Dynamic Consolidation of Virtual Machines with Multi-Agent System

Esha Barlaskar
Dept. of Computer Science & Engineering and I.T,
Don Bosco College of Engineering and Technology,
Assam Don Bosco University
Guwahati, Assam,India

Y. Jayanta Singh
Dept. of Computer Science & Engineering and IT,
Don Bosco College of Engineering and Technology,
Assam Don Bosco University
Guwahati, Assam,India

ABSTRACT
Cloud computing systems supply massive infrastructures for high-performance computing that are flexible as they are able to acclimate to user and application requirements. Cloud Computing offers on demand services which are used by virtue of a service-oriented interface that execute the anything-as-a-service archetype. The enlargement of Cloud computing has resulted in the establishment of large-scale data centers around the world containing thousands of compute nodes and these data centers consume excessive amounts of electrical energy resulting in high operating costs. Therefore, to cut down the cost of energy consumption the cloud providers must optimize resource usage by performing dynamic consolidation of virtual machines (VMs) in an effective way to improve energy efficiency in cloud data center. The problem of VM consolidation can be split into four sub-problems namely physical machine overload detection; physical machine under-load detection; VM selection and VM placement. Each of the afore-stated sub-problems must operate in an optimized manner to maintain the tradeoff between energy and performance. In this research paper a new multi-agent system (MAS) for dynamic consolidation of VMs is proposed with the aim of making the cloud system smarter by incorporating the five traits of multi-agent systems which are ubiquity, interconnection, intelligence, delegation and human orientation. The Cloud Computing systems require intelligent and perceptive based software with progressive, elastic, self-ruling style which can be provided by MASs. The proposed method has significantly reduced energy consumption and at the same time ensures a high level of constancy to the Service Level Agreements (SLA).

General Terms
Cloud Computing, Multi-Agents

Keywords
Dynamic consolidation, energy efficiency, multi-agent system, virtual machine placement

1. INTRODUCTION
Agent systems constitute in its own a complete and independent software programs exhibiting realm knowledge. Agents have the capability to function independently to a certain extent to execute actions required to attain stated goals in a dynamic environment. Generally all agents comprise a set of lineaments which are as described in the following:

• Autonomy: the ability to act independently to some extent on behalf of other programs and users by altering the approach in which they attain their objectives.

• Pro-activity: the ability to follow their own distinctive set goals by constructing decision according to the internal decisions.

• Re-activity: the ability to respond to extrinsic occurrence and provocation and thereupon making decisions to execute their tasks by adapting their behavior to the sudden new changes.

• Communication and Cooperation: the ability to connect and collaborate with other agents (in multi agent systems) so as to exchange message, accept instructions and contribute answers and coordinating in order to fulfill their own goals.

• Negotiation: the ability to execute conversations in a systematized form to obtain intensity in participation and collaboration with other agents.

• Learning: the capability to become better over time by enhancing performance and decision making while communicating with the extraneous surroundings [1].

Even though a single agent can operate and manage to achieve an inclined task, however the agent archetype was created as a distributed reckoning model where a number of agents communicate with one another by interchanging information and cooperating to accomplish convoluted tasks where synergy, agility, rework and dynamicity are pivotal concerns to be taken care of. In order to take the full advantage of the agent archetype it is required to acknowledge agents as entities working in a group of agents thus putting into action the actual multi-agent system(MAS) paradigm, therefore will work much efficiently compared to an agent which is working in isolation. As a result of which it is rather challenging to envision that an agent will survive as a stand-alone entity without communicating with other agents in its surroundings. However, there are some stand-alone agents such as personal or information agents that work in exclusion to solve their problems, but it is seen that even such agents can improve their performance to a great extent if they work in unison with other agents. Henceforth, one of the most crucial features of agents is its social dimension. The main traits of MASs stated in [2] are as follows:

A single agent has insufficient knowledge or potential for resolving complicated issues. Therefore, each agent’s way of thinking is inadequate on the global task to be accomplished;

i. Global mastery over the system is not present;

ii. Disperse data are there; and

iii. Asynchronous computations are there.

The above mentioned traits are of distributed computing paradigms and as such MASs also belongs to distributed paradigms. Hence, MASs have many common characteristics with the other existing distributed paradigms like Grid Computing, Cloud Computing, Peer-to-Peer networks, Sensor networks, Autonomic computing etc. However, MASs have some unique set of lineaments, as mentioned before, which make it distinct from other distributed computing models.

In cloud computing the customers can provision resources on demand in a pay-as-you-go manner over the Internet, therefore, service providers without making any advance investment in infrastructure could simply rent resources from infrastructure providers to whatever it needs and pay for only its own usages and hence the cost for maintaining the infrastructure is omitted. But at the side of infrastructure providers the cost for maintenance increases and also results in high energy consumption. However, the energy consumption in cloud data centers could be minimized by the application of virtualization technology as it allows using of fewer physical servers with much higher per-server utilization but at the same time it harbingers new challenges for the management of virtual machine (VM). Thus, VM management in cloud data centers must be provisioned and managed very productively and hence must pave the way for optimizing the energy-performance trade-off. In order to manage the resources by making it energy efficient and also to improve the quality of service based on SLA in a cloud-computing environment the dynamic assignment of physical resources to virtual machines must be done very effectively by doing the mapping of virtual machines to physical machines in an intelligent way. Researchers in this filed have been trying to design an optimal management system that can be able to satisfy these constraints. Dynamic VM consolidation as a dynamic control procedure is an effective management system to improve energy efficiency in a cloud data center [16]. In general, the problem of dynamic VM consolidation can be split into four sub-problems (1) determining when a physical machine is considered as being overloaded (physical machine overloading detection), in this situation live migration is required to migrate one or more VMs from the overloaded physical machine; (2) determining when a physical machine is considered as being under-loaded (physical machine under-loading detection), in this situation the physical machine is ready for switching to sleep mode, thus, all VMs have to migrate from it; (3) determining which VMs must be selected to migrate from overloaded physical machine (VM selection) and (4) determining which physical machines must be selected to place migrated VMs (VM placement). The goal of this procedure is to operate in a way that optimizes energy-performance tradeoff inside cloud data center. To this end, each of afore-stated sub-problems must operate in an optimized manner. In this paper a multi-agent system was proposed for performing the dynamic consolidation of VMs. The whip hand of the proposed model is that it will minimizes the time for which the physical machine remain overloaded and as such maximizes the performance and reduces energy consumption.

In this paper the contributions of MASs is discussed by conferring their applications in Cloud Computing, in particular and also proposed MAS for dynamic consolidation of VMs. The rest of the paper is arranged as follows. Section 2 describes briefly about Cloud Computing and its integration with MAS. In Section 3, literature survey is presented followed by Section 4 which describes the proposed model of MAS for dynamic consolidation of Virtual Machine (VM) in cloud data center. In Section 5 metrics used for measuring energy consumption and SLA violation are explained and also the simulation tools to be used are described. Section 6 demonstrates the experimental results and finally Section 7 concludes the paper.

2. BRIEF ABOUT CLOUD COMPUTING AND ITS INTEGRATION WITH MAS

The definition provided by the National Institute of Standards and Technology (NIST) have a complete reference [3][2] which goes like this: “Cloud computing is a pay-per-use model for enabling available, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.”. Furthermore, NIST also stated that: “Cloud model promotes availability and is comprised of five key characteristics, three delivery models, and four deployment models”. However, the definition of Cloud Computing has been assorted from the time it came into existence. Some definitions aim attention at virtualization technology (VT) which is the key factor in Cloud, others aim at dynamic resource provisioning which is on-demand and pay-as-you-go model policies. While some other focus attention on service-oriented interface and many more. Thus, it can be stated that Cloud support flexible services, high performance and extensible storage of data to a huge and ever increasing number of users [4]. Cloud has magnified the field of distributed computing systems by supplying progressive Internet services that enhance and complete the functionalities offered by other distributed computing paradigm. Indeed, Cloud supplies large-scale infrastructures that are dynamically readjusted to user and application needs which are typically for high-performance computing. Clouds can handle highly accelerated computing workloads by providing very huge data storage abilities, which potentially helps in maintaining the goal of reducing the usage and management cost.

On the other hand, another distributed computing paradigm includes MAS consisting of number of agents that interact with each other in a very intelligent manner. MASs, being part of distributed computing paradigm, can fix issues by subsidizing solutions with the coordinating and cooperating behavior of several agents. Software agents have got many special features and one of the most important aspects is the intelligence that is encapsulated into them according to some composite artificial intelligence approach that can operate on distributed or parallel computers to attain the required high performance solutions with less time for executing complicated problems. Even though there are many differences between MASs and Cloud Computing but since both belong to distributed computing models, hence some common issues can be realized whereupon benefits can be achieved by the incorporation of MASs and Cloud. In [1] an attempt was made to highlight the scopes of cloud meeting agents wherein the research activities of both Cloud and MASs were discussed. It was mentioned that the research movements in the field of Cloud Computing were mainly concentrated on the profitable use of computing infrastructure, storage of data, extensible virtualization techniques, service delivery and energy optimization. On the other side, MASs research movements were aimed at the intelligent traits of agents and their use with respect to simulation of complicated systems, robust systems, software-exhaustive applications, decentralized computational intelligence and composite learning. However, MASs and Cloud have some common research problems that overlap and thus cloud can supply a very persuasive, trustworthy, stable, easy to foretell and extensible computing infrastructure for the beheading of MASs implementing agent-based complex applications by providing simulation and modeling of such complex systems. Whereas on the other end, MASs can be implemented as elementary components for executing intelligence in Cloud systems making them autonomic in resource management, provisioning of services, adaptive and capable of running massive applications

In Cloud Computing innovative techniques must be integrated for developers and users to request, explore, convene and use computing resources. Adaptive and self-governing agents and MASs are appropriate tools for bargaining user access, for the resource and service detection automation and configuration, transaction, exploiting of Cloud resources. Henceforth, a new paradigm of agent-based Cloud computing can be established for supplying agent-oriented solutions organized on the design and growth of software agents for the advancement in managing and discovering Cloud services and resources, service level agreement (SLA) negotiation and composition of services. In huge Cloud data centers, autonomous agents can hunt, filter, inquiry and revise the enormous amount of stored data. Cloud agents can work independently to supply intelligent services for data accessing, auditing, strategies for automatic assignment of processor to applications, and most importantly the usage of Cloud infrastructure in an energy-efficient manner. Many researchers have implemented agent-based Cloud computing for the three type of services, such as, agents can assist in provisioning resources to customer applications in an intelligent manner for Infrasture-as-a-Service (IaaS), while in Platform-as-a-Service (PaaS) agents can efficiently deploy and run programming framework that are used for application implementation by developers. Lastly, in Software-as-as-Service (SaaS) agents can be programmed to enhance the use of application services that are provided to users by properly managing fundamental software and hardware underlying infrastructure, optimizing the utilization along with the Quality of Service (QoS) preservation. MASs are capable of dealing with dynamic composition, variety, and excitability and therefore can help users and providers to achieve their goals by providing favorable approaches. Finally, trust and security are the most crucial issues in Cloud as users perform everything on machines which are not owned by them and are directly under the control of Cloud providers. Therefore, agent-based algorithms and models for security and trust in Cloud can prove to be very helpful. In brief it can be said that agent-based techniques when integrated in Clouds can make services proactive, autonomic, intelligent and pliable.

3. LITERATURE SURVEY

3.1 A Review on Applications of MASs in Cloud Computing

MASs have been implemented in solving diverse problems related to cloud and few of which are discussed as follows:

In [5] a new scheduling technique by using genetic algorithm was proposed by integrating satisfaction of users with credibility of resources. The objective function was the user satisfaction with resources credibility as a part of the user satisfaction, therefore, differing from other conventional Cloud resource scheduling approaches which enhance either satisfaction of users or efficient execution of systems but never considered both the traits simultaneously. The proposed scheduling strategy was integrated with MASs and agent-based Cloud computing system architecture was also proposed and this approach was validated by performing experiment on the simulation tool CloudSim based on the JACK agent language. The results indicated that this scheduling strategy with the support of MASs improved both user satisfaction and efficiency of system execution.

In [6], the authors stated one of the emerging issues in Cloud Computing which is the availability of social media contents created from users are growing dynamically which requires supplying high-quality media contents in sites of social media. Anyhow, such high quality SNS service in mobile environment is not acquired by users because of the limitation in the performance of the hand device and the explosion of the mobile data. Therefore to overwhelm this issue the authors proposed an intelligent multi-agent model based on virtualization rules for resource virtualization (IMAV) for inevitably allocating service resources convenient for mobile devices. The proposed MASs have the ability to infer the demands made by users by examining and learning the information about user context and also IMAV was capable of allocating service resources according to types of the usage in order to provide reliable service resources utilization by users. This approach was capable of advocating effective virtualization by examining user context and the condition of the system. Further, through the implementation of this model MASs reliable services are provided to the users as the MASs supply suitable services for users based on various situation faced by users.

In [7] a Cloud service discovery system (CSDS) was presented with the target of supporting the users in discovering Internet based Cloud services. The proposed CSDS communicate with Cloud ontology to find the common factor among and between services. This work was an attempt to develop an agent-based discovery system that confers ontology when fetching information about services in Cloud by launching a generic search engine to discover required services. A similar work like [7] was presented in [8] but this approach didn't make use of a generic search engine which consumes more time. Indeed the searching process of cloud services were finished before the submission of user query.

The authors in [8] proposed a framework based on MASs and ontology for Cloud service discovery, where the agents help the users in finding their requested services and fulfilling user's demands very quickly and correctly. A cloud service discovery environment was implemented by building a prototype using JADE in designing the agents for mobile agent platform. The Recall and Precision values were utilized in evaluating the correctness of the proposed MASs model and the results showed the proposed method was capable in efficiently discovering the desired services automatically in Cloud.

In [9] a four-stage agent-based Cloud service discovery protocol was presented. The authors developed MAS that coordinates efficiently by recommending a adaptable ontology-based matching utilizing an ontology-based definition where resources were defined semantically and comparatively to other resources. A database was used to store and maintain record of archival data for making intelligent suggestions by predicting the attribute value. The experimented was performed on a test-bed devised for agent-based Cloud service discovery and by using the described ontology reasoning for finding and matching the users' service requests and the service specifications of providers.

The results show better results when the broker agent in the proposed system make use of Cloud ontology and a connection procedure with a recommendation phase than when it uses only the connection procedure without the Cloud ontology and when it didn't use both. Thus from the empirical result it was proved that the proposed system was very efficient in dealing with enormous amount of information with the help of the recommendation mechanism.

In [10] a cloud intelligent and energy-saving information interface agent with web services was presented where it was seen that agent simply cannot search relevant technologies to build web service platform but can explore the development of cloud interactive diagrams by making use of web service techniques for widely and consistently merging backend information systems on the Internet. This work was the first cloud energy-saving information agent which used web service techniques by presenting a three-stage intelligent decision processing strategy. The initial system development and display function of the maximum contract electrical capacity provided the detailed energy-saving information services. However, final result on energy-saving processing was kept for future work.

In [11] a cloud resource discovery and SLA negotiation was reinforced by building agent-based teastbed system where various jobs were handled by specific agents. Different agents were assigned different tasks such as one agent has the work of finding all cloud computing system that can content the user demand, another agent is responsible to do the negotiation at a particular stage and keeping in mind the users need the most excellent offer given by cloud provider is chosen. The user job is even divided into different parts on various cloud system in order to get improved result.

In [12] agent-based technology for provision and management in Cloud computing was proposed where a new application was build to guide the cloud users in the various stages of accessing cloud services starting with the selection of proper cloud resources, then configuration of cloud and lastly continuous monitoring of the cloud system.

In [13] another model based on agent-based technology was designed to optimize the management of resources in Cloud by aiming at the provider side so that the Cloud providers can make use of a tool which is able to do automatic optimization in resource management. MASs proposed in this work consists of two different types of agents, namely supervisor agent (SA) and data center agent (DCA). The SA is single in number and multiple number of DCA which represent each of the data centers. The aim of all the communication agents was to reduce the response time of users' requests and to appease these constraint reactive agents were used. Therefore, a simply message exchange between SA and DCA was used in a centralized architecture of agents. The work of SA is to deal with the user requests in first come first serve (FCFS) basis and send a message to DCAs to query about the availability of resources. On receiving this message from SA all the DCA will respond by sending their status. After getting the respective responses from all DCAs SA will choose the most appropriate one depending on some predefined rules. However, this work was just a proposal with the design phase and the authors are yet to implement this system to evaluate its performance.

In [14] autonomic cloud management was performed by developing an agent-based architecture. The proposed system contains two separate interacting layers, namely, a low level execution layer and a high level management layer which correlate to the physical and simulation portion of a symbiotic system respectively, where resources and virtual machines were monitored by worker agents and these worker agents were supervised by network management process for taking adaptive actions. In order to achieve the global objectives what-if simulations were conducted by the management processes to upgrade the local rules of the worker agents when required. Therefore, such a supervised decentralized decision making method could lessen the burden on the system management stack by making the resource management much efficient and improve the responses for attacks and failures. A qualitative case study with respect to data center business resilience assurance was presented to support the effectiveness of the proposed model for embellishing the strength of virtual infrastructure.

In [15] a mobile agent-based approach of ensuring trustworthiness in Cloud was presented with three important entities, namely, Cloud Service Provider (CSP), Cloud Broker and Cloud Customer where trust was assured on the basis of prize points, penalties and monitoring of mobile agents. The customers and CSPs which are trustworthy were identified with the help of penalties and prize points and thus the vicious ones were abolished. The Cloud broker's responsibility was to continuously monitor and control the entire system so as to provide trustworthiness in the cloud.

3.2 A Review on Resource Management in Cloud Computing

K. Mukherjee et. al. [16] proposed eco-friendly cloud computing which not only diminished global warming but also minimized operational cost by reducing power consumption. They have introduced noble algorithm (using honey bee and ant colony algorithm) for proper utilization of energy for cloud computing. In cloud computing it is noticed that in IaaS some CPUs are overloaded for handling customer services, some are underutilized and some are fully idle. The energy consumption can be saved by switching these free CPUs off and moving services from overloaded CPUs to under loaded CPUs. Their proposed technique will not only save energy but will also avoid probable SLA violation. But the problem lies in managing these idle CPUs, overutilized CPUs and underutilized CPUs effectively; for this Bee-Ants colony system was proposed by the authors. At the starting, they divide the jobs into two parts; the first part does the service rescheduling, that is, the appropriate management of overloaded and under loaded CPUs whereas the second part manages the idle CPUs, that is, it does power consumption management. Thus, for service rescheduling they proposed bee colony algorithm and for the power consumption management they proposed ant colony algorithm. They have validated their study by a set of tests using Ubuntu's 10.04 Server editions by setting up a small private cloud consisting of two servers. They have emphasized on an eco-friendly cloud computing by turning off the switches of idle CPUs, at the same time they have focused on avoiding SLA violation for reducing energy consumption and functional cost. They have only focused on overload, under-load and idle host detection but didn't consider VM-migration in their work.

The problem in load balancing of virtual machine (VM) resource scheduling in cloud computing environment is that it mostly considers the present state of the system but hardly considers historical data and system fluctuations which leads to load inequality of the system. In connection to this problem, Jinhua et. al. [17] proposed a scheduling strategy of resources based on genetic algorithm which considers historical data and current state of the system and therefore estimate in advance the influence it will have on the system after the arrangement of the required VM resources and then chooses the least-affective solution by which it attains the best load balancing and also minimizes or avoids dynamic migration, hence this strategy solves the problem of load imbalance and huge migration cost .They have proved that this method is able to accomplish load balancing and moderate resource utilization both when system load is balanced and divergent, by experimenting with different workload conditions. However, their strategy is not suitable in situations where the user requirements differ with time, that is, it becomes almost impossible to predict the future requirements if the user keep on changing the type of their requests every time. A possible solution for such situations is to maintain a record of available VMs. Thus, record keeping will make the process of resource allocation much faster by provisioning the VMs based on the cloudlet requirements as it arrives instead of predicting ahead by considering the historical data.

N. Bobroff et. al.[18] proposed a new algorithm for preserving performance. Their algorithm remaps the VM to PM for future resource demand.

Barbagallo et al. [19] described a bio-inspired algorithm hinged on the scout-worker migration method where some of the scouts are professed to move from one physical node to another so that they can cooperatively find a suitable destination for the workers (i.e. the VMs) which are migrated. Their algorithm also focused on performance only.

In current years, conferring to the growth in the power consumption of datacenters of Cloud, the energy efficiency issue of has become one of the most crucial research problem. VT was tried by many software-oriented techniques have tried to minimize the consumed energy in Cloud environments as mentioned by R. Buyya et. al.[20].These efforts started in 2007 by Nathuji et. al. [21] where it was mentioned that the scintillating features of VT such as migration can be used to cause systems to be power-aware. Scheduling and resource allocation by means of VMs is one of the most well known ways to reduce the consumed energy of virtualized cloud datacenters.

In this context, Ajith Singh et. al.[22] proposed a nature-inspired honey bee algorithm for solving the dynamic VM placement problem for energy-aware Infrastructures as a Service (IaaS) cloud computing environments. They have tackled the problem of power efficient resource management in virtualized datacenters to maximize the Cloud provider's profit by minimizing both power consumption and service-level agreement (SLA) violation.

Bernardetta Addis et. al. [23] pointed out that there are five main problem areas which must be considered in system allocation policy design:1) application/VM placement, 2) admission control,3) capacity allocation,4) load balancing, and 5) energy consumption. They integrated all five problem areas within a unifying framework, providing very efficient and robust solutions at multiple time-scales. They have proposed resource allocation policies/algorithm for the management of multi-tier virtualized cloud systems with the aim at maximizing the profits associated with multiple-class of service level agreements(SLAs) and minimizing the infrastructure energy costs.

Finally, Mohsen Sharifi et. al. [24] considered energy efficiency along with performance. They showed that amaurotic consolidation of VMs does not minimize the power consumption of datacenters but it can also cause energy wastage. They then proposed a scheduling algorithm which was energy-aware using a group of objective functions in terms of consolidation fitness metric. Their proposed scheduling algorithm minimizes the overall power consumption of the data center by electing a set of VMs to a set of PMs in a very power efficient manner and the result of such assignment showed significant improvement in power savings by 24.9% and also performance degradation came down to 1.2% . Thus the practical result of the proposed scheduling algorithm was much better when compared to other scheduling algorithms.

4. PROPOSED MODEL WITH MAS FOR DYNAMIC CONSOLIDATION OF VMs

The proposed multi agent system consists of three different types of agents namely, Actor agent, Advisor agent and User agent. The work of all three agents is described as follows:

1. USER AGENT (UA): The users send requests to UA which either accepts or rejects those requests depending on the availability of resources required by the users. The updates about the resources are provided to the UA by the Advisor agent(AA) in regular interval of time. If AA send updates about the availability of resources required by the users, then UA will accept those requests otherwise it will reject.

2. ADVOSOR AGENT (AA): The AA will keep track of the actions of the Actor agent and also the resultant system changes and performances due to actions taken. The AA will keep records of the following data in the form of distributed data repository, which are explained as follows:

i) Action log data – that records all the actions which are taken by the Actor agents and the resultant system changes. It contains information about which algorithms were being used for 1) host overload detection, 2) VM selection, 3) VM placement.

ii) Distilled metric data – that collectively captures the current state of the entire infrastructure, the overall performance of the system such as the total energy consumed and number of SLA violations, number of free resources, number of fully occupied resources.

iii) Historical data – information about different algorithms and their performance in different load conditions. For example, under certain condition the algorithm number one gives better result than other VM placement algorithms if energy consumption is to be minimized and the algorithm number 2 gives better result if number of SLA violation has to be minimized. Thus, performance of different algorithms will already be stored and the respective algorithms can be used if similar situations arrive. It will have information about four different VM selection algorithms, namely MC (Max.Correlation), MMT (Min. Migration Time), MU (Min. Utilization), RS (Random Selection) , five host overload detection algorithms THR (Static Threshold), MAD(Median Absolute Deviation, IQR (Inter quartile Range), LR (Local Regression) and LRR(Robust Local Regression) and two VM placement algorithms, namely, Power Aware Best Fit Decreasing (PABFD)[16][17] and clustered based honey bee[18].

The AA will send suggestions to Actor agent to swap the current algorithms with the most suitable algorithms for a specific situation if the performance is degraded by certain system changes and also if the energy consumption is high. AA will use the historical data about the performance of different algorithms and in which situation they work better. Suppose honeybee algorithm works better than other VM placement algorithms when the aim is to reduce the overall energy consumption then AA will suggest the Actor agent to change the current algorithm with honeybee algorithm in order to achieve the goal of minimizing the energy consumption.

The AA will also have the current state of the entire infrastructure and will send information regarding the availability of resources to the User agent.

3. ACTOR AGENT (ACA): ACA will receive feedbacks about the performance of algorithms and orders to swap those algorithms (if the goal is not achieved) from the AA. ACA will receive the updates of the resource utilization from the local manager and will work accordingly with the aim of minimizing energy consumption and at the same time reduce the number of SLA violation. All the actions of ACA is monitored by AA and is changed as and when required after getting suggestions from AA.

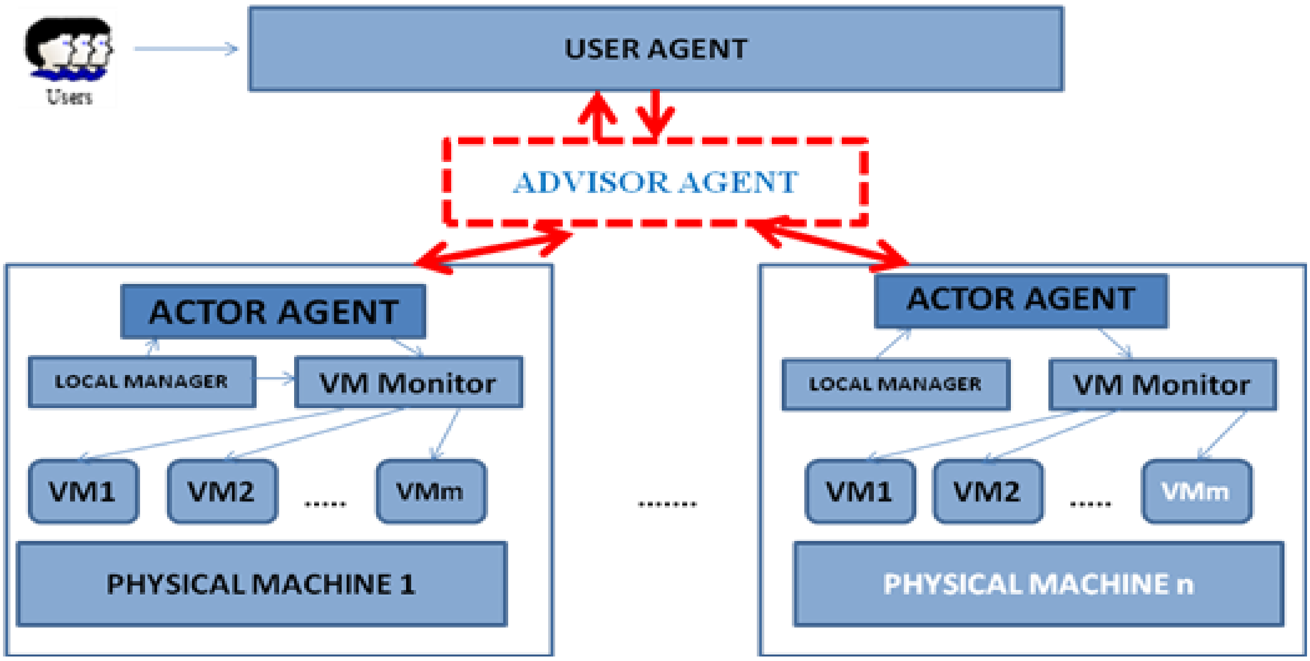

Fig 1: Proposed System Model with MAS

5. METRICS USED FOR MEASURING ENERGY CONSUMPTION AND SLA VIOLATION

The performance of the proposed work has been evaluated using existing metrics [25]. This algorithm is used to optimize two main parameters -energy consumption and SLA violation related to performance degradation. To portray the energy-performance tradeoff both the definition for energy consumption and performance degradation must be defined distinctly. In this study, the Energy Consumption (EC) by a server is defined as a linear function from CPU utilization, and performance is defined as a function that evaluates the SLA delivered to any VM deployed in an IaaS. The SLA violation is defined with the help of two metrics-SLA Violation Time per Active Physical machine (SLATAH) that rise with overload time period of the PM, and Performance Degradation due to Migrations (PDM) that rise due to live migration[1]. Hence, these metrics were defined with the assumption that the SLAs are delivered when 100% performance requested by any kind of applications inside a VM is provided at any time.

$$SLATAH = \frac{1}{N}\sum_{i=1}^{N}\frac{T_{si}}{T_{ai}} ....(1)$$

$$PDM = \frac{1}{M}\sum_{j=1}^{M}\frac{D_{dj}}{D_{rj}} ....(2)$$

Where N is the number of active physical machines; T_{si} is total time during which physical machine i has experience maximum CPU utilization ; T_{ai} is total time during which physical machine i being in the serving VMs; M is the number of VMs; D_{dj} is an estimation of the performance degradation of the VM j caused by migration (10% for this study); D_{rj} is total CPU capacity requested by VM j during its lifetime. A metric for describing SLA violation can be defined:

$$SLAV= SLATAH \times PDM....... (3)$$

Therefore, the best metric in cloud data center that can be able to describe energy performance tradeoff is the product of EC, SLATAH and PDM that the authors of [1] denoted it as Energy SLA Violation (ESV):

ESV = EC × SLAV------ (4)

In consideration of aforementioned formulation the SLA Time (SLAT) for each physical machine can be defined as follow:

$$SLAT_i = \frac{T_{si}}{T_{ai}} \ldots\ldots(5)$$

$$1 \le i \le N$$

5.1 Simulation tools

The simulation tools that are chosen are described in the following subsections. Two different tools are required to implement the proposed model: one for the MAS which is responsible for making the decisions and another one for performing the scheduling of VMs into physical machines. The tools are described as follow:

5.1.1 CloudSim Simulation Tool [26]

CloudSim is simulation software of cloud computing which is announced by Gridbus project in April 2009. CloudSim software framework consists of SimJava, GridSim, CloudSim, User Code. CloudSim is an extensible simulation toolkit that enables modeling and simulation of cloud computing systems and application provisioning environments.

The CloudSim3.0.3 [26] simulation tool is chosen to evaluate the proposed system. Proposed method for VM scheduling is implemented by using different existing algorithms from [16][27].

For VM placement, firstly a hierarchical clustering algorithm is used which makes cluster of resources based on CPU, memory, Bandwidth. Each cluster is considered as single resource and the main advantage of forming clusters is that the pursuit for available resources becomes easily accessible and helps in minimizing the response time while migration. After the clusters of resources are formed then VM placement is performed by using Honey Bee algorithm and evaluation of this clustered based honey bee algorithm [27] is done with the Planetlab Workload traces which is available with the CloudSim package. The simulation is executed for 24 hours with VM placement algorithm [27] along with the existing overload detection and VM selection algorithms proposed in [25][28]. The simulation process has been implemented with four different VM selections [25] -MC (Max. Correlation), MMT (Min. Migration Time), MU (Min. Utilization), RS (Random Selection) and five host overload detection algorithms[25] THR (Static Threshold), MAD(Median Absolute Deviation, IQR (Inter quartile Range), LR (Local Regression) and LRR(Robust Local Regression).

5.1.2 MAS- Agent Language [29] [30]

To implement the MASs a simulation experiment is performed in the CloudSim environment based on the some agent language. The consolidation algorithms [25][27][28] are implemented in CloudSim layer of the simulation platform, and MAS is implemented by the JACK agent simulation program in UserCode layer. However, the exploration process related to some more MASs platforms and software is undergoing. Some other tools have been studied which are the design and implementation tools, namely, Java Agent Development Framework (JADE). JADE platform is a well-known FIPA submissive platform for the development of Multi Agent Systems (The Foundation of Intelligent Physical Agents). JADE is software framework fully implemented in Java language. It simplifies the implementation of MAS. The only system requirement for JADE is Java Run time version 1.4 or higher. Other tool like JESS may also be helpful. Although JACK agent simulation program has been chosen, but it is believed that if the proposed system model is evaluated using other agent simulation tools than there may be variations in the output and this study is put as a future endeavor.

6. RESULT OF THE STUDY

The study is one of ongoing research. In each result only the statistical data and current trend of improvements are considered. The simulations of the Honey Bee Cluster [27] with respect to its impact generated by the proposed new concept of MAS are depicted in the following figure number 2, 3 and 4. The results of this study have been analyzed in three different scenarios, namely, energy consumption, Service Level Agreement (SLA) Violation and VM migration. It is observed that using MAS the overall system performance was optimized as compared to the conventional methods which are depicted as follows:

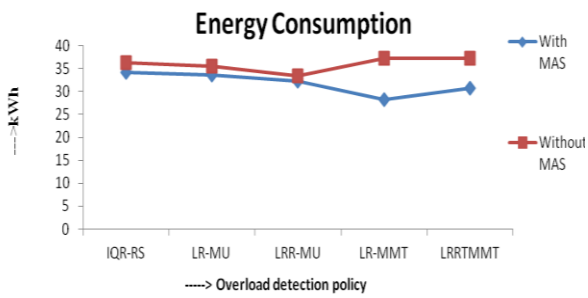

Fig 2: Energy Consumption Comparison

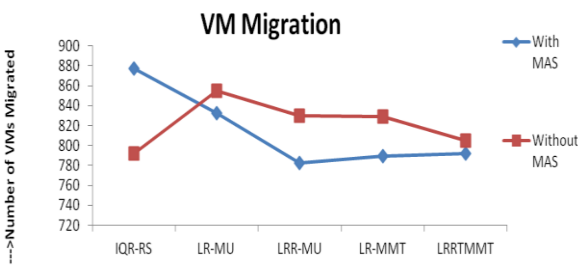

Fig 3: VM Migration Comparison

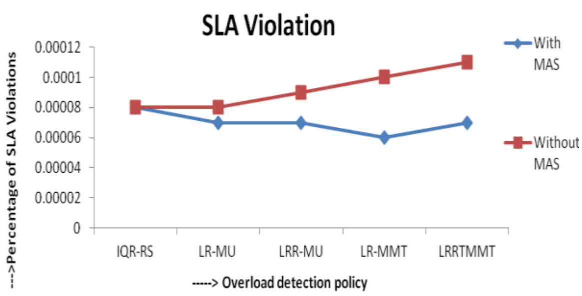

Fig 4: SLA Violation Comparison

By introducing MAS, the energy consumption is minimized as shown in figure 2 due the efficient resource usage and proper implementation of the consolidation. As depicted in figure3 it is seen that with the MAS the SLA violation was minimized as the performance degradation was significantly reduced due to the proactive manner in the processing. The rate of VM migration was also reduced as presented in figure 4, because the MAS using its percepts can make intelligent decisions while making the VM placements and as a result migrations can be avoided or lessen to a great extend. The graphs show the importance of using such MAS to optimize the system performance and minimize the usage of energy.

7. CONCLUSION

In this research paper MAS integrated with Cloud Computing assessment model for performing dynamic consolidation of VMs is proposed. Due to the introduction of MAS the energy consumption in cloud data center has been reduced to a great extend because of the intelligent decisions made by the agents in managing the resources. It is expected that the overall SLA violation and number of VM migrations will further be optimized. This work consists of the statistical data and improvements as this is one of the ongoing research works. As a future work, the implementation of the system will be performed in other MAS simulation environment like JADE, JESS etc. It is believed that the operating efficiency of the cloud computing system will be significantly improved by the introduction of MASs and further enhanced result is expected to be acquired in future.

8. REFERENCES

[1] Domenico Talia, "Clouds Meet Agents: Toward Intelligent Cloud Services", IEEE Internet Computing . Vol. 16, Issue 2, pp- 78-81. 0p , March 2012.

[2] K. P. Sycara, "Multiagent systems," AI Magazine, vol. 19, no. 2, pp.79-92, 1998.

[3] NIST Cloud Computing Program. www.nist.gov/ itl/cloud/. [Online]. Accessed on June 2013.

[4] M. Armbrust, et al., "A view of cloud computing," Communications of the ACM, vol. 53, no. 4, pp. 50-58, April 2010.

[5] Wuxue Jiang, Jing Zhang, Junhuai Li, Hui Hu, "A Resource Scheduling Strategy in Cloud Computing based on Multi-agent Genetic Algorithm", TELKOMNIKA, Vol.11, No.11, pp. 6563~6569, November 2013.

[6] Myougnjin Kim, Hanku Lee, Hyogun Yoon, Jee-In Kim and HyungSeok Kim, "IMAV: An Intelligent Multi-Agent Model Based on Cloud Computing for Resource Virtualization", In : 2011 International Conference on Information and Electronics Engineering, IPCSIT, vol.6,pp- 199-203, IACSIT Press, Singapore 2011.

[7] T.Han, K. M. Sim, "An Ontology-enhanced Cloud Service Discovery System", Lecture Notes in Engineering and Computer Science 2180(1) 2010: 644-649, 2010.

[8] Yue-Shan Chang, Tong-Ying Juan, Che-Hsiang Chang and Jing-Shyang Yen, "Integrating Intelligent Agent and Ontology for Services Discovery on Cloud Environment", IEEE International Conference on Systems, Man, and Cybernetics, pp: 3215 – 3220, 2012.

[9] Jaeyong Kang and Kwang Mong Sim, "Towards Agents and Ontology for Cloud Service Discovery", 2011 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, pp: 483 – 490, Beijing, 2011.

[10] Sheng-Yuan Yang and Dong-Liang Lee, "Developing a Cloud Intelligent and Energy-saving Information Interface Agent with Web Services", 2012 26th International Conference on Advanced Information Networking and Applications Workshops, pp: 1310 – 1315, Fukuoka , 2012.

[11] K. M. Sim, "Agent-based Cloud Computing", IEEE Transactions On Services Computing, Vol.5, No.4, pp.564,577, Fourth Quarter 2012.

[12] L. Tasquier, S. Venticinque, R. Aversa, and B. Di, "Agent Based Application Tools for Cloud Provisioning and Management",3rd International Conference on Cloud Computing (CLOUDCOMP 2012), Wien, Austria, September 24–26, 2012.

[13] Taha Chaabouni, Hamdi Kchaou and Maher Khemakhem, "Agent technology based resources management in Cloud Computing", 2013 World Congress on Computer and Information Technology (WCCIT), pp:1-3, Sousse, 2013.

[14] Qi Liu, Georgios K. Theodoropoulos, Dilma Da Silva and Elvis S. Liu, "Towards An Agent-Based Symbiotic Architecture For Autonomic Management Of Virtualized Data Centers", Proceedings of the 2012 Winter Simulation Conference, Article no. 147, 2012.

[15] Aravindh Ramaswamy, Aswath Balasubramanian, Palaniappan Vijaykumar and Varalakshmi P, "A Mobile Agent based Approach of ensuring Trustworthiness in the Cloud", IEEE-International Conference on Recent Trends in Information Technology, MIT, Anna University, Chennai. June 3-5, 2011.

[16] K. Mukherjee and G.Sahoo, "Green Cloud: An Algorithmic Approach", International Journal of Computer Applications (0975 – 8887), Volume 9– No.9, November 2010.

[17] Jinhua Hu, Jianhua Gu, Guofei Sun, Tianhai Zhao, "A Strategy on LoadBalancing of VirtualMachine Resources in Cloud Computing Environment", IEEE Third International Symposium on Parallel Architectures, Algorithms and Programming (PAAP), pp. 89 – 96, 2010.

[18] N Bobroff, A. Kochut, and K. Beaty, "Dynamic placement of virtual machines for managing SLA violations", In Proc. International Symposium on Integrated Network Management ',2007.

[19] D. Barbagallo, E. Di Nitto, D. J. Dubois, and R. Mirandola, "A bio-inspired algorithm for energy optimization in a self-organizing data center," In Proc. SOAR'09. Berlin, Heidelberg: Springer-Verlag, pp. 127–151, 2010.

[20] Anton Beloglazovy, Rajkumar Buyya,Young Choon Lee,Albert omaya, "A Taxonomy and Survey of Energy-Efficient Data Centers and Cloud Computing Systems", Technical Report, CLOUDS-TR-2010-3, Cloud Computing and Distributed Systems Laboratory, The University of Melbourne, 2010.

[21] Nathuji R, Schwan K, "Virtual power: coordinated power management in virtualized enterprise systems", In: ACM, pp 265–278. 2007.

[22] Ajith Singh. N and M. Hemalatha, "Virtual Machine Placement by Using Honey Bee ForagerAlgorithm in Cloud Computing", Karpagam Journal of Computer

Science, Vol7 Issue 4,Page No: 209-215, ISSN 0973-2926, May-June 2013.

[23] Bernardetta Addis, Danilo Ardagna, Barbara Panicucci, Mark Squillante, Li Zhang: "A Hierarchical Approach for the Resource Management of Very Large Cloud Platforms", IEEE Transactions On Dependable And Secure Computing, 2013.

[24] Mohsen Sharifi, Hadi Salimi, Mahsa Najafzadeh, "Power-efficient distributed scheduling of virtual machines using workload-aware consolidation techniques", The Journal of Supercomputing, Vol 61(1), pp. 46-66, 2011.

[25] Anton Beloglazovy, R. Buyya, "Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers". Concurr. Comput. Pract. Experience 24(13), 1–24 (2011)..

[26] R. Buyya, Rajiv Ranjan, Rodrigo N.Calheiros, Anton Beloglazov, C´esar A. F. De Rose, "CloudSim: a toolkit for modeling and simulation of cloudcomputing environments and evaluation of resource provisioning algorithms" , Advanced Information Networking and Applications (AINA), 2010 24th IEEE International Conference on, On page(s): 446 – 452, April 2010.

[27] Ajith Singh. N and M. Hemalatha, "Cluster Based Bee Algorithm for Virtual Machine Placement in Cloud Data Centre", Journal of Theoretical and Applied Information Technology, Vol. 57, No. 3, pp. 01-10, November 2013.

[28] Anton Beloglazovy, J. Abawajy, R. Buyya. "Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing". Future Generation Computer Systems 2011; doi:10.1016/j.future.2011.04.017

[29] JACK INTELLIGENT AGENT. http://en.wikipedia.org/wiki/JACK_Intelligent_Agents. [Online]2014.

[30] JADE. http://jade.tilab.com/. [Online]. Accessed on February 2014.