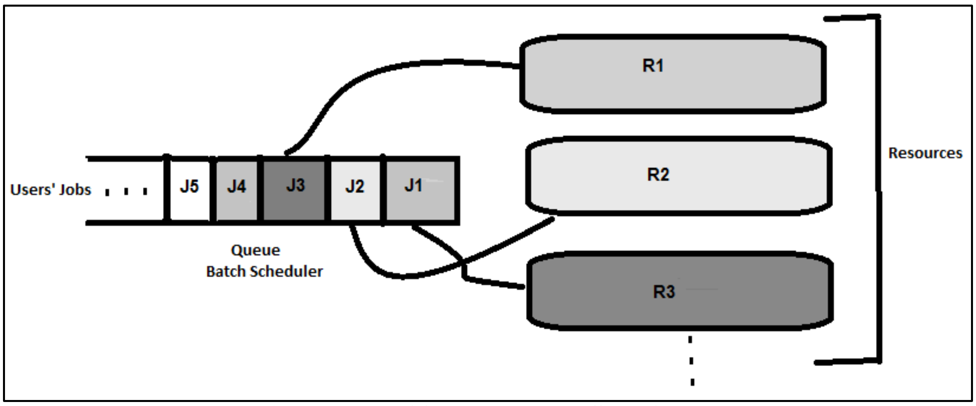

Figure 1. Grid Resources Allocation Process

3.2. Resources Allocation in Cloud Technology

Resource allocation and provisioning is the process of mapping the available resources to different applications of the Cloud over the Internet. It must be managed precisely in a way that prevents overloading and resources wastage (wastage of bandwidth, processing unit, memory, etc.). The service provider is responsible for managing its resources, while the Cloud's users are responsible for specifying their application requirements and the Service Level Agreement (SLA) [13],[24],[25]. Cloud resource allocation can be identified by four phases. Figure 2 illustrates abstracted resources allocation process:

Phase 1: Virtual Machine (VM) Creation Each VM is created according to user's request, which is usually specified in SLA document. Given that Cloud could accept hundreds of users' requests simultaneously, hundreds of VMs can be created simultaneously too.

Phase 2: VM Deploying This is the process of mapping a VM to a suitable available physical machine. The scheduler should have detailed information about the physical machines: which machines are overloaded and which ones are able to host new VMs. Each VM will be accommodated by one host (Physical Machine (PM)). This process is similar to resource selection in Grid resource allocation with extra difficulties. Many VMs can be deployed to one PM. Therefore, the deploying decision should be handled in a way that maintains good resources utilization and avoid performance degradation.

Phase 3: Task Encapsulation this is the process of encapsulating tasks to be processed into the VM. In reality, the VM may be allocated for a specific user or it can be used for multiple users depending on Cloud service level. User task should be encapsulated in its VM. This commonly happens in the Cloud models of Infrastructures-as-a-Service (IaaS) and Platform-as-a-Service (PaaS). Pre-created VMs can be used to serve user incoming tasks, which are usually taking place in the Software-as-a-Service (SaaS) model. Generally, Cloud applications are executed on VMs. Each application has specified requirements of processing power, and the created VM must provide the processing power to the application.

Phase 4: VM Usage In this phase VM will be running to process the encapsulated tasks.

Public, private and hybrid Clouds all have the ability to provide VMs with unlimited scalability. The cost of purchasing, operating and maintaining the physical resources of these Cloud datacenters is enormous [26]. Thus Cloud providers intend to optimize the usage of these resources, through planned allocation of VMs in hosts. Cloud service providers are also keen to maintain a positive client experience. The difficulty of resources allocation problem comes in twofold: maintaining the performance requirement of VMs on one side while reducing the operational cost on the other side [26], [27]. An optimal resource allocation should avoid the following criteria [19,20,27]:

- Resources contention is the situation when two or more VMs try to use the same resources at the same time.
- Scarcity of resources arises when the available resources are limited.
- Resources fragmentation is the situation where the resources are isolated.
- Over provisioning arises when the application acquires surplus resources more than the required one.
- Under-provisioning of resources happens when the application is allocated with fewer numbers of resources than the demands.

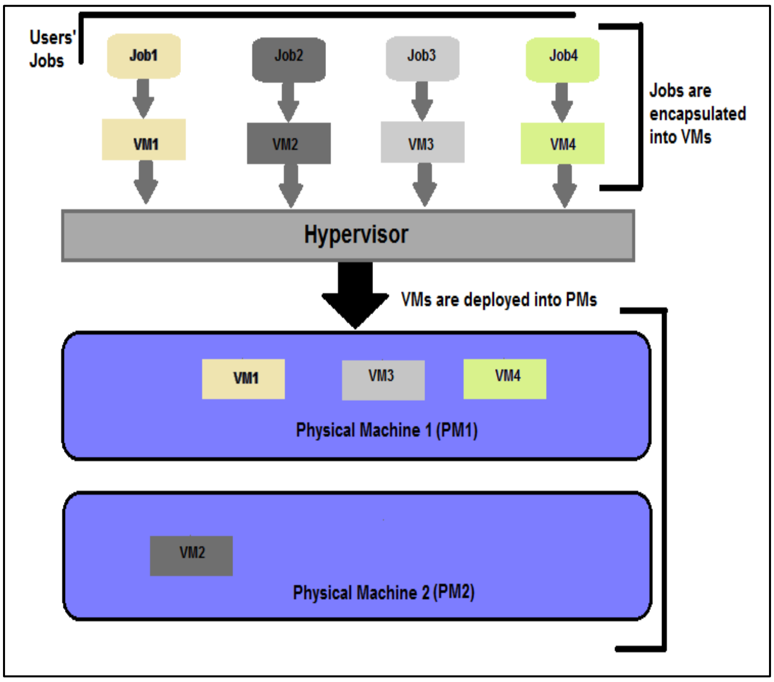

Figure 2. Cloud Resources Allocation Process

4. Comparison of Resources Allocation Problems Identified on Grid and Cloud Computing

In both Cloud and Grid models, there are a number of factors connected to resource allocation problem such as the level of virtualization, the applications hosted, scheduling policies and some of special associative problems. Table 1 summaries all these factors. This section provides a comprehensive classification of all associated problems and surveys the existing the techniques to deal with these issues.

4.1. Full virtualization

Virtualization plays a crucial role in management resources of Cloud platform; given that Cloud computing is based on full virtualization technology where a single PM is able to host several VMs which are completely isolated. Each VM has its own virtual processor, memory, and its own operating system which is called guest operating system [28]. The virtualization process is performed through a Hypervisor. Renowned implementations are e.g. Xen, Kernel based Virtual Machine (KVM), Open Nebula and VMWare [26,29]. A Hypervisor is dedicated software running on top of the physical hardware enabling to create virtual environment to operate virtual machines. The reason behind using fully virtualized hosts in Cloud is to get efficient hardware utilization, lower energy consumption, and improve fault tolerance [4,25,26,27,30]. However, VMs are provisioned in two ways: direct provisioning in which the computing power (VM) provides directly to the customers; this usually is the provisioning method in an IaaS, and indirect provisioning which is carried by wrapping the provisioned applications in a VM this case is widely used in Software as-a-Service (SaaS) and PaaS. Popular Clouds use different resource allocation strategies: greedy first fit and rotating scheduling are used e.g. in Eucalyptus [4].

On the other hand, full virtualization concept has not been used in Gird systems yet. Each individual organization in Grid maintains full control of their resources. However, dedicated resources allocation is a likely strategy for the job hosting as illustrated in figure1, scalable provisioning or unplanned increase for the job's requirements are not allowed in Grid [1,2].

4.2. Scheduling Strategies

Batch scheduler is the well-known scheduler in GC specifically in computational Grid in which the LRM such as Portable Batch System (PBS) is a job resource manager in a batch cluster environment [31], [32]. Condor [33] is governing resources by using a queueing system. Bouyer et al. (2013) in [34] developed a novel queuing algorithm for local Grid scheduling called market-oriented job scheduling and applied it to the grid scheduling problem. This algorithm is different from the previous queuing algorithms such as First Come First Serve (FCFC), Round Robin (RR), and Back Filling. The proposed algorithm uses the coming tasks' requirements and restrictions like minimum completion time (MCT), reliable completion time), and minimum execution cost to make the queue.

According to Section 2.2, mapping services with complex computing requirements to physical machines with specified capacities is not a trivial issue. It is referred to as a classic vector (or multi-capacity) bin packing problem (VBPP) [30,35]. Each dimension of the problem is corresponding to a type of resources such as CPU, RAM memory, disk space and bandwidth. Several approaches have been used to solve this problem like approximation algorithms (FF, FFD, PP) [35], [36] or Evolutionary Algorithms for Examples Genetic Algorithms (GA) and Ant Colony Optimisation [37]. Haizea is used as VM based lease management architecture scheduler in Open Nebula [29], and PBS (Portable Batch System) and SGN (Sun Grid Engine) are the scheduler of Nimbus [4,16].

4.3. Application Models

Recently Grid Systems have been developed far beyond their original intention. Modern GC can be applied to serve various complex applications, for example collaborative engineering, data processing and exploration, and e-Science applications. Due to the expensive scheduling decisions, data staging in and out, and potentially long queue times, many Grids do not natively support interactive applications. However, there are efforts in the Grid community to enable lower latencies to resources via multi-level scheduling, to allow applications with many short-running tasks to execute efficiently on Grids [12], [11,19].

Clouds are able to host a range of applications, including interactive applications and patch processing applications, such as HPC jobs, scientific workflow, or periodical jobs [13]. Highly available applications are the applications within high availability rate about 99.9999% (availability is the ratio between the hosted requests and the total number of required requests). They also comprise several components such as (web server, communication server, and data store). The resources allocation within the highly available applications needs extra efforts to prevent resources failure [16].

4.4. *Resources Consolidation and- VM migration*

Dynamic Server Consolidation (DSC) is a special problem which derives from virtualized hosted platforms like Cloud platform. It aims at consolidating maximum number of VMs onto minimum number of physical machines (PMs), thus improving resources utilization [20,38]. DSC is accompanied by over-provisioning and under-provisioning problems due to the Cloud characteristic of dynamic resources provisioning (elastic workloads). In case of workload changes, the consolidation decision will be modified through live migration of VMs. If a server (PM) is underutilized, all VMs running in this server should be migrated to another server then the underutilized server can be switched to low level mode. If workloads have been increased by Cloud users thus overloading the hosted server, then the migration decision should be done to avoid overprovisioning problem which may lead to performance degradation.

Consolidation decision can be taken in two ways: long term consolidation decision and short term consolidation decision. The short term decision was used in [20] and [39]. This decision is taken without considering the historical information of nodes' load which may lead to a wrong decision in terms of high migration cost. The authors of [24] employed the Markov Decision Process (MDP) to generate long term precise migration decision which aims to improve the profit by avoiding the wrong decisions which may have an adverse effect on the total profit. By comparing the performance of the proposed MDP algorithm and the existing Dynamic Management Algorithm (DMA) the conclusion is driven that the migration decisions of MDP policy are more valuable than the DMA decisions, thus producing better profit.

An integral thermal and compute controlled heuristic approach was introduced in [40] to improve the energy consumption of IaaS Cloud. The proposed approach integrates several of IaaS cloud characteristics such as server heterogeneity, workload dynamicity, VM migration, server processor SLEEP state, server processor state transition power and latency time overheads to configure a new metric called StaticPPMMax that is used for the VM allocation algorithm. The results show that the StaticPPMMax allocation metric and allocation heuristic approach improves the datacenter energy consumption savings by 12% in comparison to an independent thermal and compute controlled scheme. Considering the server processor SLEEP state and transition power and time latencies increase the datacenter energy savings (around 0.5% to 1%).

Grids use a different allocation policy (see Section III.A) that does not enable simultaneous time and space resources sharing. Accordingly, no resource consolidation problem has been introduced in Grid resource allocation yet.

4.5. Load Balancing

Efficient load balancing is one of the major challenges in Cloud resources allocation process. Basically, it is the process of assigning the load to different nodes in such a way which maintains no overloaded or under-loaded nodes. Load balancing aims to improve system performance substantially and maintain stability of the system. Many algorithms were proposed to tackle this problem for example Biased Random Sampling [41], Honeybee Foraging Algorithm, and ACCLB [42].

Some of these algorithms deal with static load whereas others consider dynamic load. In static environments, scheduling decision is made before submitting the jobs; this approach assumes that all the information about resources is known in advance and that resources properties do not vary over time. The main disadvantage of this is the danger of nodes failure. Round Robin and static greedy are used in Eucalyptus to solve static load balancing problem [9].

In dynamic environment, the decisions are taken while the job is being executed. This approach takes into account the dynamism of the cloud and it is generally preferred as resources properties are flexible and may change suddenly. The advantage of the dynamic load balancing over the static one is that the system does not need to be aware of the run-time behavior of the job before execution. LBMM (Load balancing Min Min), ACCLB (Ant Colony and Complex Network Load Balancing), and OLB (Opportunistic Load Balancing) [43] are famous dynamic load balancing algorithms. Based on OBL and LBMM the Three Level load balancing algorithm has been developed in [44] with the aim of getting better executing efficiency and preserve load balancing.

Grid resource allocation does not suffer from load balancing as it is based on reserving physical node for a specified amount of time. Space sharing is not allowed in Grid resource allocation which prohibits load balancing problems.

4.6. Energy Consumption

Energy-aware management of large-scale distributed systems has attracted significant attention by researchers in the last decade. Datacenters are considered as one of main environmental danger causes.

There was 63% increase in power consumed by datacenters for only one year between 2011 and 2012, and it was around 38 Giga Watt in 2012 [45]. The low-power hardware techniques that have been used to reduce the energy consumption are not enough to handle this problem. They must be accompanied by intelligent decision making, resource management and task scheduling mechanisms [16],[26].

Developing an efficient scheduler in both distributed systems (Cloud and Grid) is a challenging issue if energy optimization is to be considered as a scheduling objective. Shu et al. in [46] improved immune clonal selection algorithm to design Cloud and Grid schedulers with a goal of simultaneous optimization of the energy utilization. ACO was used by Suphalakshmi in [45] to reduce energy consumption based on VMs allocation. The proposed approach considers the expected behavior of each user requests even dynamic user requests to make the resources allocation decision instead of using the actual user requests. The resources allocation decision aims to minimizing the number of used servers therefore; reducing energy consumption associated with that decision.

Energy consumption, in a nutshell, is an active researching field in Cloud and Gird resource management. Ramani and Bohara in [47] defined a temperature threshold of the hosts in Cloud data center. It reduces the consumption of the maximum resources and controls the processor temperature at the same time. The proposed approach leads to minimize the overall energy consumption.

4.7. *Performance*

The extensive use of virtualization in Cloud computing has exaggerated the performance problem inside the Cloud model which adds extra challenges for resource allocation and scheduling [48]. Virtualization has had significant performance losses for some applications, which is considered as one of the primary disadvantages of using virtualization. Cloud resources manager should ensure covering the application SLA requirements which is usually done by efficient scheduling. Although performance problem is affecting tangibly in Cloud computing model, it has not been considered widely by Cloud scheduling researches. Maintaining a good Quality of Service level is widely considered in Cloud Resources Allocation research [48], [49].

Grid computing seems to supply better performance than Cloud computing, Grid performance problem has been extensively studied by a great number of Grid resources management studies. However, there is a lack of studies that compare Grid and Cloud performance.

Table 1. Comparison of Grid and Cloud Resources Allocation

Resources Allocation Aspects and Issues	Grid Technology		Cloud Technology	
Full Virtualisation	Not found		Xen KVM Open Nebula VMWare	
Scheduling strategy	Batch Scheduling	Market-Oriented RR FCFS	Bin Packing	FF FFD GA ACO
Application Models	E-Science applications HPC Jobs Scientific workflow		Interactive applications HPC Jobs Highly available applications	
Load Balancing	Not found		Static Round Robin static greedy	Dynamic LBMM ACCLB OLB

Resource Consolidation	Not found			Long term decision	Short term decision
				MDP StaticPPMMax	DMA
Energy Consumption	Immune Clonal Selection			Immune Clonal Selection ACO Temperature Threshold	
Performance	Grid resources management			Good Quality of Service level SLA	
Multi-Site Administration	Global	Inter-site level	Intra-site level	Federated Clouds	
	Hierarchical multi- level	Resource allocation decisions and Scheduling	Resource allocation decisions and Scheduling		

4.8. Multi- Site Administration

The global grid management system is defined as cooperation between the centralized and decentralized resources and service management systems; therefore, the scheduling and resource allocation decisions are delivered either at global, inter-site or intra-site levels [16], [19].

The Grid is known to have a multi-site administration problem since it comprises various collaborated virtual organization each one has its own policy to schedule its resources by using a specific scheduler. Huge efforts have been made to solve this problem; a hierarchical multi- level scheduling is one of multi-administration solution in which the global Grid management system can be defined as a compromise between the centralized and decentralized resources and service management systems, where the scheduling and resource allocation decisions are provided at global, inter-site and intra-site levels [50].

Large scale Clouds (it is also referred to as Federated Clouds) are also suffering from multi-administration problem as they consist of different Cloud services provider with complete resource management policy. This problem resulted in adding a new challenge to large scale Cloud resource allocation and scheduling. Research results in this field are quite similar to Grid resources allocation; however, they disregard Cloud resources management problems and focused on the unification of collaborated providers.

5. Conclusions

This paper presents a thorough comparison of two distributed system models - grid and cloud - in terms of the resource allocation problem, as the two computing models often get confused due to their similar conceptual properties. Resource Allocation plays a crucial role in both models as it directly affects their performance in respects to resources utilisation, energy consumption, and/or load balancing. The main findings of this paper are:

Firstly, the resource allocation problem is completely different in both models in terms of associated challenges, scheduling algorithms, and deploying strategies. In Cloud Computing, the applications are hosted indirectly, i.e. first the applications are allocated to VMs and then the VMs get deployed into physical resources; this strategy enables simultaneous time and space sharing of the underlying resources. This feature allows latency sensitive applications to operate efficiently on Cloud. Whereas, Grid Computing resources are directly and independently deployed to the coming applications, but the deployment decision is also restricted by time and space requirements.

Secondly, the comparison shows that Cloud resources are rented under full virtualisation concept which leads to improve the overall resources utilisation but it adds extra challenges in making the right allocation decision. It also indicates that both resources consolidation and load balancing are major challenges related to single Cloud resources allocation and not found in Grid resources allocation. On the other hand, Grid resources allocation suffers from the multi-site administration problem; wrong resources may be allocated with high energy consumption or low utilisation.

Thirdly, there is notable lack of performance analysis and fault tolerance research in Cloud resources allocation; ensuring that a good level of QoS is delivering to the end users is likely to be one of the major challenges in Cloud Computing since the Cloud is applicable to scalable user demands and dynamic providing scheme.

Fourthly, Clouds come in two types namely Single Cloud and Federated Cloud. Research of resource allocation and scheduling in single Cloud environment is different from that in Federated Cloud. In the Federated Cloud the major resources allocation challenge is multi-site administration.

6. References

[1] Foster, Y. Zhao, I. Raicu, and S. Lu, "Cloud Computing and Grid Computing 360-Degree Compared," Grid Computing Environments Workshop, IEEE, Austin, TX, pp:1–10, 2008. DOI: 10.1109/GCE.2008.4738445.

[2] N. Sadashiv, "Cluster, Grid and Cloud Computing: A Detailed Comparison," 6th International Conference on Computer Science & Education (ICCSE), pp:477 – 482, Singapore, , 2011.DOI: 10.1109/ICCSE.2011.6028683.

[3] S. Di, D. Kondo, and W. Cirne, "Characterization and comparison of cloud versus grid workloads," Proc. - 2012 IEEE International Conference in Cluster Computing, pp. 230–238, 2012.

[4] K. Radha, B. Rao, S. Babu, K. Rao, V. Reddy, and P.Saikiran, "Allocation of Resources and Scheduling in Cloud Computing with Cloud Migration", *International Journal of Applied Engineering Research* ISSN 0973-4562, vol 9 (19) 2014. [Online]. http://news.bbc.co.uk/1/hi/world/americas/4808342.stm

[5] A. Andreazza, A. Annovi, and D. Barberis, "Computing infrastructure for ATLAS data analysis in the Italian Grid cloud," J. Phys. Conf. Ser., vol. 331, no. 5, p. 052001, 2011.

[6] Z. A. Mann, "A Taxonomy for the Virtual Machine Allocation Problem*", *International Journal of Mathematical Models and Methods in Applied Sciences,* 2015, volume 9, pp: 269-276.

[7] M. Hu and B. Veeravalli, "Requirement-aware scheduling of bag-of-tasks applications on grids with dynamic resilience," IEEE Trans. Comput., vol. 62, no. 10, pp. 2108–2114, 2013.

[8] C. Weng and X. Lu, "Heuristic scheduling for bag-of-tasks applications in combination with QoS in the computational grid," Futur. Gener. Comput. Syst., vol. 21, no. 2, pp. 271–280, 2005.

[9] I. Brandic and S. Dustdar, "Grid vs Cloud — A Technology Comparison," it - Information Technolgy., vol. 53, no. 4, pp. 173–179, 2011.

[10] J. Kolodzieja,, S.Khanb , and L. Wangc , "Security, Energy, and Performance-aware Resources Alocation Mechanisms for Computational Grids", Future Generation and Computer Systems, pp:77-92 , 2014.

[11] R. Buyya and M. Murshed, "GridSim: A Toolkit for the Modeling and Simulation of Distributed Resource Management and Scheduling for Grid Computing Concurrency and Computation: Practice and Experience. Vol.14, pp:1175–1220, 2002. DOI: 10.1002/cpe.710.

[12] M. Frincu, "Adaptive Scheduling for Distributed Systems", Ph.D. thesis, West University of Timisoara, Faculty of Mathematics and Computer Science, May 2011.

[13] E. Carlini, M. Coppola , P. Dazzi ,L. Ricci , and G. Righetti, “Cloud Federations in Contrail”, chapter in a book “Euro- Par 2011: Parallel Processing Workshops”, part 1, Sipringer, LNCS 7155, pp. 159–168, 2012.

[14] W. Tian, X. Liu, C. Jin, and Y. Zhong, “LIF: A Dynamic Scheduling Algorithm for Cloud Data Centers Considering Multi-dimensional Resources*”, *Journal of Information & Computational Science*, vol.10, pp:3925–3937, 2013.

[15] E. Amiri, H. Keshavarz, N. Ohshima, and S. Komaki, “Resource Allocation in Grid: A Review,” Procedia - Social Behavior Sciences, vol. 129, pp. 436–440, 2014.

[16] B. Malet and P. Pietzuch, “Resource allocation across multiple cloud data centres”, 8th International Workshop on Middleware for Grids, Clouds and e-Science, Copyright 2010 ACM 978-1-4503-0453-5, 2010. DIO:10.1145/1890799.1890804.

[17] A. Yousif, A. Abdullah, M. S. Abd Latiff, and M. B. Bashir, “A Taxonomy of Grid Resource Selection Mechanisms”, *International Journal of Grid and Distributed Computing,* vol. 4, no.3, pp. 107-118, September 2011.

[18] K. Etminani and M. Naghibzadeh, “A Min-Min Max-Min selective algorithm for grid task scheduling,” 2007 3rd IEEEIFIP Int. Conf. Cent. Asia Internet, pp. 1–7, 2007.

[19] M. Zolt and E. February, “Allocation of Virtual Machines in Cloud Data Centers – A Survey of Problem Models and Optimization Algorithms”, ACM Computing Surveys (CSUR), vol.48, no.1, 2015,
DOI:10.1145/2797211

[20] F. Hermenier, X. Lorca, J.-M. Menaud, G. Muller, and J. Lawall, “Entropy: a Consolidation Manager for Clusters,” Proc. 2009 ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environments, no. 41, pp: 41-50, 2009.

[21] A. A. Haruna., N. B. Zakaria and N.Haron, “Grid Resource Allocation: A Review*”. Research Journal of Information Technology*, vol. 4(2), pp: 38-55, 2012.

[22] Butt, F., Bokhari, S.S., Abhari, A., and Ferworn, “A Scalable Grid Resource Discovery through Distributed Search”, *International Journal of Distributed and Parallel Systems*, pp: 1-19, 2011.

[23] M. Hassan and A. Abdullah, “A New Grid Resource Discovery Framework,” *the International Arab Journal of Information Technology*, vol. 8, no. 1, pp. 99-107, 2011.

[24] M. E. Frincu, “Scheduling highly available applications on cloud environments,” Future Generation and Computer Systems, vol. 32, no. 1, pp. 138–153, 2014.

[25] SI. Version, D. Sivapriyanka, and S. Santhanalakshmi, “Allocation of Resources Dynamically In Cloud Systems”, *International Journal of Engineering Research and Applications*, vol. 4, no. 5, pp. 113–118, 2014.

[26] A. Hameed, A. Khoshkbarforoushha, R. Ranjan, and P. P. Jayaraman, “A Survey and Taxonomy on Energy Efficient Resource Allocation Techniques for Cloud Computing Systems,” “Computing”, Springer, pp.1-24, 2014.

[27] K. Radha, B. Thirumala Rao, S. M. Babu, K. Thirupathi Rao, V. Krishna Reddy, and P. Saikiran, “Allocation of Resources and Scheduling in Cloud Computing with Cloud Migration,” *International Journal of Applied Engineering Research.,* vol. 9, no. 19, pp. 5827–5837, 2014.

[28] A. S. Tanenbaum, Modern Operating Systems, 3rd edition, Person Prentice Hall, US, 2009.

[29] S. S. Manvi and G. Krishna, “Resource Management for Infrastructure as a Service (IaaS) in cloud computing : A survey,” *Journal of Network and Computer Applications*., vol. 41, pp. 1–17, 2013.

[30] A. Alahmadi, A. Alnowiser, M. M. Zhu, D. Che, and P. Ghodous, “Enhanced First-Fit

Decreasing Algorithm for Energy-aware Job Scheduling in Cloud,” Proc. - 2014 Int. Conf. Comput. Sci. Comput. Intell. CSCI 2014, vol. 2, pp. 69–74, 2014.

[31] R.L. Henderson. “Job Scheduling under the Portable Batch System” In D. Feitelson and L. Rudolph, editors, Job Scheduling Strategies for Parallel Processing (Pro- ceedings of the 1st International JSSPP Workshop; LNCS no. 949), pp:178– 186. Springer-Verlag, 1995.

[32] Research computing and cyber infrastructure, http://rcc.its.psu.edu/user_guides/system_utilities/pbs/, accessed August 07, 2011.

[33] M. Litzkow, M. Livny, M. Mutka, “Condor – A Hunter of Idle Workstations”, in: 8th International Conference of Distributed Computing Systems, pp:104–111, June 1988.

[34] A. Bouyer, B. Arasteh, F. Nasrollahi, “Maximizing Job Submission Rate in Market-Oriented Grids”, AWERProcedia,Information Technology & Computer Science, vol. 3, pp:1877-1885, 2013. http://www.world-education-center.org/index.php/P-ITCS.

[35] W. Tian, X. Liu, C. Jin, and Y. Zhong, “LIF: A Dynamic Scheduling Algorithm for Cloud Data Centers Considering Multi-dimensional Resources*”, *Journal of Information & Computational Science*, vol.10, no.12, pp:3925–3937, 2013.

[36] A. Alahmadi, A.Alnowiser, M. Zhu, D. Che and P. Ghodous, “Enhanced First-fit Decreasing Algorithm for Energy Aware Job Scheduling in Cloud“, International Conference on Computational Science and Computational Intelligence, vol.2, pp: 69-74, 2014.

[37] K. Chandrasekaran, and U. Divakarla, “Load Balancing Virtual Machine Resources in Cloud Using Genetic Algorithm”, ICCN 2013, pp.156-168, Elsevier2013.

[38] H. Ferdaus and M. Murshed, “Virtual Machine Consolidation in Cloud Data Centers Using ACO Metaheuristic”, Springer, Euro-Par 2014: Parallel Processing: 20th International Conference, pp. 306–317, 2014.

[39] G. Jung , K. Joshi, M. Hiltunen , R. Schlichting, and C. Pu, “A Cost-Sensitive Adaptation Engine for Server Consolidation of Multitier Applications”, chapter in “Middleware 2009”, , pages 163–183, International Federation for Information Processing, 2009.

[40] M. Kumara, and S. Raghunathan, “Heterogeneity and Thermal Aware Adaptive Heuristics for Energy Efficient Consolidation of Virtual Machines in Infrastructure Clouds*”, Journal of Computer and System Sciences*, vol. 82, pp:191–212, 2016.

[41] M. Randles, D. Lamb, A. Taleb-Bendiab, “ A Comparative Study into Distributed Load Balancing Algorithms for Cloud Computing”, IEEE International Conference on Advanced Information Networking and Applications Workshops, IEE Computer Society, pp: 551-556, April 2010.

[42] Al-Jaroodi, and N. Mohamed, “DDFTP: Dual-Direction FTP,” IEEE Computer Society Washington, DC, USA, pp: 504-513, 2011.

[43] X. Ren, R. Lin, and H. Zou, ”A Dynamic Load Balancing Strategy for Cloud Computing Platform based on Exponential Smoothing Forecast”, Cloud Computing and Intelligence Systems IEEE, pp: 220 - 224, 2011.

[44] S. Wang, K. Yan, W. Liao, and S. Wang, “Towards a Load Balancing in a Three-level Cloud Computing Network” Computer Science and Information Technology (ICCSIT), 3rd IEEE International Conference IEEE, vol. 1, pp:108 - 113, 2010.

[45] A. Suphalakshmi and S. M, “An Intelligent, Energy Conserving Load Balancing Algorithm for the Cloud Environment Using Ant’s Stigmergic Behavior”, *International Journal of Communications and Engineering,* vol.4, no. 3, March2012.

[46] W. Shu, W. Wang, and Y. Wang, “A Novel Energy-Efficient Resource Allocation Algorithm based on Immune Clonal Optimization for Green Cloud Computing”, *Journal on Wireless Communications and Networking,* vol. 2014, no. 1, p. 64, 2014.

[47] M. Ramani, and M. Bohara, “Energy Aware Load Balancing In Cloud Computing Using Virtual Machines”, *Journal of Engineering Computers & Applied Sciences*, vol. 4, no.1, pp:1-5 , 2015.

[48] A. Tchernykh, L. Lozano, U. Schwiegelshohn, P. Bouvry, J. E. Pecero, S. Nesmachnow, and A. Y. Drozdov, “Online Bi-Objective Scheduling for IaaS Clouds Ensuring Quality of Service,” *Journal of Grid Computing*., pp. 911–918, 2015.

[49] J. W. Lin, C. H. Chen, and C. Y. Lin, “Integrating QoS Awareness with Virtualization in Cloud Computing Systems for Delays Applications”, Future Generation Computer Systems, vol. 37, pp. 478–487, 2014.

[50] N. Bessis, S. Sotiriadis, V. Cristea, and F. Pop, “Modelling Requirements for Enabling Meta-Scheduling in Inter-Clouds and Iinter-enterprises”, Third International Conference on Intelligent Networking and Collaborative Systems, pp. 149–156, 2011.

Acknowledgment

This work has been funded by the Ministry of Higher Education and Scientific Research of Iraq. The authors gratefully acknowledge this support.