Asian Journal of Information Technology 15 (19): 3673-3686, 2016
ISSN: 1682-3915
© Medwell Journals, 2016

AIS-DAG: Artificial Immune System for Directed Acyclic Graphs Model Based Fair Resource Allocation for Heterogeneous Cloud Computing

[1]M. Kandan and [2]R. Manimegalai,
[1]Department of Information and Communication Engineering, Anna University, Chennai,
Tamil Nadu, India
[2]Department of Computer Science and Engineering, Park College of Engineering and Technology,
Coimbatore, Tamil Nadu, India

Abstract: This study focuses on the issues of resource allocation in Cloud computing. The user requirements, strategies, execution time and resource management for resource allocation is reviewed in different Cloud applications. The study also indicates the problems faced during resource allocation and how to improve the resource allocation efficiency in the Cloud. Load balancing guarantees the optimum use of available resources and thereby empowers reliability and performance of the overall system. In this study, an optimization technique Artificial Immune System (AIS) is utilized to provide a fast, accurate and nearest resource to the appropriate user request. The communication model used in this study follows the Directed Acyclic Graph (DAG) model which can improve the communication and make the cloud take the right decision for resource allocation. This study utilizes an artificial immune system based on time, cost, energy and the relevant resource allocation in a Cloud environment. The performance of the proposed resource allocation model, AIS-DAG is analyzed using NS3-GreenCloud. The simulation results show that the proposed AIS-DAG model has enormous potential as it offers major enhancements in the traits of response time, high potential for the improvement in energy efficiency of the data center and can effectively meet the service level agreement requested by the users.

Key words: Resource allocation, cloud computing, directed acyclic graph, artificial immune system

INTRODUCTION

The resource allocation process is a vital process in Cloud computing. With the advent of Cloud computing, the demand for Cloud resources has been increasing. The steep increase in demand for resources made the resource allocation a very essential part of a Cloud computing system. This necessitates the design and development of efficient resource allocation strategies in Cloud computing research. Efficient use of resources means that the resources are neither over-utilized nor under-utilized. The purpose of employing resource allocation is to maintain the usability of resource under those limits. The main constraints considered while developing a resource allocation strategy are the capacity of the server, available bandwidth and fault-domain definitions. Cloud computing is a novel standard for safe, reliable, highly secured computing environments for the users with assured Quality of Service (QoS). Parallel and distributed computing together with grid computing constitutes Cloud computing as shown in Fig. 1. The availability of 100 GE links (IEEE 802.3ba) reduces the number of the core switches, cabling and considerably increases the maximum size of the data center due to physical limitations. The general concept of Cloud computing is that the user's data are stored in the network data center rather than in a personal hard disk or any other storage device, so that the user can use the data from any system from anywhere in the world over a network connection. The data stored in the data centre are maintained by the service provider of the individual. The users access the stored data from anywhere in the world using Application Programming Interface (API) software. This software is also available as freeware and shareware in public networking sites and also in the service provider's site. Service providers offers three types of services such as Infrastructure, Software and Platform. These services provided in a Cloud network are named Infrastructure as a Service (IaaS), Software as a Service (SaaS) and Platform as a Service (PaaS). The main benefits of using Cloud computing is for remote access of resources, low cost and reliability.

The cloud computing process requires a simple infrastructure to access even huge data applications

Corresponding Author: M. Kandan, Department of Information and Communication Engineering, Anna University, Chennai, Tamil Nadu, India which in turn reduces the big investment in physical infrastructure. Also the services can be upgraded or expanded to a required limit quickly, since the Cloud computing network is more flexible and does not require any physical set up for expansion. For the Cloud computing system to function unceasingly without any disruptions in resources, the system needs a resource allocation strategy which allocates resources available to the applications in demand of resources. If the resources are not managed correctly, the services become famished of resources. The resource allocation strategy is vital in the case of stringent management of resources available. The resource allocation strategy collects the type and quantity of resources required and allocates resources dynamically within its limit of resources available. The strategy of allocation also needs information about time and order for efficient allocation of resource to the process.

The main objective of this study is to improve the efficiency of DAG model for resource allocation according to the communication strategy. And it has been optimized using AIS algorithm using an AIS based on cloning and selecting the best clone in a Cloud environment. AIS has a powerful global searching capacity in a given feasible solution value in a stipulated time interval. Therefore, the utilized AIS is well enhanced and balanced in exploration and exploitation in terms of resource allocation. Here, AIS-DAG shows its effectiveness to optimize resource allocation when compared with other existing resource allocation algorithms (other DAG models). AIS-DAG model is validated by conducting a performance evaluation study using the GreenCloud simulation tool. The contributions of this study include the following:

- The discussion and comparison about various DAG models for resource allocation and analyzing their advantages and disadvantages are presented
- Defining a new communication aware model based Cloud application and defining the properties of the Jobs
- An effective optimization model for resource allocation in Cloud environment is proposed
- An AIS algorithm for resource allocation in Cloud computing environments inspired by cloning and selecting the best clone is proposed
- Performance analysis and evaluation of the AIS-DAG model with respect to the other existing algorithms are presented

Literature review: This study discussed briefly about the various existing algorithms proposed and used for resource allocation which mainly consider the energy, time, cost and relevant resource allocation in a Cloud environment. Resource allocation is a process of allocating the available resources to the applications in need for the Cloud computing process. A Cloud computing system requires an efficient resource allocation to function without a hitch. The main purpose of a resource allocation strategy is to assign a resource to the process that is in demand of the resource. In this process the resource allocation strategy should maintain the usage of the resources between under-utilized and over-utilized limits. Not only the over usage of the resource, a disrupting agenda in the process of Cloud computing, the less utilization of available resource is also an unnecessary waste of resource which affects the efficiency of the Cloud computing system. The resource allocation strategy should maintain the usage of resources within the above mentioned limits to maintain the efficiency of the system.

Kandan and Manimegalai (2015a, b) presents a comprehensive survey of resource allocation strategies and discusses various optimal allocations schemes for mapping physical resources to the corresponding virtual resources. Kandan and Manimegalai, (2015a, b) A Multi-Agent based Dynamic Resource Allocation (MADRA) strategy is proposed to improve the Quality of Service (QoS) in terms of time and energy efficiency. Abirami and Ramanathan (2012) designed a new scheduling algorithm called as Linear Scheduling for Tasks and Resources (LSTR) for performing the Cloud tasks and scheduling the resources. To improve the efficiency the author integrated the Nimbus and Cumulus services together deployed in the second layer of the network architecture to obtain best scheduling and performing the tasks respectively. One of the main disadvantage of this LSTR is it should monitor the individual resources as it takes more time. Singh and Chana (2014) mainly focused on QoS in terms of energy efficiency and effective resource allocation in Cloud. The results obtained from the simulation is compared with the existing scheduling algorithm and proved that the QoS is better than the scheduling algorithm which functions in stages. Initially, the author utilized Green Service Allocator (GSA) as a part of the proposed approach where GSA communicates with the Cloud manager during the request entry. Next, Service Level Agreement (SLA) is called for service allocation. Finally the server takes care about the response generation. The approach proposed in Singh and Chana (2014) saves energy and there is no centralized control. Li *et al.* (2010) proposed a two stage contribution where in the first stage a scattered architecture is taken into account.

The resource management is handled by dividing the jobs and accomplishes the jobs by self-governing Node Agents (NA) in a cyclic manner. First placing the virtual machine, allocate the virtual machine into a relevant physical machine. Second, all the virtual machine and physical machines are monitored and if necessary virtual machines are migrated in case of no local accommodation. This method is difficult in large number of systems. Migrating virtual machines aid to load balancing which provides high provisioning and avoid hot-spots in data centers which reduce the energy consumption Li and Zheng (2014) and Chen *et al*. (2014) stated that by consolidating multiple utilized servers into a single server the energy consumption can be reduced. It helps to reduce the burden in load balancing too. Katyal and Mishra, 2014 presented a discussion about selective algorithms for resource allocation. The resource allocated in on-demand nature where min-min, min-max and heuristic algorithms are integrated in the selective algorithms. All the above three algorithms provides a minimized make span on allocation and it works well only for independent jobs. Shu *et al*. (2014) proposed an improved clonal selection algorithm for optimizing the time, cost and energy based resource allocation. Since more iteration to be done in Cloud simulation the computational complexity is more. Resource allocation is the key technology of Cloud computing which utilizes the computing resources in the network to facilitate the execution of complicated tasks that require large scale computation discussed by C-J Huang *et al*. (2013). Resource allocation needs to consider many factors, such as load balancing, make span and energy consumption. Selecting favourable resource nodes to execute a task in Cloud computing must be considered and they have to be properly selected according to the properties of the task as said by GS (Mukherjee and Sahoo, 2009). In particular, Cloud resources need to be allocated not only to satisfy Quality of Service (QoS) requirements specified by users via Service Level Agreements (SLAs) but also to reduce energy consumption stated by Rodero-Merino *et al*. (2010).

Numerous strategies were proposed earlier for efficient resource allocation in a Cloud computing system. Mu *et al*. (2010) proposed a strategy for resource allocation which utilizes the time taken for the execution of time along with the pre-emptable schedule. This method is tested in a heterogeneous VM and is specially designed for IaaS type of Clouds. The researchers from (Majumdar, 2011) states that, the technique used in (Mu *et al*., 2010) (utilizing the heterogeneous VM) for calculating execution time for a particular job is a critical process which also consumes more time. Based on the method proposed by Jiyani a similar strategy of resource allocation is proposed by Melendez and Majumdar (2010). They developed this method for distributed environment based Cloud systems. This method does not use any detailed knowledge about the process and is compatible for any type of scheduling. Chadwick *et al*. (2013) has proposed a new resource allocation strategy for mitigating the drawbacks of centralized resource management system. The centralized resource management system is poor in managing users and resources. To overcome this drawback a new layer named domain is added. Using this layer the resources are allocated based on RBAC (Role-Based Access Control) in this study. Resource fragmentation in multi cluster environment is a very challenging job in resource allocation.

From the above discussion it is clear and essential that allocating a relevant resource for the user request is much more important. Resource allocation also facilitates the other functions such as load balancing, energy consumption, time and cost reduction. In this study the entire topology used for simulation is depicted in Fig. 1. A centralized server, two servers of two networks or more number of servers of more number of networks are aggregated together to form the Cloud environment. The number of sub-networks aggregated with the server shows the scalability of the Cloud and there are no restrictions on the number of sub-networks. The access point of the networks helps to interconnect the user directly to the network.

MATERIALS AND METHODS

Proposed approach

Resource allocation models: One of the assumptions is made to generalize the AIS-DAG model is a set of jobs is divided into sub-jobs, where each job depends on the precedence of job. Each job is assigned to the available resources (Zhao *et al*., 2009) and each resource has its own assigned capacity like memory, network, storage and processing speed (Randles *et al*., 2010). One sub-job is processed at one machine at a time and makes the resource available continuously. The common methodology of resource allocation in a Cloud procedure is as follows:

Inputs and initializations: Let $R = \{R_1, R_2, R_3, \ldots., R_n\}$ is the set of n available resources which can process n independent jobs represented as $J = \{J_1, J_2, J_3, \ldots., J_n\}$, $I = 1, 2, \ldots., n$, $j = 1, 2, \ldots, n$. Here all R_i are parallel and every J_i is processed on any subset $R_i \in R$ is the available resources.

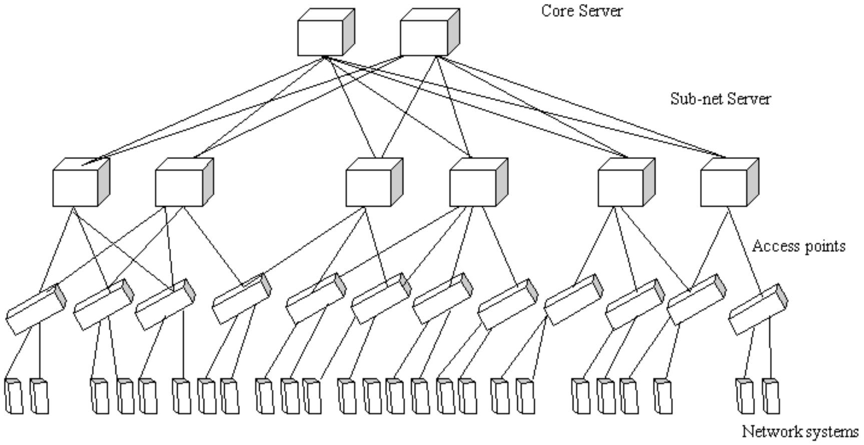


Fig. 1: Three-tier model in cloud environment

Outputs: The result is a best resource including a scheduled job having higher energy within a stipulated time interval.

Conditions: The time taken to execute each job should be less in terms of relevant resources. Each job should be completed without fail due to insufficient energy, long time process or any interruption. Also more than one job cannot be processed at a time.

Objectives: The main motto of these models is to improve the efficiency of energy in the data center by allocating appropriate resources within a stipulated time interval.

Since various optimization algorithms are involve simultaneously on multiple objectives (Araujo *et al*., 2011) and increase the computational complexity, the proposed AIS-DAG model is designed for resource allocation optimization which fully integrate three factors of energy, time, relevancy, availability and cost optimization.

Energy optimization: In this section, the energy optimization model is proposed based on the DVFS (Dynamic Voltage and Frequency Scaling) (Rizvandi *et al*., 2010) method. The energy of the resource depends on the supply-voltage and frequency of the resource. The dynamic energy consumption is achieved by improving the node capacitance by dynamic charging and it is represented as:

$$P = \gamma \times v^2 \times f \qquad (1)$$

In Eq. 1, $\gamma = A \times C$

Where:

A = The frequency of the flip representing the number of switches per clock cycle
C = The capacitance
v = The supply voltage
f = The frequency of the resource node

Definition 1: Let assume S^i represents the voltage supply of the resource R^i and S^i has K level of DVS (Dynamic Voltage Scaling), then the supply voltage and frequency relationship matrix of S^i can be written as follows:

$$V_i = \begin{bmatrix} \left(\left(V_1(i)\right), f(i)\right); V_2(i), f_2(i); \\ \ldots\left(V_k(i), f_k(i)\right) \end{bmatrix}^T \qquad (2)$$

Where:

$V_k(i)$ = is the supply voltage for the resource
R_i = At the level k,
k = Is the number of levels in S^i and
$f_k(i)$ = Is the frequency at k level where $0 \le f_k(i) \le 1$.

Definition 2: Similar to the definition-1, we can define that the job J_i requires CT(i, j) time to complete at the resource r_i. Then the completion time for the job J_i on the resource r_i can be represented mathematically in Eq. 3

$$CT[i, j] = \begin{bmatrix} \frac{1}{f_1(i)} \times CT(i, j), \frac{1}{f_2(i)} \times CT(i, j), \\ \ldots \frac{1}{f_k(i)} \times CT(i, j) \end{bmatrix} \qquad (3)$$

Definition 3: Similar to the above definitions 1 and 2, let assume $V_k(i)j$ is the voltage and $f_k(i)j$ is the frequency and CT(i, j) is the required completion time of the jobJ_i on the resource r_i. Then, the energy consumed for completing the job J_i on r_i at k DVFS level during the supply strategy S^i is defined as follows:

$$E_{ijl} = \gamma \times f \times \left[v_k(i)_j \right]^2 \times CT(i,j) \tag{4}$$

In Eq. 4, $\gamma = A \times C$ is the fundamental property of the resource.

Definition 4: In case of $Idle_i$ time of r_i, L(j) says a set of DVFS level utilized for all the jobs assigned to the resource r_i; then the energy utilized by the r_i for completing the jobs can be defined as follows:

$$E_i = \gamma \times f \times \left\{ \sum_{j \in J(i), k \in L(j)} \begin{pmatrix} \left[V_k(i)_j \right]^2 \times CT(i,j) \end{pmatrix} + \atop V_{min}(i) \times f_{min}(i) \times Idle_i + \lambda \right\} \tag{5}$$

In Eq. 5, V_{min} (i) and f_{min} (i) denotes the minimum voltage and frequency value of the resource r_i transition to sleep mode during the Idle time. Also λ is the load factor of the resource r_i.

Time optimization: The entire work completion time is the time required between the starting and ending of a sequence of jobs on a resource (Zhu *et al*., 2011). Cloud computing deals with assigning jobs dynamically to the resources according to the need requested by different users. Completion time CT(i,j) includes request time, response time, waiting time and receiving time. The completion time CT for resource r_i should be reduced and is denoted as Min-CT and it can be defined in Eq. 6:

$$\min - CT = \max \left\{ \begin{array}{l} \dfrac{CT_{ij}}{CT_i} \in CT, i = 1,2,...,n \text{ and} \\ R_i \in R, j = 1,2,...,m \end{array} \right\} \tag{6}$$

AIS-DAG selects the resources according to the minimum CT.

Multi-objective optimization: In the multi objective, the energy, time and resource allocation are together optimized for resource allocation in GreenCloud computing is represented in Eq. 7-8.

$$E_i = \gamma \times f \times \left\{ \sum_{j \in J(i), k \in L(j)} \begin{pmatrix} \left[V_k(i)_j \right]^2 \times CT(i,j) \end{pmatrix} + \atop V_{min}(i) \times f_{min}(i) \times Idle_i + \lambda \right\} \tag{7}$$

$$\text{Min} - CT = \max \left\{ \begin{array}{l} \dfrac{CT_{ij}}{CT_i} \in CT, i = 1,2,...,n \text{ and} \\ R_i \in R, j = 1,2,...,m \end{array} \right\} \tag{8}$$

DAG model: The intensive Cloud application for Cloud communication is defined by AIS-DAG model. It is also compared with the existing DAG models such as CU-DAG (Communication Unaware-DAG) (Srikanth *et al*., 2012; Thulasiraman) and EB-DAG (Edges Based-DAG) models (Choudhury *et al*., 2012; Chen *et al*., 2014). In general, the Cloud computing applications requires the available resource communication for their new operations. The previously proposed models which rely on the HPC (High Performance Computing) concepts were based on DAGs that are formed by vertices collection. It may represent a computing job and the direct edges. But they lack in communication requirements. Though this was fulfilled by the models Communication-unaware and Edges based DAG, it has some frailty. In the former model has a single vertex making it difficult to schedule them properly. Whereas EB-DAG model has the drawback of preventing the different computing jobs from the same data in receiving the input and keeping the accuracy of the network inaccessible. To overcome this, a model named CA-DAG (Communication Aware DAG) is defined as below. The scheduling in communication aware distributed acyclic graph model is known to be NP-complete even for DAG without vertices or edges (Ullman, 1975). Scheduling is most important in Cloud application while resource allocation. Most of the scheduling strategies consider single objective criteria. But AIS-DAG considers and provides best solution on multiple objectives. And we also take the CA-DAG model as our DAG model for optimization.

It is represented by the directed acyclic graph G = (V, E, ω, φ). The set of two vertices V = {Jc,Vcomm }. Where, Jc and Vcomm are the two non-overlapping subsets. The set ⊆ JcV , represents computing jobs. The set ⊆VcommV, represents communication jobs of the program. A computing job Jc is described by the pair (I, Dc) with instructions I which is the amount of work that has to be executed with a deadline Dc. The communication job Vcomm is described by the pair (S, Dcomm) with information S in bits within a deadline Dcomm. ω(Vic) is the positive weight represents the computing cost vc and φ(Vicomm) is the positive weight represents communication cost Vcomm at the nodes Vc and Vcomm, respectively.

E, the set of edges consists of directed edges eij which represents dependence between the nodes vi∈V and vj∈ V. This means that a job vj relies on the input and cannot be started until the input is received from the job

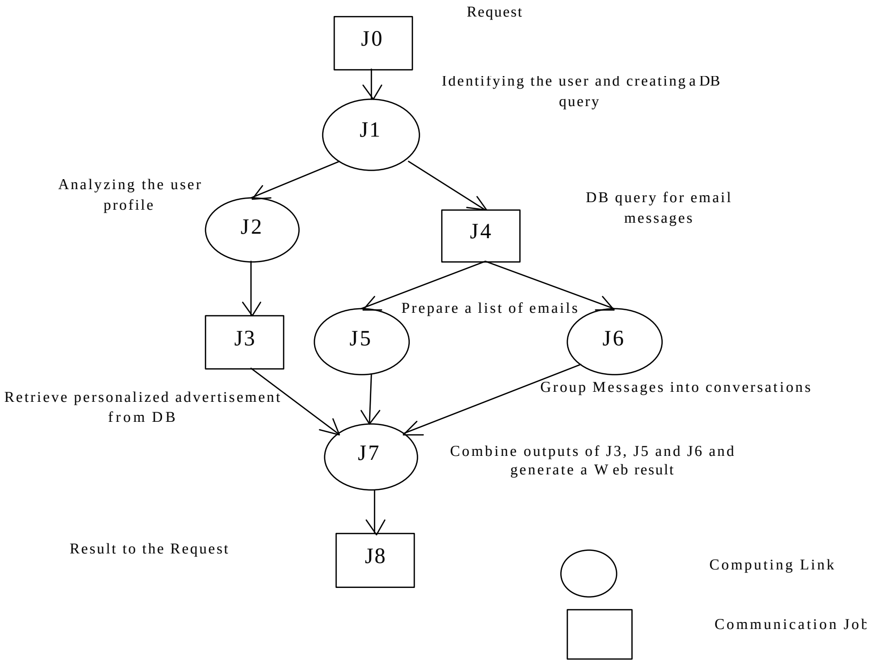

Request
J0
Identifying the user and creating a DB query
J1
Analyzing the user profile
J2
J4
DB query for email messages
J3
J5
Prepare a list of emails
J6
Retrieve personalized advertisement from DB
Group Messages into conversations
J7
Combine outputs of J3, J5 and J6 and generate a W eb result
Result to the Request
J8
Computing Link
Communication Job

Fig. 2: DAG model of cloud mail application

vi. When the input size is zero, it helps in defining the execution order of the jobs which exchanges null data.

Vcomm represents the communication job that occurs in the network subjects to communication content, link errors and significant delay. Edges E represent the results of exchange in the jobs which is executed on the same physical server. It is fast and the associated delay is neglected. In an illustrative way, a typical Cloud computing application was considered. Its level of operation was performed in four steps as follows:

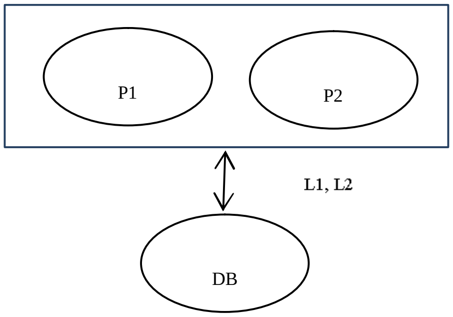
- User request was received and processed
- Generating personalized advertisement
- The email messages are requested from the database
- HTML page is generated and was sent to the user

All these steps represents communication aware DAG with the help of AIS. In Fig. 2, the DAG vertices with respect to the computing jobs are represented by the circles whereas the communicating jobs Vcomm are represented by squares. It is associated with 7 jobs.

- Job 0 is associated with user request arrival and its delivery to the computing resources in the data center network
- Job 1 Involves with request processing, user identification, database query preparation
- Job 2 involves user profile analysis which determines the targeted advertisement traits
- In job 3, the requested personalized advertisement was obtained from the database
- In job 4, for the list of user email messages, database was queried. When the reply is arrived, it is fed to job 5 and parallel running job 6. Where the job 5 prepares email messages list, job 6 chooses the messages to be grouped into conversations
- In the final process, the outputs of job 3, 5, 6 are combined by job 7 and HTML page was generated and sent to job 8

When the optimization of total execution time was considered as in Fig. 3, for a set of identical computers to schedule job with communication, DAG computes the resources with two processors of a data center P1 and P2. L1 and L2 are the network links connecting the computing resources and database DB. There are two processes P1 and P2 are in queue to obtain a relevant resource in the Cloud. They are connected to the DB for resources via the links L1 and L2 respectively.

In Fig. 4 as per the CA-DAG model, computing jobs 1, 2, 5, 6 and 7 are scheduled on p1 processor while the jobs 0, 3, 4 and 8 are scheduled at the link L1. The methods of representing the communication jobs with own distinct vertices help to control an allocation and execution time. By this the processor time is not wasted in the waiting



P1, P2: Processors of computing work

DB: Database

L1, L2 Communication link

Fig. 3: Infrastructure of scheduling illustration

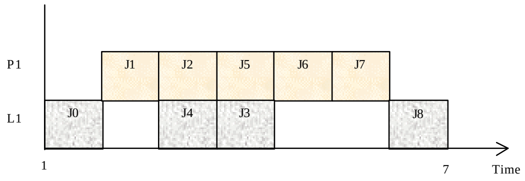


Fig. 4: Communication aware DAG model

time in order to complete the communication time. Additionally the scheduling of flexibility is not available too. In a single vertices representation job parallelization is not possible. Opposite to the computational jobs, communication jobs are parallel executed. Each job $V_i^{comm} \in V_{comm}$ is divided into n different independent communication jobs V_j^{comm}, j=1, 2, ..., n has the size of the job is in bits equal to $\varphi(V_j^{comm})/n$. All the bits are transmitted independently in different network path and they will be re-assembled in the form of original sequence at the receiver side. Every V_j^{comm} is split into a number of data and scheduled parallel or sequentially. The sender and the receiver are treated as common nodes and the communication depends on the network path, size of the data, bandwidth and the protocol deployed in the network.

In this study the AIS-DAG model is considered, to represent the various communication resources of different types used in real time systems. It concentrates on decision making to allocate the resource efficiently, scheduling the request-response maintenance and handling the jobs which are applied based on the DAG model.

Artificial immune system: One of the optimization techniques is Artificial Immune System. It is used to find the best value for large problems in various domains. Artificial Immune System is used to optimize large size problems which can emulate processes by utilizing the mechanisms of the biological immune system. AIS have local and global searching methodologies by creating affinity based mutation and cloning. AIS compare the results of the small problems with the results of the large problems. Through which it can able to obtain a best solution for large problems. Since, in this study AIS is utilized to fetch the optimum value of DAG model. When a request comes for a resource, the system calls the AIS to choose the best resource from a DAG model. Before obtaining an optimum solution with AIS, first get the resource according to the communication strategy. Once a request comes from Cloud user, the AIS-DAG will run to optimize the overall resource allocation by verifying the objective constraints. Initially the jobs and the resources are mapped using a bi-linear mapping method using a binary code which is called as initial set of population P(0). An individual P_i^G is represented $P_i^G = \left(P_{i1}^G, P_{i2}^G, \dots P_{ip}^G\right)$ as where G says the present population generation, i =1, 2, …, s and s says the size of the population.

Every antibody (individual) is treated as a candidate solution which can be represented by a binary string of bits. The user can set the string length to obtain the best solution for the problem. 0 or 1 represents the gene in the chromosomes. After generating the populations successfully, the affinity value of each individual is investigated and stored for future operations. The AIS-DAG model is applied for resource allocation to contract with the optimization problem where the affinity function is contract with energy efficiency, time and cost. The affinity function in AIS algorithm is defined in Eq. 9.

$$\text{aff}\left(x\right) = e^{\min E_i + \min M_s} \tag{9}$$

AIS-DAG based resource allocation is summarized as follows: Cloning operation combines identical cells in the human body which is generated from the same ancestor and antibodies with high affinity will be cloned to attack pathogens. Also cloning is an antibody random map persuaded by affinity. According to the affinity function all the antibodies are evaluated and stored in decreasing order. After arranging in decreasing order the higher affinity values are selected for next generation. Then the selected antibodies proliferate into certain copies and the copied original ones are replicated in the current population. Afterwards, the antibodies in the population will implement mutation operation. Then the antibodies in the population are mutated. AIS have two different mutation operations as pair wise and inverse mutation which can generate a new mutated individual P_i^G. Figure 5 and 6 shows the pair wise and inverse mutation operation respectively.

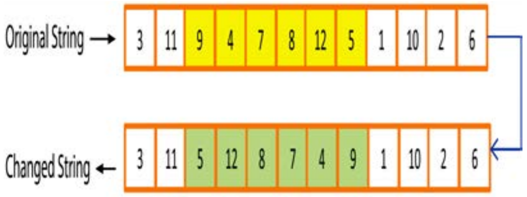


Fig.5: Inverse mutation

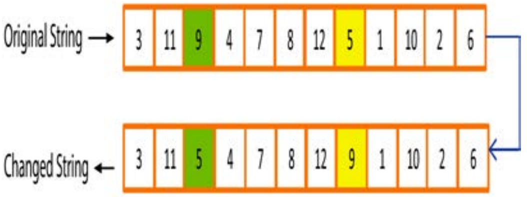


Fig. 6: Pairwise mutation

Algorithm

Artificial_immune_system

```
{
•   Generate the random population antibodies $P(0)$, total size of P is S
•   Calculate the affinity of antibody $P(k)$
•   Selects the R% of higher antibodies $P_l(k)$
•   Clone the individuals $P_l(k)$ and get $P_C(k)$
•   Apply mutation on the population $P_l(k)$ to form the new mutated
    population $P'(k)$
•   Compare the affinity of $P'(k)$ with $P_l(k)$ replace the larger affinity with
    the lesser affinity
•   Select the population $P'(k)$ and obtain the new population generation
    as $P(k+1) = P_C(k) \cup P'(k)$, $k = k+1$
•   Repeat the above steps 2-4 until $K>X_G$ reached [ stopping criteria] or
    the reached Pm
}
```

Where:

S = Population size
P = Populations
X_G = Number of generation
Pm = Mutation probability
K = Iteration operation
R = Replacing factor

AIS-Numerical illustration: The entire functionality of the AIS is described in terms of optimizing the resource allocation. The objective function value is obtained using certain number of parameter assignment as P, K and R. In this study; P = 100, K = 500 and R = 20%. These values may be changed to get best solution.

Initialization: It is clear that the population-P, Iteration-K and replacing factor-R are initialized. The population is generated randomly to create feasible combinations through which each combination represented by one string called as clones and the number of clones is P. The entire set of population is taken into account of the initial population. Example, the string is:

$$S = \{J, J_T, J_s, J_L, J_E, R\}$$

Each resource has more values in S and it can be mutated with different value for obtaining optimum value as OFV (Objective Function Value).

Objective function: The main objective function of AIS-DAG model is to allocate the best resource according to the optimum values such as less energy, less time, available resource and taking less cost for processing the job in the resource. The objective function value is denoted as OFV calculated for all the solutions P and affinity value is computed using the following Eq. 10.

$$\text{Affinity value} = \frac{1}{\text{OFV}} \tag{10}$$

It is denoted that the affinity value is inversely proportional to the OFV value.

Clonal selection: In Eq. 11, the String S is selected and cloned. This number of cloning is generated and applied that are directly proportional to the affinity value with a Rate Of Cloning (ROC) and it can be calculated by:

$$\text{ROC} = \frac{\text{affinity value of P}}{\text{total affinity value in the solution}} \tag{11}$$

Clonal expansion: Original copy of one string is called as a clone. According the ROC value, new clones can be generated. Example if ROC value is 1.4, then the number of new clones is 2. It increases the population for obtaining best value as optimum.

Mutation: Mutation is a process can make new clones of the population. Mutation can change the clone using reverse mutation and pairwise mutation. The following Fig. 5 illustrates the inverse mutation of the string S, where the numbers of resources are 12 and the inverse mutation starts at location 3-8.

In Fig. 5, after the inverse mutation, the OFV is calculated for the mutated string and the mutated OFV which is lesser than the clone OFV is checked, the original string is replaced by the mutated string, or else, the original string is retained. The following Fig. 6 illustrates the pairwise mutation of the string S, where the numbers of resources are 12 and the pairwise mutation starts at location 3 and 8.

After the pairwise mutation, the OFV is calculated for the mutated string and check the mutated OFV is lesser than the clone OFV, then the original string is replaced by the mutated string, else, the original string is retained. After Mutation, the number of strings in the improved string is higher than the population size. To maintain the initial population size, arrange the improved strings in ascending order and remove the strings having highest OFV.

Receptive editing process: R = 20% where 20% of the highest OFV based clones are replaced by newly generated population and repeat the same process described above to obtain the best OFV. Repeat the above steps until the number of iterations mentioned.

The entire AIS process is applied for DAG model where the request parameters, user information and the resource behaviour such as speed, time taken to process, cost to process all optimization and the best resource is chosen and assigned to the appropriate user request. The AIS selects the resource with less cost, less time, less energy consumption for processing the job requested from the user. Hence AIS-DAG model provides communication aware, best resource allocation strategy for Cloud computing.

RESULTS AND DISCUSSION

In this study the efficiency of the proposed approach is investigated by simulating its procedure in Green Cloud (GC) simulation tool. GC is an online tool used for packet level simulation. The GC tool is used to develop new solutions in resource allocation monitoring, scheduling the work as well as optimizing the communication in terms of network infrastructure. GC is integrated with Network simulator which can be used for analyzing the functionality of the network. GC is a well-known tool used to do simulation for Cloud computing. It offers simulation of current Cloud environment based resource allocation. GC is based on ns-2 simulation platform. It is able to make Cloud elements such as data centers, switches, servers and network links. GC supports three-tier architecture based application simulations. GC provides a simulation environment helps to do simulation for homogeneous as well as heterogeneous networks.

Experimental results: This section provides the results and the performance evaluation on the results which confirms the advantages of the AIS-DAG model for resource allocation in Cloud environment. AIS-DAG model is compared with the CA-DAG (Kliazovich *et al*., 2016) which is already compared with CU-DAG and EB-DAG given in the earlier research works. The number of resources, with more number of requests is modelled in DAG and according to the time, cost, energy and available resources optimized using AIS algorithm. The number of nodes N and the number of communications C is changeable to fetch more graphical representation based results. The simulation is repeated with various numbers of nodes as 20, 30, 40 and 50. DAG has two kind of communication as frequent and occasional and it is denoted by the terms frequency and the values assigned as 0.5 for occasional and 1 for frequent communication.

Figure 7 and 8 show the obtained result of the approximation value for CU-DAG (Srikanth *et al*., 2012; Thulasiraman), EB-DAG (Choudhury *et al*., 2012), CA-DAG (Kliazovich *et al*., 2016) and AIS-DAG with the occasional and frequent communications respectively. AIS-DAG model proves efficient as shown in both figures. The CCR (Communication Computational Ratio) value is close to 0.1 for all the DAG models. Since CCR value is very small, the communication is also less. Among all the DAG models AIS-DAG model provides more communication than the other DAG models. By increasing the CCR values like 1 and 2 the benefits of the AIS-DAG increased in terms of amount of data transmissions. The approximation value defines the relevance of the resource allocation where the relevancy is only a approximate value not exact value. In Fig. 7, various resource allocation approaches were verified for same CCR value as 0.1. The relevant resource allocation for the assumed CCR values in AIS-DAG model gave more relevant resources than the expectation from other DAG models. The entire efficiency of all the approaches compared in this study has been given in different colours. Each approach obtains three different results for three different CCR values. In all the approach AIS-DAG obtains the better performance than the other DAG models as proved in this researc.

Figure 9 and 10 illustrates the efficiency of the scheduling method utilized by various Fig. 9 and 10 illustrates the efficiency of the scheduling method utilized by various communication models. It is very clear that the results have been obtained with approximate values. Similarly, the scheduling applied for communication under AIS-DAG is superior than the other DAG models for CCR value ranging from 0.5-3.5. Figure 10 the emphasizes the importance of communication.

From Fig. 7-10, it is evident that AIS-DAG significantly improves the approximation value and scheduling efficiency. More communication happens with less CCR. AIS search the best fitness value and provide efficient communication among the users and the Cloud.

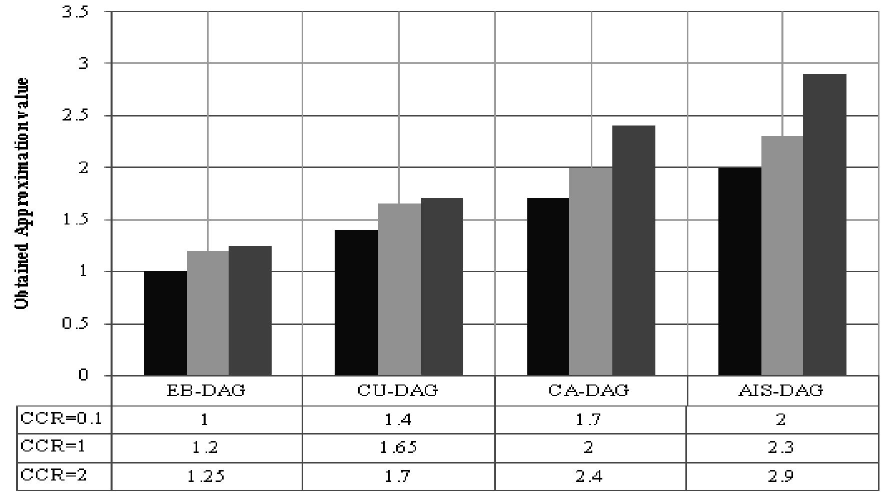


Fig. 7: Approximation factor for DAGs with occasional communications

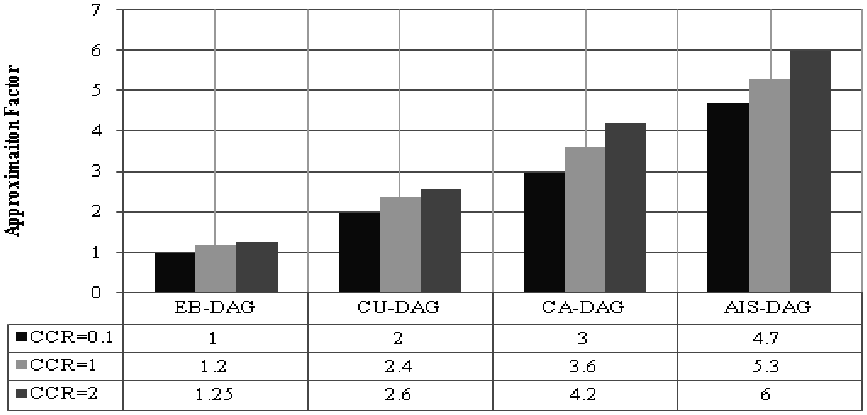


Fig. 8: Approximation factor for DAGs with frequent communications

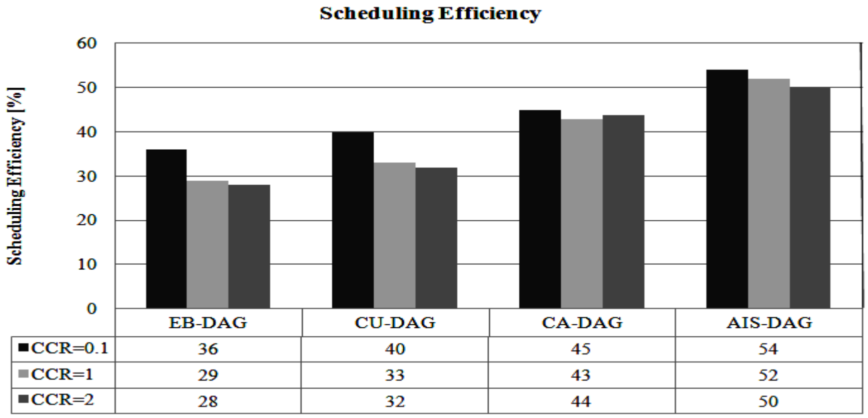


Fig. 9: Schedule efficiency for DAGs with occasional communications

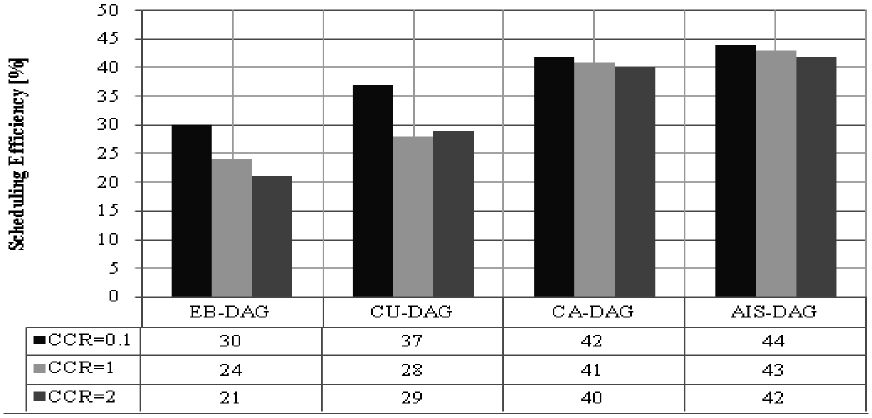


Fig. 10: Schedule efficiency for dags with frequent communications

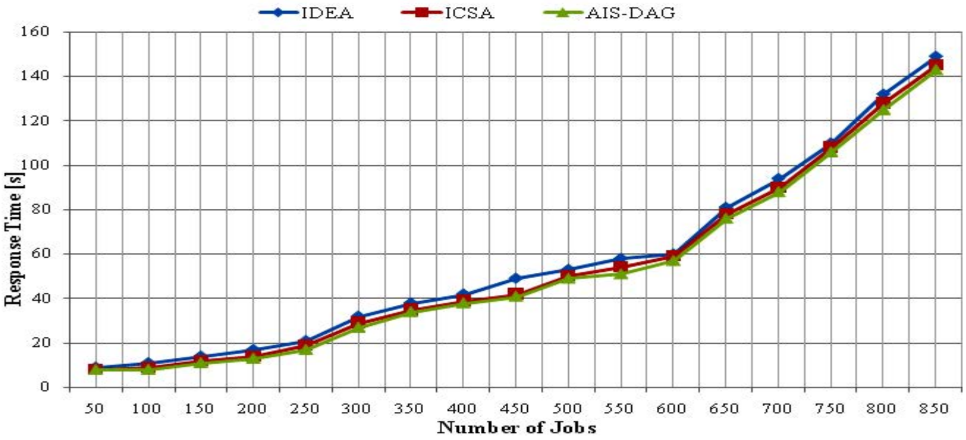


Fig. 11: Comparison of the response time of the three resource allocation algorithms: for different tasks

In AIS-DAG model, the communication among the users and the resource allocation is serialized one.

The complete experiment is handled in two parts. In part-1 there are three scenes set for the simulation based experiment. First the response time obtained by three optimization algorithms [IDEA (Tsai *et al*., 2013), ICSA (Shu *et al*., 2014) and AIS-DAG] are compared and shown in Fig. 11. The number of jobs and the response time is represented in X-axis and in Y-axis respectively. In part-2 the Job completion time required by all the optimization algorithms are compared and shown in Fig. 12. The Y-axis represents the time and the X-axis represents the number of jobs. One of the performances obtained from optimization methodologies is response time. Response time depends on the number of requests. Optimization process optimizes the time according to the number of requests. The resource allocation process allocates a resource that has less response time, better accuracy (relevancy) and nearer to the server.

Comparing with the other existing resource allocation methods AIS-DAG model allocates the relevant resources in least time as shown in Fig. 12. We thus prove that AIS-DAG is best in term of response time.

Since AIS-DAG works under optimization methodology, we compare AIS-DAG model with the other optimization based resource allocation models. In terms of relevancy, Jobs incoming vs. response time and jobs incoming vs. job compilation time, this method gives the best result. According to the response time and job compilation time AIS-DAG model is more efficient than the other DAG models and it is clearly depicted in Fig. 11 and 12. In this study, the energy consumption is calculated. The amount of energy need to do a process depends on the number of process, size and the type of the resource. From the simulation it clear that AIS-DAG model takes only less amount of energy than the other DAG models. The energy consumed by all the models is shown in Fig. 13.

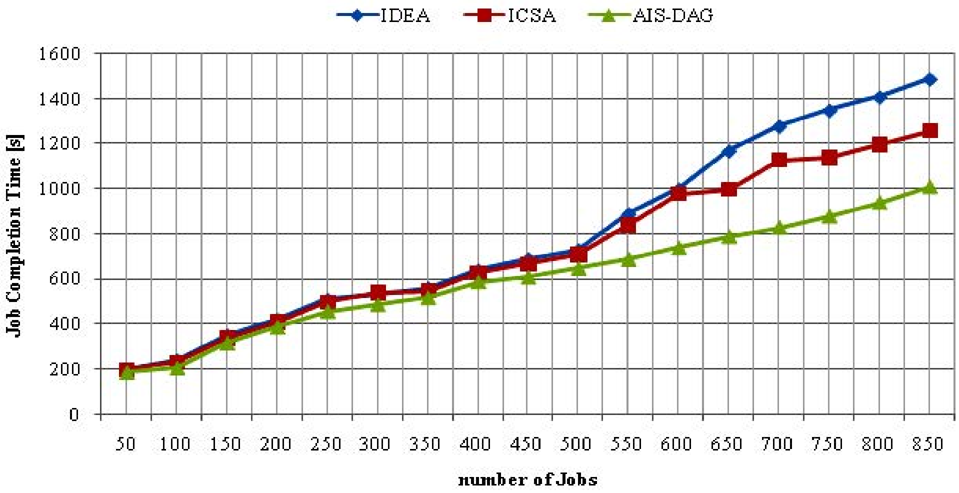


Fig. 12: Comparison of the compilation time of the three resource allocation algorithms for different tasks

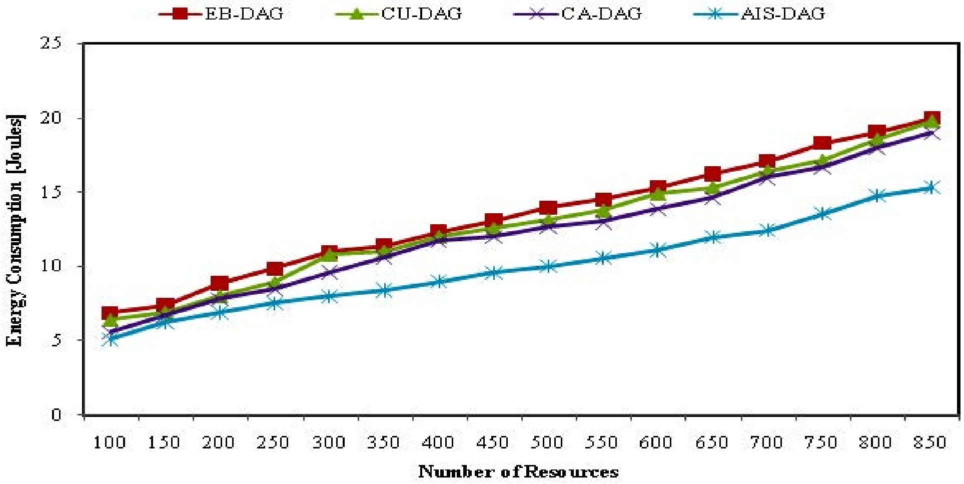


Fig. 13: Number of resources vs. Energy consumption

CONCLUSION

Cloud computing is a group of virtual computers treated as resources. GreenCloud computing is a new trend in recent research. Allocating a resource within a stipulated time interval with minimum energy consumption is a great challenge in Cloud computing. In this study one of the best communication models is proposed for providing energy efficient, fast response time and low cost resources to the requested users in the Cloud. It is done by AIS-DAG model which consumes less power and process the resource allocation in less time with good response. The experimental results shows that AIS-DAG model has immense potential as it offers significant improvement in reducing the execution time which demonstrates high potential in improving energy efficiency of the data centres in the Cloud. Also it can effectively meet the service level agreement requested by the users. In future, it can be improved by considering the operators and computation complexity to make further works more practical in Cloud computing.

REFERENCES

Abirami, S.P. and S. Ramanathan, 2012. Linear scheduling strategy for resource allocation in cloud environment. Int. J. Cloud Comput.: Services Archit., 2: 9-17.

Araujo, D.R., F.C.J. Bastos, E.A. Barboza, D.A. Chaves and F.J.F. Martins, 2011. A performance comparison of multi-objective optimization evolutionary algorithms for all-optical networks design. Proceedings of the 2011 IEEE Symposium on Computational Intelligence in Multicriteria Decision-Making (MDCM), April 11-15, 2011, IEEE, Paris, France, ISBN: 978-1-61284-068-0, pp: 89-96.

Chadwick, D.W., M. Casenove and K. Siu, 2013. My private cloud-granting federated access to cloud resources. J. Cloud Comp. Adv. Syst. Appl., 2: 1-16.

Chen, F., J. Grundy, J.G. Schneider, Y. Yang and Q. He, 2014. Automated analysis of performance and energy consumption for cloud applications. Proceedings of the 5th ACM/SPEC International Conference on Performance Engineering, March 22-26, 2014, ACM, Dublin, Ireland, ISBN: 978-1-4503-2733-6, pp: 39-50.

Choudhury, P., P.P. Chakrabarti and R. Kumar, 2012. Online scheduling of dynamic task graphs with communication and contention for multiprocessors. IEEE. Trans. Parall. Distrib. Syst., 23: 126-133.

Huang, C.J., C.T. Guan, H.M. Chen, Y.W. Wang and S.C. Chang *et al*., 2013. An adaptive resource management scheme in cloud computing. Eng. Appl. Artif. Intell., 26: 382-389.

Kandan, M. and D.R. Manimegalai, 2015a. Strategies for resource allocation in cloud computing: A review. Int. J. Appl. Eng. Res., 10: 10: 1-10.

Kandan, M. and R. Manimegalai, 2015b. Multi agent based dynamic resource allocation in cloud environment for improving quality of service. Aust. J. Basic Appl. Sci., 9: 340-347.

Katyal, M. and A. Mishra, 2014. Application of selective algorithm for effective resource provisioning in cloud computing environment. Int. J. Cloud Comp. Serv. Archit., 4: 1-10.

Kliazovich, D., J.E. Pecero, A. Tchernykh, P. Bouvry and S.U. Khan *et al*., 2016. CA-DAG: Modeling communication-aware applications for scheduling in cloud computing. J. Grid Comp., 14: 23-39.

Li, J., M. Qiu, J.W. Niu, Y. Chen and Z. Ming, 2010. Adaptive resource allocation for preemptable jobs in cloud systems. Proceedings of the 2010 10th International Conference on Intelligent Systems Design and Applications (ISDA), November 29-December 1, 2010, Cairo, Egypt, pp: 31-36.

Li, X. and M. Zheng, 2014. An Energy-Saving Load Balancing Method in Cloud Data Centers. In: Frontier and Future Development of Information Technology in Medicine and Education. Shaozi, Li., Q. Jin, X. Jiang and P.J. James (Eds.). Springer Netherlands, Netherlands, Europe, ISBN: 978-94-007-7617-3, pp: 365-373.

Majumdar, S., 2011. Resource management on cloud: Handling uncertainties in parameters and policies. CSI. Commun., 35: 16-19.

Melendez, J.O. and S. Majumdar, 2010. Matchmaking with limited knowledge of resources on clouds and grids. Proceedings of the 2010 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS), July 11-14, 2010, IEEE, Ottawa, Canada, ISBN: 978-1-56555-340-8, pp: 102-110.

Mu, P., J.F. Nezan, M. Raulet and J.G. Cousin, 2010. Advanced list scheduling heuristic for task scheduling with communication contention for parallel embedded systems. Sci. China Inf. Sci., 53: 2272-2286.

Mukherjee, K. and G. Sahoo, 2009. Mathematical model of cloud computing framework using fuzzy bee colony optimization technique. Proceedings of the International Conference on Advances in Computing, Control and Telecommunication Technologies ACT'09, December 28-29, 2009, IEEE, Thiruvananthapuram, India, ISBN: 978-1-4244-5321-4, pp: 664-668.

Randles, M., D. Lamb and A. Taleb-Bendiab, 2010. A comparative study into distributed load balancing algorithms for cloud computing. Proceedings of the 24th IEEE International Conference on Advanced Information Networking and Applications Workshops, April 20-23, 2010, Perth, Australia, pp: 551-556.

Rizvandi, N.B., J. Taheri, A.Y. Zomaya and Y.C. Lee, 2010. Linear combinations of dvfs-enabled processor frequencies to modify the energy-aware scheduling algorithms. Proceedings of the 2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing (CCGrid), May 17-20, 2010, Melbourne, Australia, ISBN: 978-1-4244-6987-1, pp: 388-397.

Rodero-Merinoa, L., L.M. Vaquero, V. Gilb, F. Galana, J. Fontanc, R.S. Monteroc and I.M. Llorentec, 2010. From infrastructure delivery to service management in clouds. Future Generation Comput. Syst., 26: 1226-1240.

Shu, W., W. Wang and Y. Wang, 2014. A novel energy-efficient resource allocation algorithm based on immune clonal optimization for green cloud computing. EURASIP. J. Wirel. Commun. Netw., 2014: 1-9.

Shu, W., W. Wang and Y. Wang, 2014. A novel energy-efficient resource allocation algorithm based on immune clonal optimization for green cloud computing. EURASIP. J. Wirel. Commun. Netw., 2014: 1-9.

Singh, S. and I. Chana, 2014. Energy based efficient resource scheduling: a step towards green computing. Int. J. Energy Inf. Commun., 5: 35-52.

Srikanth, G.U., A.P. Shanthi, V.U. Maheswari and A. Siromoney, 2012. A survey on real time task scheduling. Eur. J. Sci. Res., 69: 33-41.

Tsai, J.T., J.C. Fang and J.H. Chou, 2013. Optimized task scheduling and resource allocation on cloud computing environment using improved differential evolution algorithm. Comput. Oper. Res., 40: 3045-3055.

Ullman, J.D., 1975. NP-complete scheduling problems. J. Comput. Syst. Sci., 10: 384-393.

Zhao, C., S. Zhang, Q. Liu, J. Xie and J. Hu, 2009. Independent tasks scheduling based on genetic algorithm in cloud computing. Proceedings of the 2009 5th International Conference on Wireless Communications, Networking and Mobile Computing, September 24-26, 2009, IEEE, Beijing, China, ISBN: 978-1-4244-3692-7, pp: 1-4.

Zhu, R., Y. Qin and C.F. Lai, 2011. Adaptive packet scheduling scheme to support real-time traffic in WLAN mesh networks. KSII. Trans. Internet Inf. Syst. TIIS., 5: 1492-1512.