Energy-aware task scheduling based on the DAG applications using Genetic Algorithm in Cloud computing

Thesis submitted in partial fulfillment of the requirements for the award of degree

of

Master of Engineering
in
Software Engineering

Submitted By

Khushleen

(Roll No. 801531006)

Under the supervision of

Dr. Ravinder Kumar

Assistant Professor

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT

THAPAR UNIVERSITY

PATIALA – 147004

JULY 2017

Certificate

I hereby certify that the work which is being presented in the thesis entitled, "*Energy-aware task scheduling based on the DAG applications using Genetic Algorithm in Cloud computing*", in partial fulfillment of the requirements for the award of degree of Master of Engineering in *Software Engineering* submitted in Computer Science and Engineering Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of *Dr.Ravinder Kumar* and refers other researcher's work which are duly listed in the reference section.

The matter presented in the thesis has not been submitted for award of any other degree of this or any other University.

Signature Khushleen

Khushleen

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

13.07.17

(Dr. Ravinder Kumar)
Assistant Professor
Computer Science and Engineering Department

Acknowledgement

No volume of words is enough to express my gratitude towards my guide, **Dr. Ravinder Kumar** , Assistant Professor, Computer Science and Engineering Department, Thapar University, who has been very concerned and has supervised the work presented in this thesis report. He has helped me to explore this vast field in an organized manner and provided me with all the ideas on how to work towards a research oriented venture. His timely advice, meticulous scrutiny, scholarly advice and scientific approach have helped me to a very great extent to accomplish this task

I am also thankful to Dr. Maninder Singh, Head of Department, CSED for the motivation and inspiration that triggered me for the thesis work.

I would also like to thank the staff members and my colleagues who were always there in the need of the hour and provided with all the help and facilities, which I required, for the completion of my thesis.

Most importantly, I would like to thank my parents, friends and the Almighty for showing me the right direction out of the blue, to help me stay calm in the oddest of the times and keep moving even at times when there was no hope.

Khushleen

Khushleen
(801531006)

Khushleen
(801531006)

Abstract

Cloud computing has offered services related to utility aligned IT services. Reducing the schedule length is considered as one of the significant QoS need of the cloud provider for the satisfaction of budget constraints of an application. Task scheduling in a parallel environment is one of the NP problems, which deals with the optimal assignment of a task. To deal with the favorable assignment of some task, task scheduling is considered as one of the NP problem. In this research, the enhancement of DAG (Directed Acyclic Graph) algorithm has been considered for less energy consumption, more efficiency and less makespan and cost. For the optimization of the traditional scheduling and balancing algorithm, an algorithm has been designed for reducing the delay. For this, DVFS (dynamic voltage and frequency scaling) mode is applied that permits the devices for performing the required tasks with the less amount of required power. It also scale upwards for increasing the performance. The job placement also has a great impact on the cost computation. Here, the placement is done by using Optimization algorithm that is genetic algorithm for generating high-quality solutions for optimizing and searching the problems by depending on bio-inspired operator, namely mutation, crossover and selection. Metrics namely, make span, CCR(Computation Cost Ratio) and Energy consumption are used for the evaluation of the proposed work.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

DVFS- Dynamic Voltage Frequency Scaling

CPU- Central Processing Unit

IAAS- Infrastructure as a Service

PAAS- Platform as a Service

SAAS- Software as a Service

QoS- Quality of Service

MCC- Mobile Cloud Computing

DAG- Directed acyclic graph

GA- Genetic Algorithm

RBF- Radial basis function

DVAFS- Dynamic voltage accuracy frequency scaling

CU- Cloud user

CSP- Cloud Service Provider

RQ-Request Queue

GAQS- Genetic Algorithm Module Queue Sequencer

HCPT- Heterogeneous Critical Parent Trees

PEFT- Predict Earliest Finish Time

CCR- Computation Cost Ratio

Chapter 1 : Introduction

Energy supply should be diverse Modern times have seen unprecedented developments in the field of embedded systems and very high speed wireless networked technology with Mobile Computing devices like Smart phones, Smart watches (wearable devices) and Smart tablets etc. With the aid of wireless high speed technology a number of jobs are being performed with these smart devices [1].

But till date the Smart phones are using traditional power supplying materials with limited Battery life before preceding further it is imperative to define the following terms

1. Cloud Computing
2. Mobile Cloud Computing
3. Energy Efficiency and Energy optimization
4. Task Scheduling
5. Energy Model and their role in Acyclic graph
6. Dynamic Voltage Frequency Scaling(DVFS)

1.1 Cloud Computing

Cloud computing is almost everywhere whether we pick up journal or visit some Website of IT. Cloud computing is in talks all over. So much hype has been given to cloud computing but what actually cloud computing is? As per NIST definition, “Cloud Computing is considered as a model for having ubiquitous, convenient, on demand network access for different resources of configurable computing (like. Servers, storage, Networks, services and applications) that may be rapidly provisioned with the releasing of less effort of management or server supplier interaction”. So cloud let you access the information from anywhere at any time you don’t need to be at that particular location where the data is being stored. Cloud Computing provides service based utility like our normal utilities like water supply, Electricity, telephone services or gas supply. Soon it will be listed as basic necessity or amenity. In cloud computing users have to pay according to the services being consumed by them so consumers get profit. Services can be storage, networking applications. So consumers

are benefited as the cost associated with “in house” provision of all the above activities gets reduced. Cloud provider like IBM, Google, Amazon, Sales force are establishing new data centers which host applications of cloud computing world wide as the requirements for cloud computing differ. So service providers have to be flexible in the delivery of their services making the underlying infrastructure isolated from the customer [2].

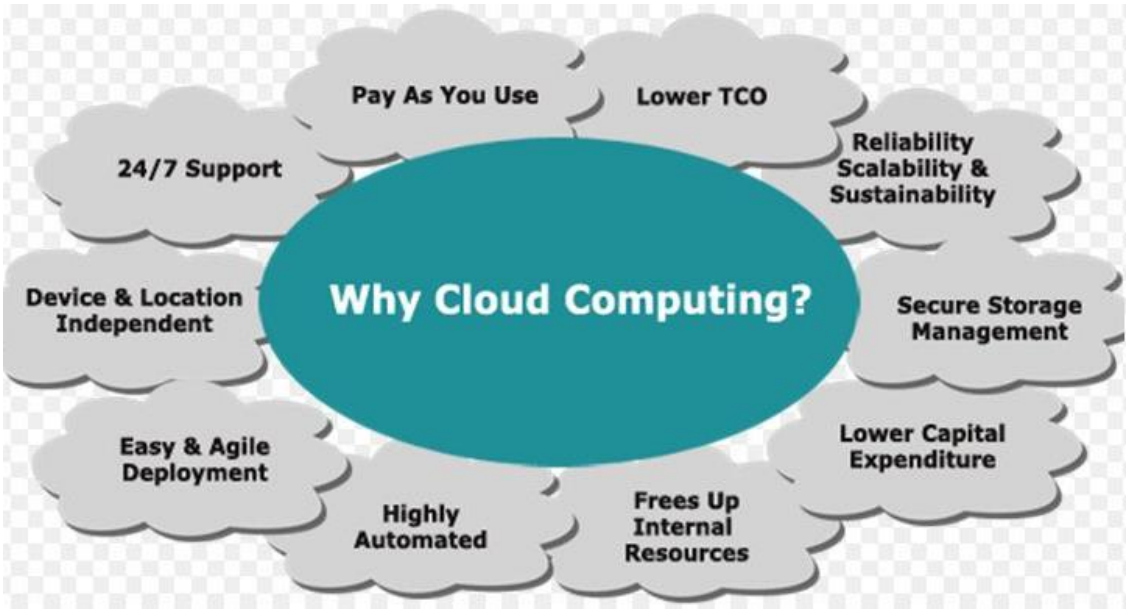


Figure 1.1 Cloud computing

NIST definition as following five important characteristics:

1. **On demand Self Service:** Resources may be provisioned by user without human interaction, they are mostly done through web based portal.
2. **Broad network access:** Cloud Computing systems are stored in the network. They support heterogeneous client platforms. For example. Smartphone, Workstations.
3. **Resource Pooling :** The resources on logical level are securely separated to provide service to multiple customers from the single physical resource and customers usually do not care about the physical location of resources but they should be aware about the risks when these resources are located offshore
4. **Rapid Elasticity:** The capabilities of the system can be expanded or they can be released automatically i.e great CPU power, or the ability to provide services to

additional users. For the customer it appears to be seamless or limitless according to the changing needs.

5. **Measured service:** Customers/users are charged according to their usage of application and meter concept is there which can record, monitor, control and report how much resource has been used by the consumer so it provides transparency for both the consumer and the provider of how much service has been utilized/provided. Users access data, applications or other services in cloud computing with the use of a browser regardless of user location and the devices in use. The infrastructure used with the help of internet provided by the cloud provider which significantly is the cost of computing task [3-5].

Less IT professionals are needed for implementation and maintenance is easy for cloud computing as there is no need for installing them on every user's Personal Computer.

The NIST definition lists following three service models of cloud computing:

1. **IAAS (Infrastructure as a Service):** This services are computing resources which providesstorage,processing, networks and another computing resource. Instead of providing raw hardware infrastructure. IAAS offers virtualized infrastructure. Virtualization is generally used in IAAS cloud to integrate/decompose physical resources to meet the increasing or reducing resource demand from cloud consumer. Virtualization's basic strategy is to setup Virtual Machines (VM's) which are independent and isolated from underlying hardware or any other VM's. so multiple cloud users can run on a single machine. Like. Amazon provides EC2 (Elastic Compute Cloud) for processing and for storage it gives S3 i.e. Simple Storage Service [6].
2. **PAAS (Platform as a Service):**these system givess applications on the internet and deliver hardware as well as software tools for the development of applications to its consumers as a service. The provider of PAAS gives both hardware and software on their own infrastructure. So users are freed from installing software/hardware to run/develop any new application. Most PAAS systems are aimed towards development of software and they provide several advantages to the developers. For example: PAAS allows its developers to

upgrade or modify the features of Operating System. Users generally access PAAS by a web browser and users are charged as per their usage [7].

3. **SAAS (Software as a Service):** SAAS helps the organization for eliminating need of installing and running applications on their own PC's or data centers, so it reduces the expenses of hardware usage and maintenance and software installation and its licensing. SAAS model have many benefits for example. Flexible payments in which user don't have to buy the software for installing and hardware as a supporting system. So the customers generally subscribe to any SAAS offering and the pay according to the services offered on monthly basis. Users are free to terminate/stop SAAS offering whenever the services are not needed and automatic updates of software are handled by SAAS offerings, Users/Consumers don't need to worry about them [8].

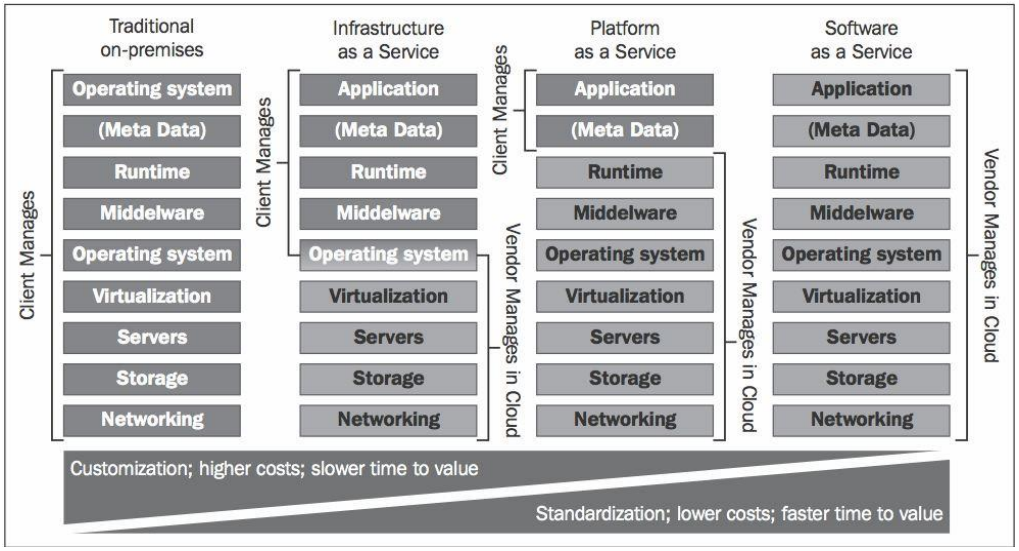


Figure 1.2 Cloud computing service model

Below are the four types of cloud deployment models:

A. Public Cloud: It allows its users to access cloud by the means of interfaces using web. Users have to pay accordingly to the duration of time for which they use the service. It can be compared to electricity billing system which we use in homes i.e. Pay as per their use. It helps in minimizing the cost of operation on IT expenditure. Public clouds are not secure/safe in comparison to other clouds as they are accessible to various malicious attacks because the public cloud is generally run by a single service provider. Many clients use the resources provided by one service vendor and customers are not aware of the

location of cloud infrastructure. Computing resources are shared between many clients. So they are liable to various malicious attacks. In order to save it from malicious attack security checks be made by cloud provider as well as user. Both vendor and cloud should understand their responsibilities [9].

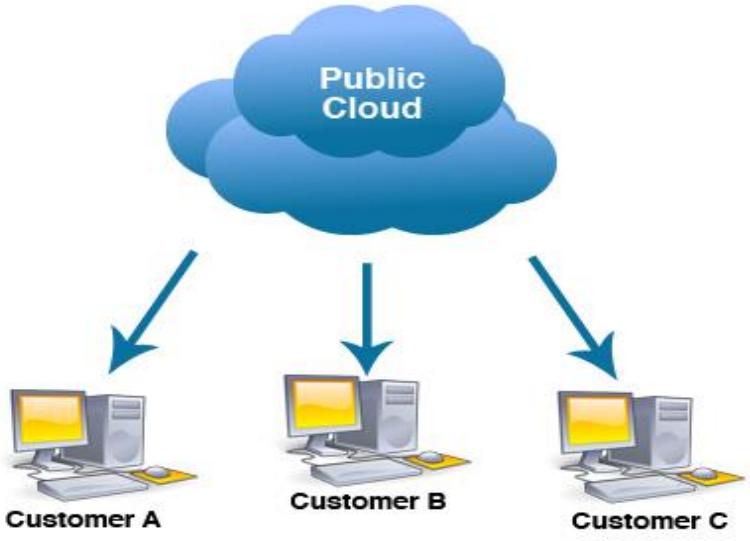


Figure 1.3 Public cloud

B. Private Cloud: A Private cloud is generally owned or operated within an internal data center of an organization. Infrastructure is not shared with other organization so it is easy to manage its security than public clouds. As it is under the control of an enterprise. Now, in the private cloud, the resources being managed by some organization itself so security is enhanced and Quality of Service (QoS) is also enhanced. In a public cloud all the applications and resources are managed by the provider where as in the private cloud all the services being made available toward its user by the organization itself [10-11].

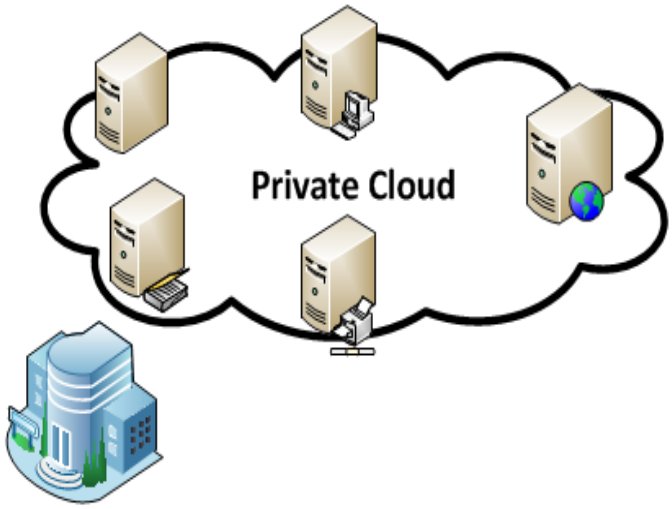


Figure 1.4 Private cloud

C. **Hybrid Cloud:** The public clouds with the private cloud are integrated together for forming a Hybrid Cloud i.e. private cloud could be linked to some external public cloud service provider. It is also a secure way to control over the deployment and use through hybrid cloud an organization can get services from a public cloud whenever the need occurs. So this type of cloud computing is generally used by many companies to cope up with the needs of the company for intensive computing resources and the hybrid cloud provides great deployment options. So hybrid cloud is valuable in dynamic and ever changing workloads. By a hybrid cloud model a company can run its business, can test and do queries on a public cloud, so it supports multiple computing tasks [12].

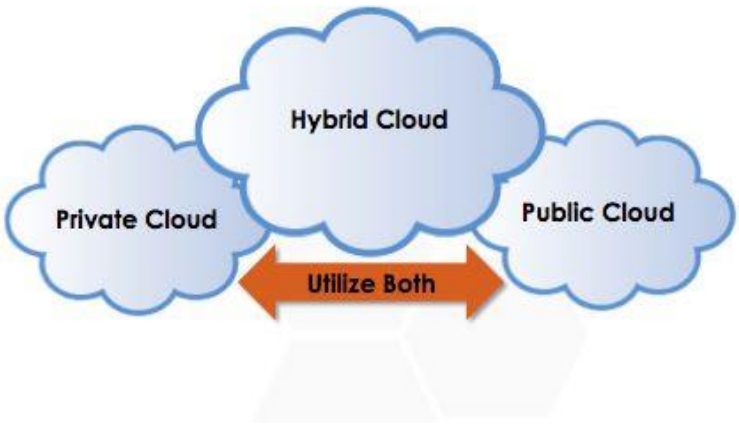


Figure 1.5 Hybrid cloud

D. **Community Cloud:** Community model is generally shared by several organizations having common needs or requirements/policies. In it cloud infrastructure is generally provided by a third party. Such companies work together as they have common objectives and want to achieve them [13-14].

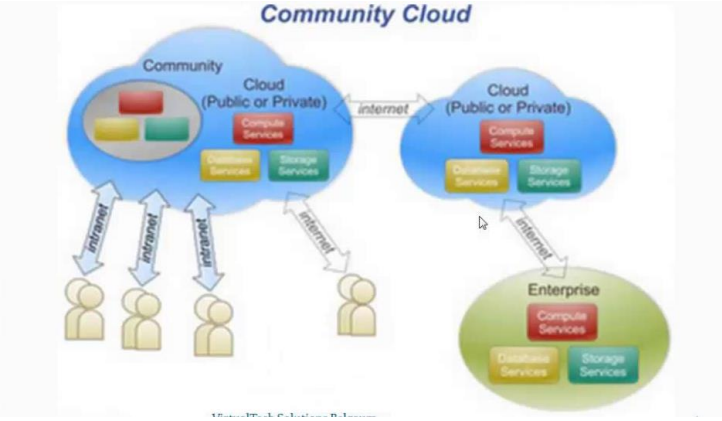


Figure 1.6 Community cloud model

Advantages of Cloud Computing

1. **Easy to manage:** By cloud it is easy to manage hardware/software infrastructure it means less burden on IT professionals. Moreover it is quite easy to store applications in the cloud as compared to those when they are used by an organization.
2. **Lowering computation cost:**Higher power or higher priced computers are not needed for cloud computing web based applications as the supplications are run in the cloud and not on any computer system. Cloud computer reduces the cost for SME's(Storage made easy) while using cloud based application computer system with small hard disk or less memory can serve the purpose. So less expensive computer system can be used moreover no CD or DVD is needed by computer system as no software programs are to be loaded and there is no need to save any document file. It is clear that an applications like Email can be set freely through Googleplay store
3. **Limitless storage capacity:** Cloud computing gives limitless storage capacity. PC's hardware of 300 Gbytes is very small when compared to several hundreds of Pbytes which are available in a cloud. So this feature of storage provides limitless storage capacity.

4. **Enhanced Data reliability :** In a PC whenever a hard disk crashes, it results in the destruction of all valuable data wherever when a computer gets crashed in a cloud then the stored data is not effective it mean in a cloud data is accessible even if your computer crashes. So we can say that cloud computing is more data safe in the case of hard disk crash.
5. **Universal access to documents :** In case of cloud computing one does not need to carry one's document with one self they can stay in cloud and one can access these documents whenever/wherever they are needed. One only needs to have computer or tablet along with internet connection. So all documents can be easily available at ay poit of time or place
6. **Availability of latest version:**The advantage of cloud computing is that it always provides the latest version of your documents as you can edit any documents whenever you need to edit. So there is no danger that only outdated versions are accessible.
7. **Easy collaboration of groups :** Many users share documents for better collaboration it can be easily provided by cloud computing as documents are present in the cloud not on any personal computer
8. **Independence of device:** One is no longer dependent on one's own personal computer or network by accessing the service of cloud one can easily access one's data through his Smartphone, tablet or any other portable device.
9. **Green Computing:**Green cloud is ecofriendly services being offered by ICT over the internet. Here word green means environment friendly and cloud is the service delivery system called cloud computing and huge amount of energy is being consumed by Data centers which are created for cloud computing. They also resulted in high operational costs and emission of excess of carbon dioxide (Co_2) in environment. We need Green Computing as a solution to save energy and also to reduce operational cost hence enriches environment sustainability [15,16].

The issues related to Cloud Computing:

1. **A constant internet connection:** Cloud computing cannot be operated if one does not have any internet connection. So no internet connection will lead to non availability of the documents or any data, so it means no task can be done in the area where internet connection is not available. One has to be online to

work on cloud computing. Low speed internet connection does not serve the purpose as cloud application need broad bandwidth to do any task

2. **Security and Privacy concerns:** The safety and security is the biggest concern of cloud computing as one has to handover his/her confidential data to the providers that why various users generally hesitate to transmit their data to cloud providers and want to keep their data with themselves. However the data stored in a cloud is safe and can be replicated by multiple machines, but in case data goes missing one has no means of backup. But the cloud provider do not want to lose their users/client for security measure and they promise to provide best security services to their users and they issue an authorization.

1.2 Mobile Cloud Computing(MCC)

As per the literature available with IBM Mobile Cloud Computing is the technique which uses cloud computing to deliver applications to mobile devices like tablets, smart phones, and another wearable devices. The mobile apps can be deployed remotely with speed and flexibility. It enables the user to accessed applications that could not otherwise be supported by the devices mentioned above [17]. The mobile cloud offers the following three advantages

a. **Shared Resources :**

   Mobile applications which use the cloud are not restricted by the storage capacity of the device and the processing resources. Mobile Cloud Computing allows data intensive processing

b. **Speed and a great deal of flexibility :**

   Mobile Cloud applications can be delivered to diverse number and kind of devices which use different Operating System

c. **Integration of data :**

   MCC also allows the users to collect and integrate data from diverse and various sources with a great speed and security

   MCC offers an architecture pattern where data storage and intensive processing tasks are performed on cloud and the smart/mobile devices mainly acts as a thin client to interact with the apps and for fetching the processed outputs from the cloud, since ordinarily mobile devices are restricted by storage, battery life and processing power, the above said architectural pattern

allows smart devices to operate without the above said constraints. It may also be appreciated that currently mobile applications exist in three different kinds.

1. Mobile applications meant for particular operating system or platforms such as IOS and Android to access certain features like GPS, Camera, Accelerometer and other related sensors.
2. Mobile web applications which use CSS3 , frameworks like JQuery, HTML and Twitter bootstrap etc which run from web browser on smart device with the aid of internet connection and do not require any installation as in application type mentioned above.
3. Also available are certain tools like PhoneGap or Cordova or IBM mobile first which allow the applications to use cross platforms thus reducing the cost of development by eliminating codification for each separate platform [18-21].

1.3 Energy Efficiency and Energy Optimization

Since in modern times Mobile devices are being used to perform the functions which hitherto work performed by desktops hence arrives the need to use energy available for a device very efficiently and effectively. The energy consumption may be reduced by

(i) By lessening the energy spent on processing by avoiding the use of local Central Processing unit.

(ii) By limitingthe data transmission and ideally using high speed networking and using cloud servers to process heavy tasks through Offloading Scheduling algorithms.

The above two factors make it imperative to identify the tasks which are suitable for offloading as well as task allocation algorithms. Data compression techniques may also result in energy saving and hence an efficient energy model can also be developed for each task needless to say that simple tasks don't take much of the time and hence there is no need of offloading only complex applications and computational tasks are required to be offloaded and afterrendering the data would be transferred back to mobile device. This would save energy used on computational part but add extra cost of data communication thus network bandwidth would be a constraint. Some researchers argue that it is better for compressing the data on the mobile device

as compare to its cloud servers as it will reduce data size and save communication time [22-24].

1.4 Task Scheduling

In scheduling problem we see that some tasks are dependent on certain other tasks. So they are to be completed before starting the next task. In such a condition one has to decide the order in which these tasks are to be performed keep in view their dependency. This way of ordering tasks is known as Topological Sorting.

With the increase of use in the system of Cloud computing that has resulted in the increase of Task Scheduling in Cloud. Cloud computing makes use of low power providers to provide high usage. So selection of nodes for performing a task in cloud should also be considered, here we should know what is a task? It is an activity which use certain defined inputs to produce outputs accordingly. In cloud user applications are generally run on virtual systems/machines. Dynamic load balancing has to assign tasks dynamically to the processor. During the overloading of processors the task are redistributed. All the applications are completely different from each other in nature and are also independent of each other. Some need more CPU time for computation of task whereas others may differ in the need of memory i.e they need more memory for storage of data. The scheduling algorithm can increase the efficiency and can maintain the load balancing of a particular system. The efficiency of cloud is dependent on the algorithms which are used for the purpose of task scheduling. From the discussion carried out above we have to make a decision regarding each task as to whether it would be prudent to perform it on the device or to process it on the cloud and transmit the results rendered to the smart device. So as to use energy in an optimum manner since each task requires execution/processing of certain instructions, generation and transmission of data hence we have to keep in mind the following factors in Task scheduling

(i) The tasks which may be executed on the mobile device.

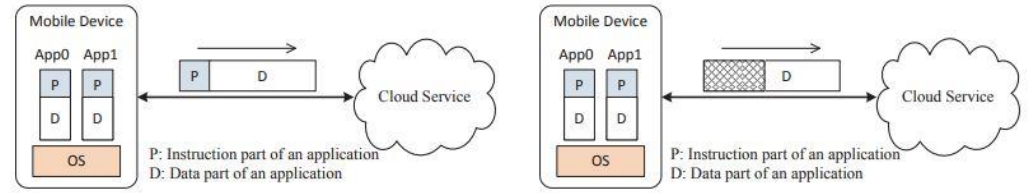
(ii) Every task might have a rigid compression ratio of output and input data and each task might have different processing time.

(iii) The network bandwidth available among the mobile device with the remote cloud.

The energy consumption in a smart device would thus depend on the following two types of load

- Computational and Processing task
- Communication and Data transmission cost

Thus we need to develop an algorithm to make wise choices in offloading. So as to save energy and time at the same time [27-28].



(a) Offloading: mobile device needs to transmit all data to server

(b) Non-offloading: the mobile device just transmits the executed data

Figure 1.7Offloading and Nonoffloading

Clearly Energy optimization requires optimum choices between communication and computational tasks.

1.5 Energy Model

In energy model following factors related to energy consumption will have to be fit in a mathematical model

1. Energy used to process the data and the Results.
2. Energy used to transmit the data and receive the data and energy required to maintain the communication thus a mobile application will have to be broken/split into
   (a) Instruction to be executed
   (b) Input data used and Output data generated
   (c) Computing process
   (d) Transmission of data

Let P_c be the energy consumed for processing an instruction in a task and P_t be the energy used for transmission of data. We make a choice to perform the task on the cloud than the wenergy consumption will be given by the following formula-

$E_{cloud} = P_t + D/B$ (where D is data in bytes to be processed and B is network bandwidth)

If we presume that the output generated is D' then it can safely be presumed that D' would be smaller than D. Now the mobile device/smart device performs the task itself than the energy consumed would be

$E_{mobile}=P_c*I+P_t*D/B$

Here I is the numerals of instructions for the execution in a task

Energy saved will be (E_s) given by $E_s=E_{cloud}-E_{mobile}$

$= P_t*D/B - P_c*I - Pt*D'/B$

$= [P_t*D/B]- P_c*I - [Pt*(D*K)]/B$ (where K=D'/D and K is called compression ratio)

$= P_t D/B (1-K) – P_c * I$

Energy will be saved if this equation is positive and only then it would be preferable to offload the task [29].

1.6 DVFS (Dynamic Voltage Frequency Scaling)

In modern times various algorithms has been developed for optimal power utilization in the cloud and DVFS (dynamic voltage and frequency scaling) schemes has been developed. Once a decision to offload a particular task is taken and a job is allocated to the victimized host. The required clock frequency of the host is measured and according to the measured time complexity the Central Processing Unit(CPU's) frequency is scaled up or down using DVFS scheme in order to save energy. Thus DVFS is a power conservation technique where by overall power consumption is reduced to enable effective cloud computing by adjustment of frequency of the processor according to the requirement. In DVFS CPU's frequency can be dynamically called according to the load and in general it can be activated in four different nodes-

1. High Frequency
2. Low Frequency
3. Available Frequency for the CPU

4. Dynamically selecting the frequency level on demand

First of all we must classify the incoming request as to whether it is CPU bound or input, output memory. The incoming request follow one or the other algorithm and every algorithm can further be classified into time, linear time algorithm, logarithmic algorithm, polynomial time algorithm with exponential time algorithm etc. In a CMOS integrated circuit such as modern CPU. The Power consumption can be reduced as by reducing frequency as given below

$P = CfV^2+P_{Static}$ (where C is considered as the transistor gates capacitance, f is frequency of the operating and V is the supply voltage and static denotes that it is not performing any operation)

The voltage required for normal and static operations is dependent on frequency on which the circuit is being clocked. This voltage may be reduced by reducing frequency. This may result in important reduction in power consumption by reducing voltage. It is pertinent to mention here that energy can only be saved if the power consumption is reduced enough to run the workload, to cover the extra time it takes to process at the lower frequency. Thus it is trade off between energy saving and time saving.

The Energy E is given by Power*Time hence E is proportional to V^2

$E\alpha V^2$ ($V \alpha f$)

Thus by lowering CPU frequency energy can be saved by providing just enough computation power.In order to maintain the minimal system energy through DVFS we have to keep in mind the following factors-

a. Timing Constraint – Different applications take different execution time hence we need to develop accurate task execution time node as a function of CPU frequency. Secondly power consumption of each component i.e active or ideal state must also be known in advance when these two factors are coupled with work decomposition model we can get optimal system energy with DVFS [30-31].

1.7 DAG (Directed Acyclic graph)

DAG is a directed graphs which has no cycles. It represents parallel tasks. Dag has great importance in the field of computer science because it analysis a task scheduling and its concurrency control. When a program is distributed across number of processors then one may be in trouble when output from one is not generated and the other needs it as its input. So DAG is the solution to such a problem which can be represented diagrammatically below:

DAG modules have been mainly used in multi-processor based on parallel system. It is a directed graphs which have no cycles [4].

A directed graph consists of direction as well as a lack of cycles.Figure 1.8 has number of parts as defined below:

Integer = Vertices set

Vertices_set= {1,2,3,4,5,6,7,8,9,10}

Edge_set= { (1,2,3,4,5,6), (3,7,8,4,9), (4,8), (5,9,7), (6,9), (9,10), 98,10), (7,10)}

A DAG graph was the combination of vertices and edges. As shown in figure below the vertices representing subtasks and the edges denote execution precedence between subtasks [32].

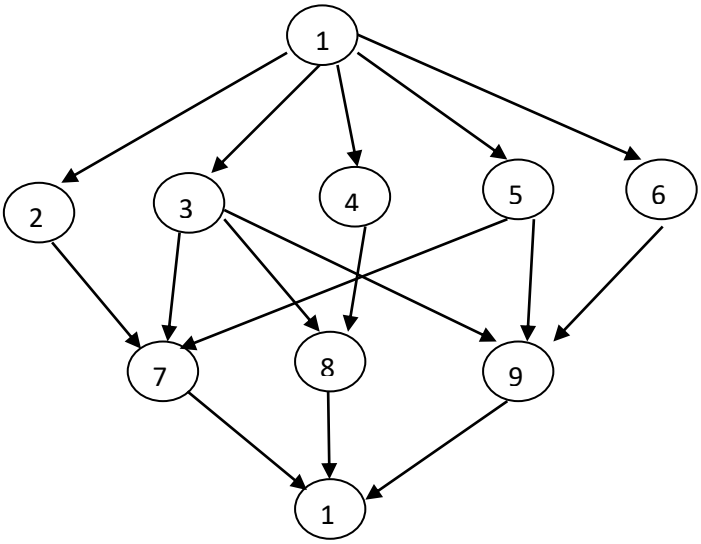


Figure 1.8 An example of DAG

DAG is termed as the tuple where G=(V,E) in which V stands for Vertices which represent tasks and these nodes are independent in terms of data and E stands for Edges that represent Communication cost. In such a model the nodes are defined as directed vertices as of the parent task to child task with the child task execution cannot that commence till parent tasks are completed. The outputs of parent task act as inputs for the child task. One has to develop algorithms to typically focus on reduction of execution times and the execution task so as to achieve optimal performance and energy levels.

1.8 Genetic Algorithm (GA)

GA was invented by Goldberg in 1989 which is generally utilized for the applications where the search space is more. The benefits of GA are that the procedure is considered as purely automatic and has no local minima. The main components of Genetic Algorithm are: mutation, crossover and a fitness function. The crossover operations are utilized for generating a novel chromosome from parent sets and the operation of mutation is that it adds variation. The fitness function implements a chromosome dependent on the basic that are already defined. The chromosome improved fitness value enhances its survival chance. Chromosome collection is the population. A novel population is done by utilizing general genetic operations like mutation, single-point crossover and selection operator [33].

The procedure of the implementation of the genetic algorithm is defined below:

Step 1: Initialization of random population having chromosomes.

Step 2: Calculation of the fitness function intended in the population.

Step 3: Development of novel population with individuals.

Step 4: To select parent chromosomes for enhanced fitness function.

Step 5: To execute crossover for having parent's copy.

Step 6: To execute mutation for mutating new off springs.

Step 7: To have novel offspring intended in the population.

Step 8: Repetition of the steps to have a satisfied solution.

Step 9: Stop

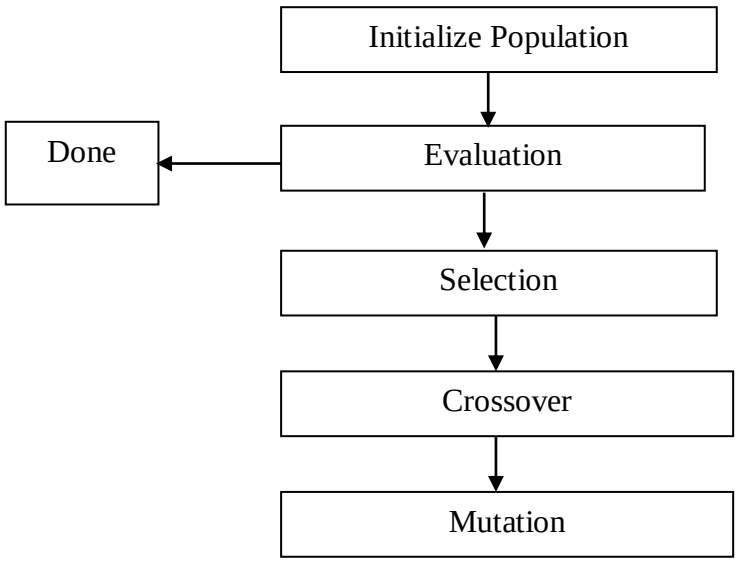


Figure 1.9 Flow Chart of Genetic Algorithm

i. Representation

There are two components in Genetic Algorithm that are problem dependent, namely, Evaluation functions and Representation. Representation is taken as the GA key as GA manipulates the representation of the coding for the problems. So, the coding scheme and character set is used. The usage of binary set is that it has prevalent schemata for any defined matrices resolution for enhancement of the implicit parallelism. In GA, the individuals are shown as the fixed length strings that shows the fixed length strings for expressing fixed binary strings showing schema as pattern for numeral {0,1,*} and depicts a binary strings set in search space. So, every string consisted of2^L schemata, where L is known as the binary string length [34].

ii. Evaluation Function

The evaluation function is also problem dependent beside with the representation scheme. GA is said to be a search technique which is dependent on the feedback received by the study of the solutions. The judge for GA exploration is known as Evolution function. The idea for fitness and evaluation are occasionally utilized interchangeably. It is important for differentiating the fitness function and evolution

function. The evolution function provides a measure for user's performance, and the fitness function gives a calculation for individual's performance. The assessment of individual is independent of another individuals and the fitness of an individual mostly dependent on another individuals.

iii. Initial Population

To choose a suitable population size for GA is essential but complex tasks for all Genetic users. If the population is small then the GA converges quickly for finding the optimal solution and if the size of population is too large then the cost of the computation might be excessive.

The initial population in the proposed algorithm would be total number of nodes over which the optimization has to be performed

Pop_size=numel(Node_Values)

Numelcalculates the total number of identities in the network

iv. Operators

According to the mechanistic view, Genetic Algorithm is an iterative method with the iterationsof two basic steps, namely, Generation and Evaluation.

- While evaluation, the data of the domain is utilized for evaluating the quality of an individual.
- In case of selection, fitness is utilized for guiding the reproduction for novel candidates for the subsequent iterations.

Fitness function maps a user to a unique number for indicating the number of offspring that a user is expecting to strain. The high fitness users basically have more emphasis on the following generations because these are selecting more frequently. When the recombination phase started, the crossover and mutation performs integrated. The crossover reconstructs selected individuals for creating two new offspring.

.

The main focus of researchers is on three operators, namely, Mutation, Selection and Crossover.The selection is always as per the fitness, the mutation and crossover are the investigating resources. Genetic integrates the utilization of existing results with the newfields of the search space. Genetic Algorithmis dependent on the exploration and exploitation as well.

Below are the operators of Genetic Algorithm

a. Selection

This operator has asignificant role in motivating the searching for best individuals and for keeping anenhanced genotypic diversity intended in the population. It has been noticed by some of the authors that the phase of selection may be divided into sampling and selection algorithm. The selection algorithm is utilized for assigning every individual a real number, known as 'target sampling rate' for indicating the expected offspring number that use to reproduce by the time. The sampling algorithm basically reproduces, dependent on the target sampling rate, individual copies for producing the population. There is a difference between probabilities of individual's actual sampling with the outcome value. The dissimilarity is known as Bias.

Generally, there are two selection algorithms types, namely, implicit fitness remapping and explicit fitness remapping.

The explicit fitness remapping re-maps the fitness inthe newscale thatis later utilized by the sampling algorithm. Proportional selection and Fitness ranking belongs to this field. The other fulfills the matching pools instead of passing via the intermediate remapping step.

The selection function here in genetic algorithm is performed by the hybrid algorithm. The selection function checks that which node should be selected for the processing .Hybrid algorithm utilizes the concept of search and provides selected nodes only.

Selected_Nodes=Search_Result(Hybrid_Routing)

b. Crossover

For exploring other point in the searching space, the variability is proposed for the middle population for few idealized operators of genetic re-combination. Crossover is

known as mainly the significant recombination operator. One-point crossover is a general used method that selects two individuals in the population that exchanges the representation portions. Assume crossover of one point as an example. Individuals are depcited as binary strings.

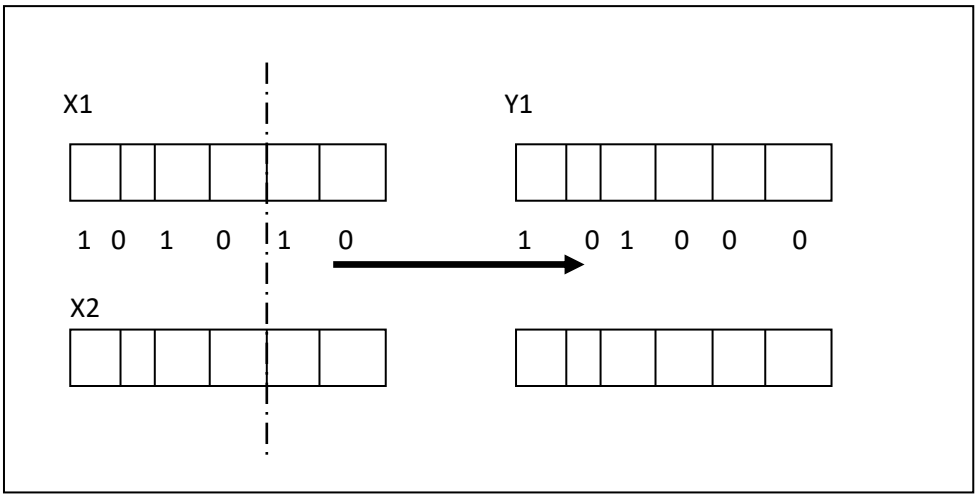


Figure 1.10 Crossover

Within single point crossover, a point is called as Crossover point being chosen randomly with the segments to the right of the points chosen randomly and the exchanging of the segments to the right is taken place.

Let's take an example, in which 'x1=101010 and x2=010100' and take the crossover points among 4 and 5 bits. The example is shown in the figure 1.10.

Crossover has two matching search abilities. Primarily, it has new points for testing the hyper planes that are already shown in the population. Secondly, it has representatives of novel hyper planes in the population.

c. Mutation

When the users are shown as the bit a string, then the mutation is consists of reversing a bit which is randomly chosen. Let takes an example, when the individuals are taken as binary strings. When the bit is selected for mutation then it will be projected to match the general bit value in the bit complement. If x1=101010 and the mutation bits are 4, next the child is 'y1=101110'. Below picturedepicts this example.If the number has bits group in the string, minute varies in the values can be followed from such

mutations. It prevents the GA from refinement of the solutions after discover the better solutions in its neighborhood.

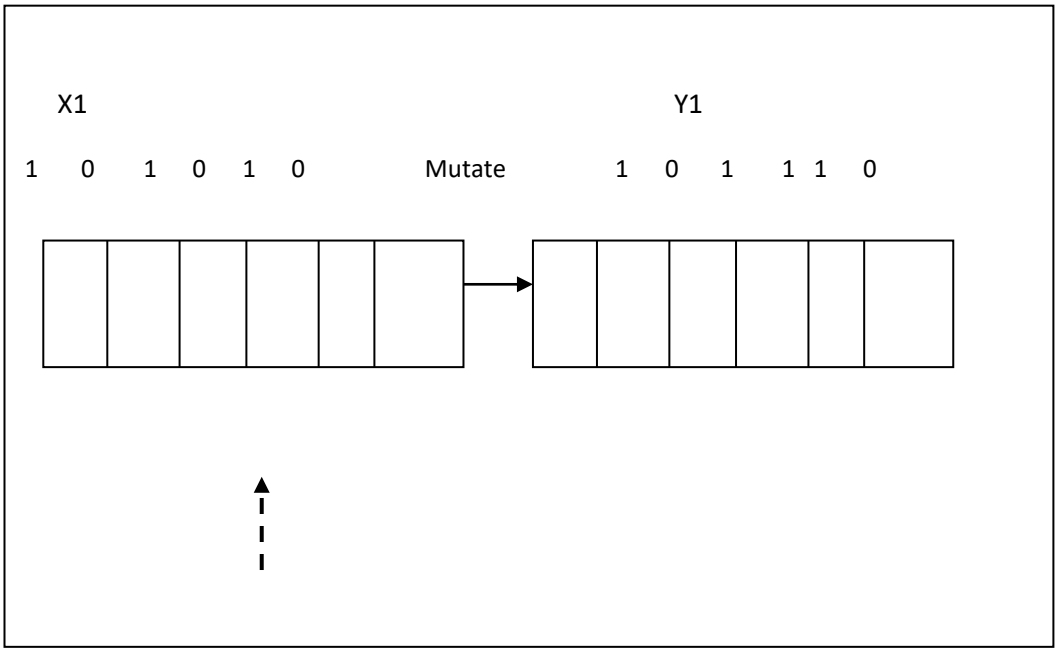


Figure 1.11 Mutation

The GA implements mutations by the option specified in the Mutation function. The default option for mutation is consider asGaussian, and it integrates a mutation or random number, taken as of a Gaussian distribution, to every entry of the parent vector. Usually, the mutation amount, being proportional to the generaldistribution deviation, lessens at every novel generation [3536].

E. Zhao et al. [37, 2017] presented a fusion framework which is dependent on the metric paradigm. This was evaluating the efficiency and quality of the existing projects and software products. The simulation has been done in cloudSim environment. Authors has used three angular metrics that are product metrics, project metrics and organization metrics. Simulation results showed that the framework was efficient, effective and operational. Authors used Radial basis function (RBF) based prediction algorithm this is similar to neural network with three layers, such as, input layer and hidden layer with the output layer. This algorithm was significantly improves the precision and quality of service in cloud computing.

Tao Lin et al. [38,2017]proposed an iterative algorithm for solving the problem of multi-objective optimization. Authors adopted a Game Theoretical approach for managing the data traffic so that energy can be optimized. Authors worked with transport layer to form an energy efficient framework in cloud computing.

X. Wang et al. [39, 2017] presented the mobile-application offloading for mobile cloud computing. The energy consumption of mobile devices has been reduced by offloading the mobile applications. Howe ever, it will add more implementation delay of the mobile applications. To resolve the tradeoff among execution delay and energy consumption, the authors formed an optimization algorithm known as the solution of polynomial time optimal being dependent on the problem of bipartite matching.

F. Azimzadeh and F. Biabani [40, 2017] presented a method used for resource management and task assignment in cloud computing thus reducing the time and increase the reliability for the system. The algorithm used will optimized the scheduling of individual tasks in cloud and assign the tasks to the resources in optimal time period. The simulation of the proposed work has been carried out in MATLAB environment.

R. Singh and M. Agnihotri [41, 2016] discussed about the job allocation to number of virtual machines inside a cloud data center by using Map-K loop abbreviated as MKL which is a classical scheduling policy. By using the map, the jobs were assigned

to the suitable Virtual Machines, and hence, generally job's waiting time and the response time get reduced.

K. S. Bahwaireth et al. [42. 2015]address the energy efficiency in MCC (mobile cloud computing) because it was necessary for the designing of mobile devices. MCC technology could reduce the consumption of energy in mobile devices. This will possible by reducing the processing time and thus power consumed also get reduced. Authors used different techniques like offloading, green MCC, cloudlet architecture and clone cloud.

Liang Luo et al. [43, 2012]considered relationship among the components of infrastructure with the power consumption of the environment of the cloud computing. A task scheduling algorithm has been used to optimize the energy efficiency.

P. Arroba et al. [44, 2015] proposed DVFS (Dynamic voltage and frequency scaling) policy that was used for solving the tradeoff among performance degradation and energy consumption. Also provide a new consolidation algorithm that as used to maintain QoS parameters like energy. In this paper energy has saved upto 39.14% under dynamic workload.

B. Moons et al. [45, 2017]introduced about DVAFS (Dynamic voltage accuracy frequency scaling). It is a circuit based method that solved the trade off between energy verses computational accuracy. This paper showed that how deep learning could be brought to IoT devices.

Abdul Razaque et.al [46, 2016], proposed a algorithm of task scheduling using a non-linear model of programming for different task scheduling that gives the tasks of correct number to every virtual machine. On the basis of this assignment, an algorithm for different load scheduling is designed forhaving the network bandwidth.

Dr. Amit Agarwal et.al [47, 2014], proposed Generalized Priorities. The algorithm was used to effectively perform tasks and assessment with Round Robin Scheduling and FCFS. The algorithm was simulated in the toolkit of cloud Sim and the results show that it provides better performance with the existing scheduling algorithms.

Raja Manish Singh et.al [48, 2014], proposed different algorithms that were compared and studied Adaptability, feasibility, adaptability in the context of the cloudSim, after which the author is tried to propose a hybrid approach can be used to further strengthen the existing platform and so on. It can help cloud providers to provide better quality of the service.

R. Durga Lakshmi[49, 2016], proposed Genetic algorithm to optimize the overall system efficiency. Cloud applications constitute QoS (quality of service) management, which is the distribution of the problem. Resources to ensure service levels along the dimension of performance, availability and so on and reliability were discussed. For improving the QoS in the system, the system's waiting time must be reduced. GA (Genetic algorithm) is known as a heuristic search method, producing the enhanced solution of the task. The proposed work has generated GA-based scheduling algorithm to optimize the full system latency. Cloud environment has been categorized into two parts, initial is the cloud user (CU), and the other is the CSP (cloud Service Provider). The CU transfers a service request towards CSP, and the requests are being stored in the RQ (request Queue) communicates directly with the GAQS (Genetic Algorithm Module Queue Sequencer) within the CSP.

B. Saovapakhiran et al. [50, 2011] used Batch based DAG scheduling that will increase the throughput and utilization proved experimentally as well as theoretically. Experimental results showed that the parameters like delay and throughput when communication time and costs constraints have been considered was better than the HEFT algorithm.

Y. Wang et al. [51, 2014]proposed a multiple DAG scheduling algorithm that will considered the link of communication processor.Initially, this algorithm will take the communication completion model to connect the communication between different processors. Authors focused on the problem of scheduling in distributed computing system. By using DAG's task model the execution time get reduced.

D. Kliazovich et al. [52, 2013] proposed a novel model of communication aware used for the applications of cloud computing known as CA-DAG. The technique is similar to DAG (Directed Acyclic graph) that utilizes vertices for representing the communication. CA-DAG will increase the efficiency of the available schedules. The proposed model of communication-aware develops the space for optimization for related solutions for resource allocation and newly developed new scheduling schemes of enhanced effectiveness.

Table 1.1 Comparison of existing techniques

Author	Existing techniques	Outcomes
Haijun Cao et al. [53,2008]	DAG model, MinMin and Max-Min selective scheduling have been used for tasks independently.	DAG map has providedenhanced performance as compare to other previous algorithms by means of speed up, makespan and efficiency.
Y. Liu et al. [54, 2015]	Directed acyclic graphs (DAGs) scheduling	The basic objective is the improvisation of resource usage for exploiting the time chips.
H. Arabnejad and J. G. Barbosa [55, 2014]	Predict Earliest Finish Time (PEFT), DAG scheduling	Different parameters like SLR, CCR have been calculated.PEFT has lower SLR than other algorithms
T. Hagras and J. Janecek [56, 2003]	Heterogenous Critical Parent Trees (HCPT)	Experimental results showed that HCPT has given comparable andenhanced results together with less complexity.
A. K. Amoura et al. [57, 1998]	Gaussian elimination method, partitioning, mapping, scheduling	Simulation results showed that both generalized coumn oriented scheduling (MGCO) and Block scheduling (BS) have better results than DSC
L.F. Bittencourt et al. [58, 2010]	DAG scheduling heuristic, Heterogeneous Earliest Finish Time (HEFT)	Lookahead versions proposed can enhance the schedule returned by HEFT, normallywhen the communication-cost is

		more than the computation-cost
C. Boeres et al. [59, 2014]	HEFT algorithm	HEFT algorithm was used for increasing the speed of processors.
Oliver Sinnen [60, 2003]	DAG and communication contention.	Proposed work is only applicable for heterogeneous cloud environment using directed acyclic graph.

Chapter3: Problem Statement

Cloud is becoming a very popular platform for the storage and operational events. Scheduling is one of the major operations in any platform. Energy Prevention is one of the hottest issues in present era which is affecting the global environment. The problem of this research work is used to design a scheduling algorithm in such a manner that the consumption of energy for a graph G(V,E) in which V is the vertex and E as the edge set, on the completion of all jobs is least . The problem statement of this research work includes the enhancement of DAG algorithm in order to attain minimum consumption of energy and maximum efficiency with minimum makes span and cost. The problem statement also includes the utilization of the energy model to make the process energy efficient. Energy Consumption has been a point of discussion for last couple of years. A lot of architectural algorithms like DAG (Directed Acyclic Graph), HEFT (Heterogeneous Earliest Finish Time) have already put their impact for the same. The above mentioned algorithms hardly focus on energy minimization. The proposed work is going to optimize the traditional scheduling and balancing algorithm by making the following changes in the traditional process.

1) Selection of the Task / Node to be executed: The order in which the jobs would be executed plays an important role for reducing the load and total execution cost of the architecture. Assume that a job has 3 dependencies and it cannot start before all the dependencies have executed, putting this job in an upper queue would create an unnecessary delay and addition in the total cost of the network. An algorithm has to be designed to reduce the delay of the architecture. DVFS energy model has to be applied to check the energy efficiency of the proposed algorithm.

2) Order of Placement of Job: Placement of job at any processor also has an essential role in the cost computation. Here, the placement has to done by avoiding the traditional method and opting Genetic Algorithm. The parameters of the evaluation would be as follows:

a) Energy Consumed

b) Make span

c) CCR(Computation Cost Ratio)

Chapter 4: Objectives and Methodology

4.1 Objectives

The aims and objectives of the research work are defined below:

i. To design and implement an enhance job selection algorithm for the minimization of the Energy consumption and cost.

ii. To use GENETIC ALGORITHM for the optimization of processor selection in order to balance the load and minimize overall makespan.

iii. To evaluate the computation parameters and for comparing the results with the previous implemented scheduling algorithm and to make it energy efficient

4.2 Methodology

Consider the following diagram

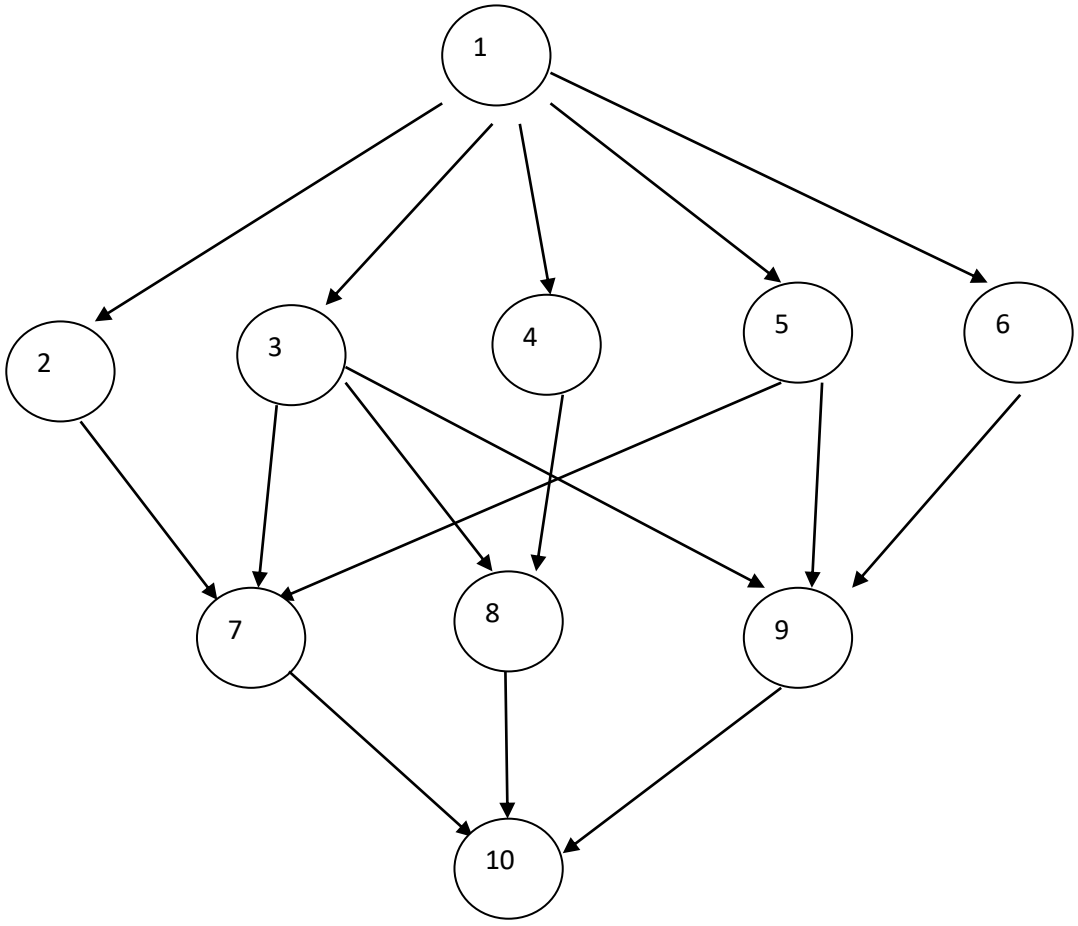


Figure 4.1 Tradition problem of load balancing and the cost computation.

The above diagram shows a tradition problem of load balancing and the cost computation.

. ------> Arrow represents a child node. Multiple ----> arrows represents multiple parents. As for example, Node 7 has three parents namely 2, 5, 3. Placing node 7 above of node 8 who has only 2 parents 3 and 4 will produce more delay into the architecture. The flow diagram of overall work is as follows.

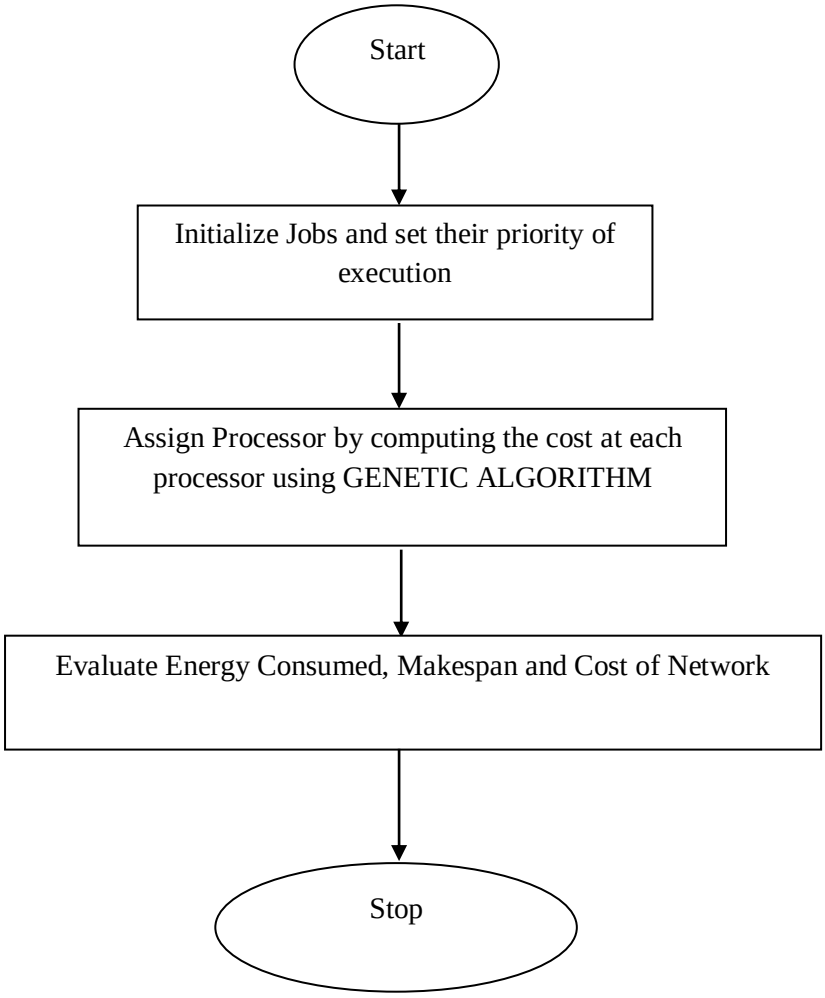


Figure 4.2 Proposed work Flow

4.2.1 Algorithms

1. Calculate the average execution time for every task $t \in T$ as per equation 6.
2. Calculate the average data send time among tasks with their successors, according to equation 7.
3. Calculate rank value for every task according to equation 8 and 9.
4. Arrange the tasks in a scheduling list Q by reducing the order of task rank value.

5. Whereas Q is not empty do.

t← eliminate the initial task from Q.

r←Calculate a resource which may finish t as earliest time

6. Schedule t to r.

7. End while.

The computation time of task is defined by the weight on the processor that reveals more information about the processor and always shows the improvement in the various parameters w.r.t time.

$$weight_{ti} = \frac{abs\ (w(t_i, (p_i) - w(t_i, (p_k))}{w(t_i, p_i)/w(t_i, p_k)}$$

Here, $w(t_i,(p_i)$ is the computation time of task t_i on processorp_j.

4.1.1 Scheduling Algorithm

1. For each level Zo do $w(t_i,(p_i)$
2. For each task Yo do
3. Find average com_cost, Max_Data Arr_Cost, and Exp_time
4. Sort all tasks from the level Z0 as per its priority
5. For every processor al in the p_set do
6. For every processor al in the p_set do
7. Compute EFT and EST of y_k on p_set

For simulating the experiment, two stages which are for scheduling that are Processor selection Phase and Task Prioritization Phase.

1st Phase: The initial phase broadly comprises of two levels that are:

i. The level in which the tasks priority at each level is computed, known as Level wise task priority.
ii. The level from where the selection of tasks from all the stages uses to take place on the basis of priority, known as Task Selection.

2nd Phase: The tasks are basically assigned to the finest processor that lowers the completion time.

A. Task Prioritization Phase

The dependence of the organized schedules length is on the generation of the tasks. The organized task list is obtained for the execution with the two phases that are given below:

Phase 1: Level wise Task Priority

The priority of the available task is executed by the standard computation cost and the greatest data arrival cost at every level. With the DAG having n tasks and p processors, the ACC (Average Computation Cost) of task T_1 is described with the division of total of computation cost of task on every processor with the processors.

ACC (t_1) $= \bar{X_1} = \sum_{k=1}^{n} x_{i,j/p}$

$X_{i,j}$ is the approximate executing time to finish the complete task T_1 on the processor p.

With a DAG having t tasks and f edges, the MDAC (Maximum Data Arrival Cost) of the task mt_j is the maximum time with the task which has to deplete the data among the parents.

$$MDAC(mt_j) = max_{mtiepred(mt_j)}(c_j)$$

where, t_i is the predecessor task of the set.

Table 5.1 Tools Used

Computer	Core 2 Duo or higher
RAM	3 GB
Platform	Windows
Other hardware	Keyboard, mouse
Software	Cloud Sim

5.1 RESULTS AND DISCUSSION

The following results have been observed for this

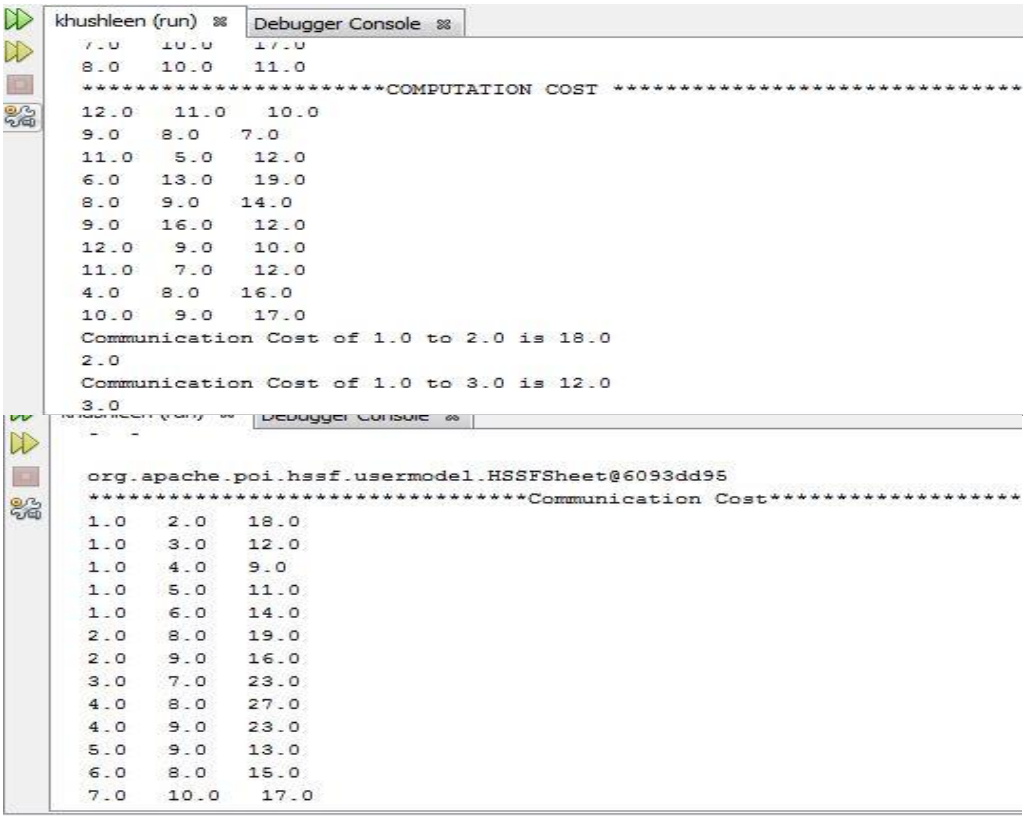


Figure 5.1 Construction of Model for establishing communication costs and computation costs.

The first of result is establishment of a connected graph in which there are V number of vertices and E number of edges. Hence a graph is drawn G (V,E). Now there are three processors for each job and to justify the significance of the proposed algorithm the processor cost has been taken as random. Consider the following table which has been drawn for 9 Nodes. The computation cost is applicable to every node at every processor. The only thing which has to be focussed is that the cost should be minimum. EST is the time of the job when it gets started on a processor and EFT is the time when it finishes off on a processor. And DAG for which these costs are assumed is in the Figure below.

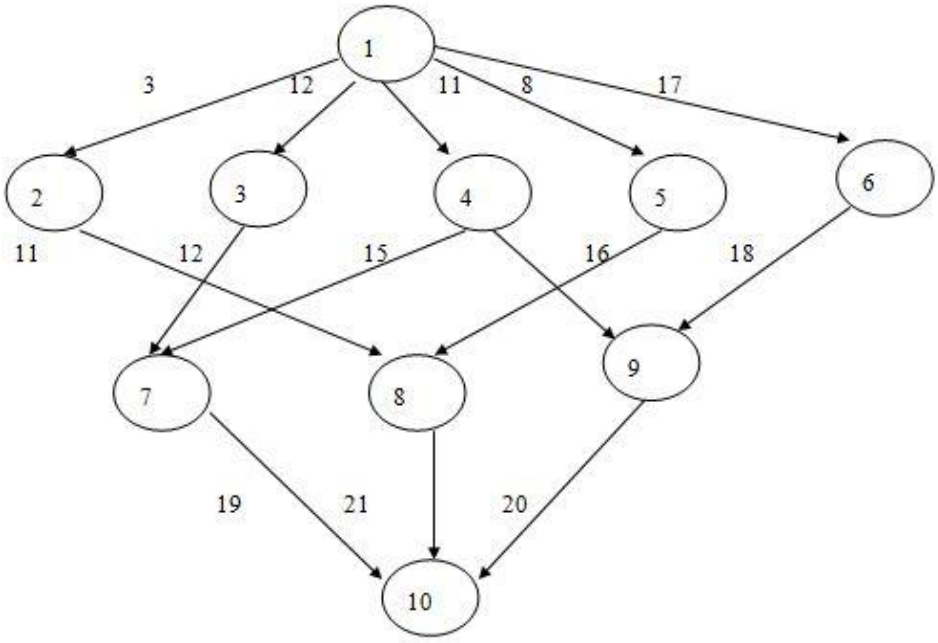


Figure 5.2 DAG with V=10 and E=14

Communication cost is applicable when the processor changes and computation cost is applicable when the processor is executing the task. If there is no Parent , there would be no communication cost and a child can never be executed before its parent.

Table 5.2 Computation cost of Node for processors

P_1	P_2	P_3
13	10	11
11	6	8
9	23	14
11	12	16
15	13	18
19	16	15
8	7	9
11	13	15
15	12	9
1	4	6

Costing in terms of milli joules have been set to signify the energy consumption concept. The table says that if NODE 1 is executed on Processor P1 at 13 mille Joules if executed on Processor 2 than it takes 10 joules . If represented graphically, the following graph would be attained. First of all the order of the jobs has to be decided. The decision making algorithm has been discussed in the methodology section. The following order is gained for one set of values.

Table 5.3 Job Order

JOB ORDER					
Level 1	1				
Level 2	2	3	4	5	6
Level 3	7	8			9
Level 4	10				

There is only one Job at level 1, 5 Jobs at level 2, 3 Jobs at level 3 and one exit Job at level 4.

The data can be represented diagrammatically as follows:

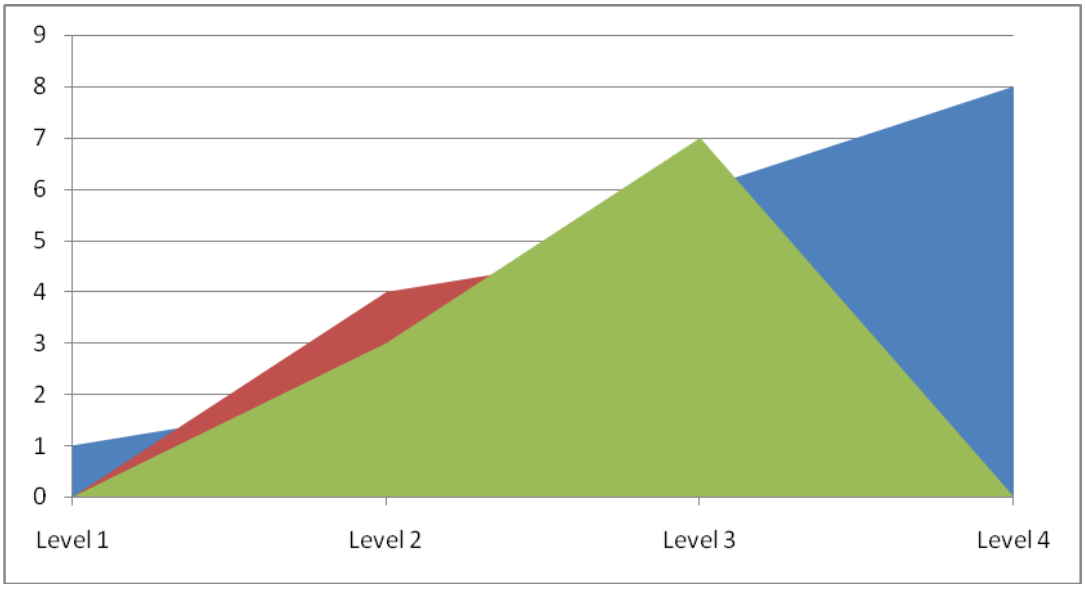


Figure 5.3 Precedence order

Now for scheduling the tasks/nodes we have two parameters Earliest Start Time(EST) and Earliest Finish time(EFT) and for calculating Earliest Finish Time we have formula.

EFT= EST + Communication Cost + Computation Cost

And as at the Entry level i.e node 1 there would be no communication cost and only the least cost of computation out of the three Processors would be chosen i.e 10 for node 1 . If a node has two or more parents than it can not be executed on the processor till it gets the information from the parent nodes. Hence the tasks would be scheduled on the processors and when we applied GA on tasks more tasks were scheduled on the Processors than the normal task scheduler within the time frame . The Energy consumption was reduced and the tasks were scheduled within the Time frame. More time taken means more energy consumption. So figures represent EFE(Earliest finish Energy Consumption) for the Energy consumed for the proposed that is optimized by GA and the the task scheduled without optimization.

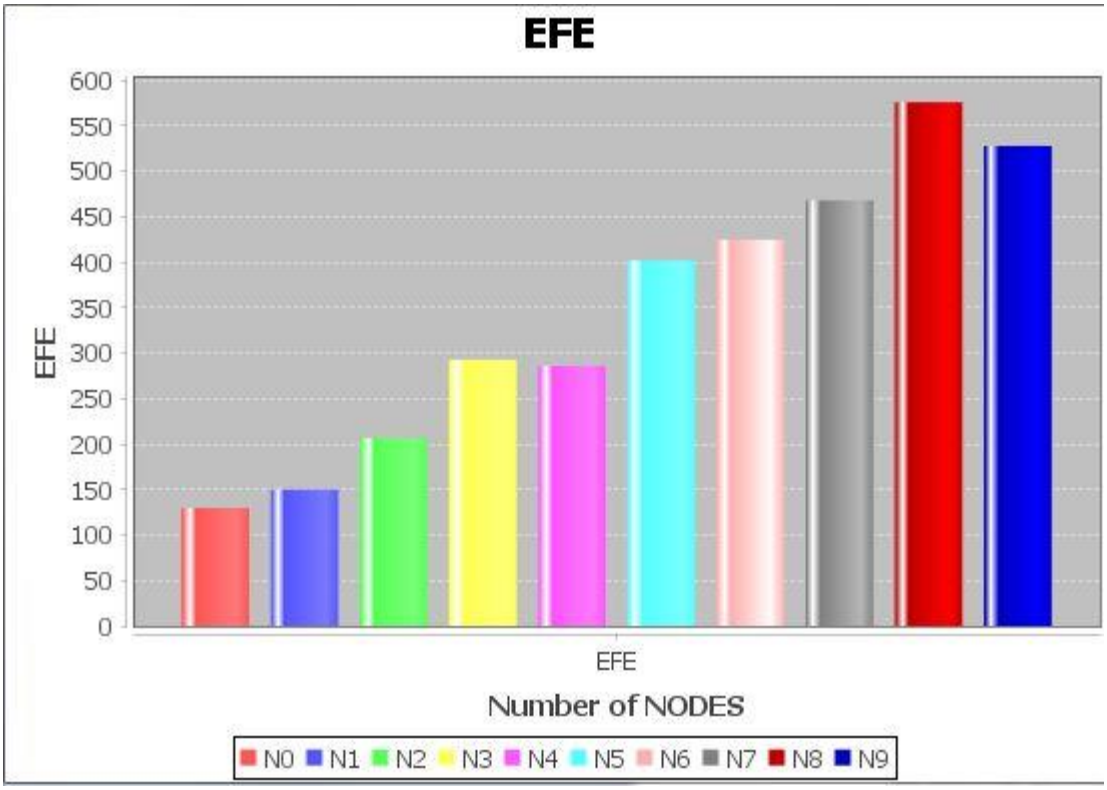


Figure 5.4 Earliest Finishing Energy Consumption DVFS inspired

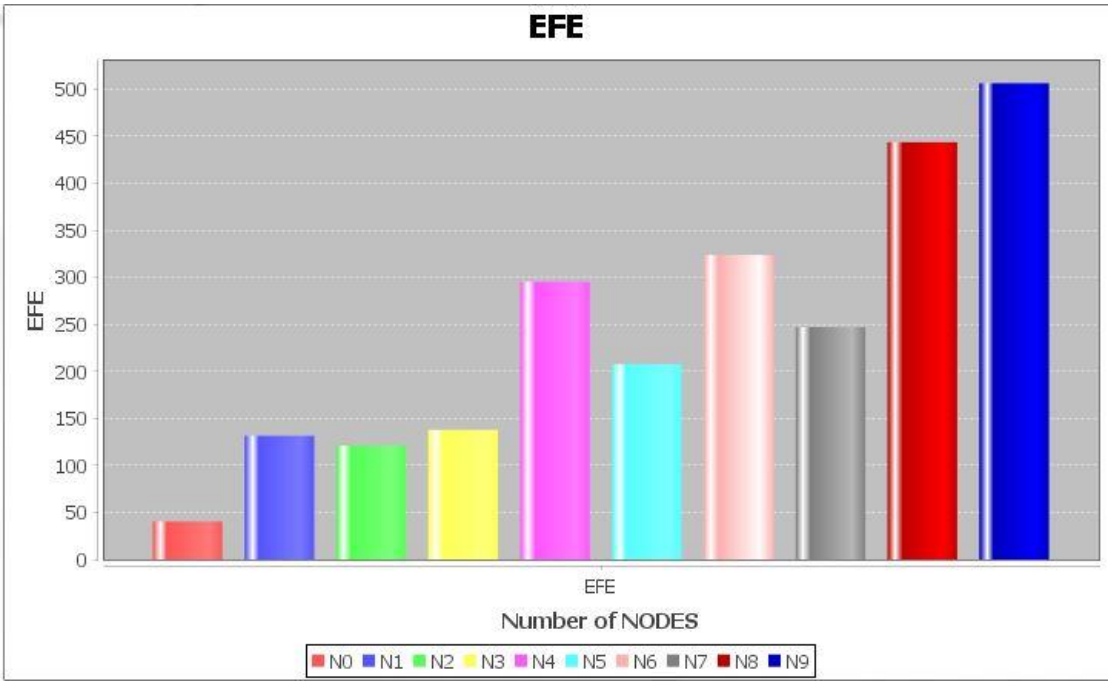


Figure 5.5 Earliest Finishing Energy Consumption GA inspired

After comparing the results of DVFS inspired and GA inspired . The results of DVFS are taking more energy for the nodes than the GA inspired. As we can see N0 in DVFS inspired takes 126.83 mj and in GA inspired N0 takes 87.937. So the results are improved by reducing the Energy consumption. Now the processor Average Energy Cost is

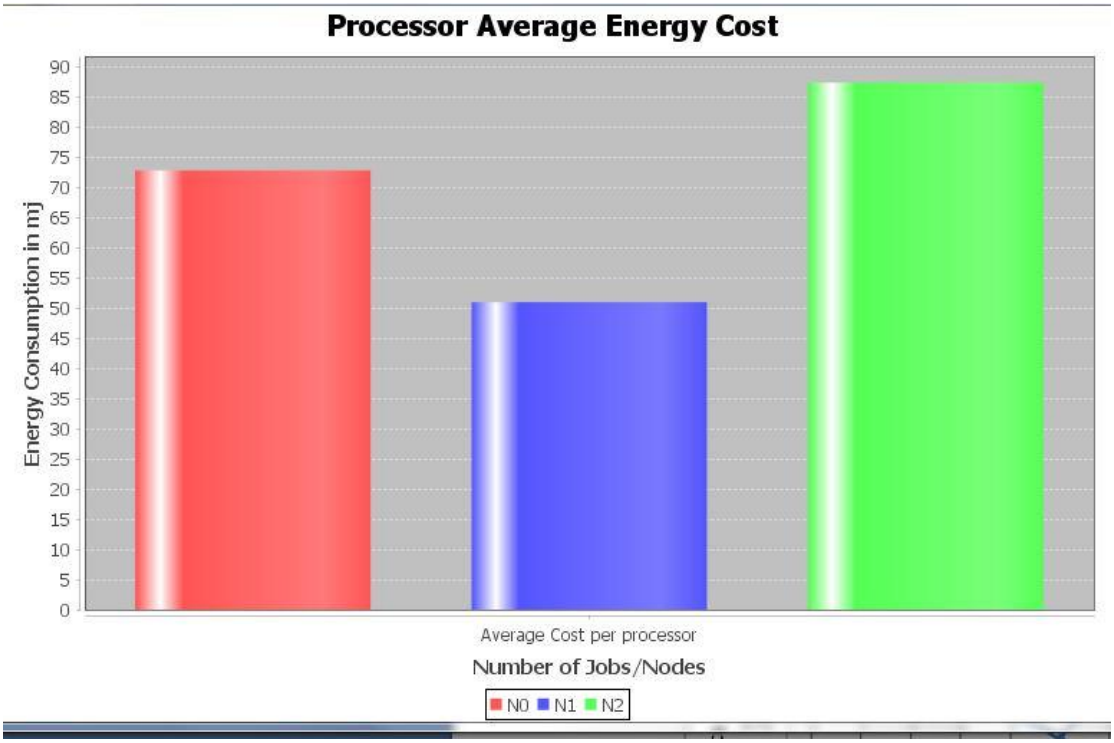


Figure 5.6 Processor Average Energy Cost in milli joules

Chapter 6: Conclusion

DAG was designed to solve multiprocessor complex problems and is quite an efficient algorithm. Even though, it is very significant, it requires some modifications into it and that what has been done in the proposed solution. The DAG overlooked the concept of dependency and in the proposed solution; the dependency diagram has been drawn. The entire framework has been designed to reduce the consumption of energy in the entire set. 10 different set of Job series has been taken to perform the proposed algorithm and it has been observed that the results are significant and almost identical. This proves that even if the situation changes, the proposed algorithm works well.

Usage of Genetic Algorithms has improved the processor selection procedure and as a reflection of the improvement the energy consumption has looked consistent in all set of jobs. The entire framework has been evaluated over 3 set of parameters namely CCR, MAKESPAN and Total Energy Consumed.

The current framework provides great opportunity for the future research workers. The researchers may opt. of other swarm intelligence techniques like Cuckoo Search, Particle Swarm would be also interesting to observe.

References

1. Z. Li; S. Tesfatsion; S. Bastani; A. A. e. Hassan; E. Elmroth; M. Kihl; R. Ranjan, "A Survey on Modeling Energy Consumption of Cloud Applications: Deconstruction, State of the Art, and Trade-off Debates," in *IEEE Transactions on Sustainable Computing* , vol.PP, no.99, pp.1-1.
2. K. Dhal, P. K. Pattnaik and S. C. Rai, "Critique of Some Fine-Grained Access Control Models in Cloud Computing Environment," *2016 International Conference on Information Technology (ICIT)*, Bhubaneswar, India, 2016, pp. 206-211.
3. Q. Lei, Y. Jiang and M. Yang, "Evaluating Open IaaS Cloud Platforms Based upon NIST Cloud Computing Reference Model," *2014 IEEE 17th International Conference on Computational Science and Engineering*, Chengdu, 2014, pp. 1909-1914.
4. R. B. Bohn, J. Messina, F. Liu, J. Tong and J. Mao, "NIST Cloud Computing Reference Architecture," *2011 IEEE World Congress on Services*, Washington, DC, 2011, pp. 594-596.
5. P. Jain, A. Datt, A. Goel and S. C. Gupta, "Cloud service orchestration based architecture of OpenStack Nova and Swift," *2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Jaipur, 2016, pp. 2453-2459.
6. W. Tan, Y. Sun, L. X. Li, G. Lu and T. Wang, "A Trust Service-Oriented Scheduling Model for Workflow Applications in Cloud Computing," in *IEEE Systems Journal*, vol. 8, no. 3, pp. 868-878, Sept. 2014.
7. S. Ahmad, S. Paul and A. P. Singh, "Tokenization based service model for cloud computing environment," *2016 International Conference on Inventive Computation Technologies (ICICT)*, Coimbatore, 2016, pp. 1-7.
8. M. U. Bokhari, Q. M. Shallal and Y. K. Tamandani, "Cloud computing service models: A comparative study," *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, New Delhi, 2016, pp. 890-895.

9. K. Xue and P. Hong, “A Dynamic Secure Group Sharing Framework in Public Cloud Computing,” in *IEEE Transactions on Cloud Computing*, vol. 2, no. 4, pp. 459-470, Oct.-Dec. 1 2014.
10. R. Soares Boaventura, K. Yamanaka, G. Prado Oliveira, E. Rodrigues Pinto and M. Braz de Andrade Maciel, “Methodology for statistical analysis comparing the algorithms performance: case of study in virtual environments in private cloud computing,” in *IEEE Latin America Transactions*, vol. 15, no. 2, pp. 333-340, Feb. 2017.
11. Z. Bouzidi, L. Terrissa, A. Lahmadi, N. Zerhouni and R. Gouriveau, “Neuro-fuzzy model for Prognostic as a Service in private cloud computing,” *2016 2nd International Conference on Cloud Computing Technologies and Applications (CloudTech)*, Marrakech, 2016, pp. 360-367.
12. X. Mao, C. Li, W. Yan and S. Du, “Optimal Scheduling Algorithm of MapReduce Tasks Based on QoS in the Hybrid Cloud,” *2016 17th International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT)*, Guangzhou, 2016, pp. 119-124.
13. Gao Xin cheng and Wang Yan, “Research of digital community service platform based on cloud computing,” *2013 8th International Conference on Computer Science & Education*, Colombo, 2013, pp. 824-828.
14. U. Idachaba and F. Wang, “A Community-Based Cloud Computing Caching Service,” *2015 IEEE International Congress on Big Data*, New York, NY, 2015, pp. 559-566.
15. A. Gajbhiye and K. M. P. Shrivastva, “Cloud computing: Need, enabling technology, architecture, advantages and challenges,” *2014 5th International Conference - Confluence The Next Generation Information Technology Summit (Confluence)*, Noida, 2014, pp. 1-7.
16. N. K. Shah, “Big data and cloud computing: Pitfalls and advantages in data management,” *2015 2nd International Conference on Computing for Sustainable Global Development (INDIACom)*, New Delhi, 2015, pp. 643-648.
17. R. G. Alakbarov, F. H. Pashayev and O. R. Alakbarov, “Optimal deployment model of cloudlets in mobile Cloud Computing,” *2017 IEEE 2nd International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)*, Chengdu, China, 2017, pp. 213-217.

18. H. Allam, N. Nassiri, A. Rajan and J. Ahmad, “A critical overview of latest challenges and solutions of Mobile Cloud Computing,” *2017 Second International Conference on Fog and Mobile Edge Computing (FMEC)*, Valencia, Spain, 2017, pp. 225-229.
19. C. R. Panigrahi, J. L. Sarkar, B. Pati and S. Bakshi, “E^3M: An Energy Efficient Emergency Management System using mobile cloud computing,” *2016 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, Bangalore, India, 2016, pp. 1-6.
20. Dongqing Liu, L. Khoukhi and A. Hafid, “Decentralized data offloading for mobile cloud computing based on game theory,” *2017 Second International Conference on Fog and Mobile Edge Computing (FMEC)*, Valencia, Spain, 2017, pp. 20-24.
21. A. H. Gamlo, N. Zhang and O. Bamasag, “Mobile Cloud Computing: Security Analysis,” *2017 5th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud)*, San Francisco, CA, USA, 2017, pp. 191-198.
22. H. Zheng; Y. Feng; J. Tan, “A Hybrid Energy-aware Resource Allocation Approach in Cloud Manufacturing Environment,” in *IEEE Access* , vol.PP, no.99, pp.1-1
23. G. Lee, H. Ko and S. Pack, “An Efficient Delta Synchronization Algorithm for Mobile Cloud Storage Applications,” in *IEEE Transactions on Services Computing*, vol. 10, no. 3, pp. 341-351, May-June 1 2017.
24. L. Ismail and A. A. Fardoun, “Towards energy-aware task scheduling (EATS) framework for divisible-load applications in cloud computing infrastructure,” *2017 Annual IEEE International Systems Conference (SysCon)*, Montreal, QC, 2017, pp. 1-6.
25. K. Zhang *et al*., “Energy-Efficient Offloading for Mobile Edge Computing in 5G Heterogeneous Networks,” in *IEEE Access*, vol. 4, no. , pp. 5896-5907, 2016.
26. Liang Luo, Wenjun Wu, Dichen Di, Fei Zhang, Yizhou Yan and Yaokuan Mao, “A resource scheduling algorithm of cloud computing based on energy efficient optimization methods,” *2012 International Green Computing Conference (IGCC)*, San Jose, CA, 2012, pp. 1-6.

27. Bei Wang, Jun Li and Chao Wang, "Cost-effective scheduling precedence constrained tasks in cloud computing," *2017 IEEE 2nd International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)*, Chengdu, China, 2017, pp. 230-235.
28. M. Kaur and M. Agnihotri, "Performance evaluation of hybrid GAACO for task scheduling in cloud computing," *2016 2nd International Conference on Contemporary Computing and Informatics (IC3I)*, Noida, 2016, pp. 168-172.
29. Y. Wen; Z. Li; S. Jin; C. Lin; Z. liu, "Energy-Efficient Virtual Resource Dynamic Integration Method in Cloud Computing," in *IEEE Access* , vol.PP, no.99, pp.1-1
30. X. Lin, Y. Wang, Q. Xie and M. Pedram, "Task Scheduling with Dynamic Voltage and Frequency Scaling for Energy Minimization in the Mobile Cloud Computing Environment," in *IEEE Transactions on Services Computing*, vol. 8, no. 2, pp. 175-186, March-April 1 2015.
31. E. Aldahari, "Dynamic Voltage and Frequency Scaling Enhanced Task Scheduling Technologies toward Green Cloud Computing," *2016 4th Intl Conf on Applied Computing and Information Technology/3rd Intl Conf on Computational Science/Intelligence and Applied Informatics/1st Intl Conf on Big Data, Cloud Computing, Data Science & Engineering (ACIT-CSII-BCD)*, Las Vegas, NV, 2016, pp. 20-25.
32. F. Cao and M. M. Zhu, "Energy Efficient Workflow Job Scheduling for Green Cloud," *2013 IEEE International Symposium on Parallel & Distributed Processing, Workshops and Phd Forum*, Cambridge, MA, 2013, pp. 2218-2221.
33. K. Gai; M. Qiu; H. Zhao, "Cost-Aware Multimedia Data Allocation for Heterogeneous Memory Using Genetic Algorithm in Cloud Computing," in *IEEE Transactions on Cloud Computing* , vol.PP, no.99, pp.1-1.
34. B. Wang and J. Li, "Load balancing task scheduling based on Multi-Population Genetic Algorithm in cloud computing," *2016 35th Chinese Control Conference (CCC)*, Chengdu, 2016, pp. 5261-5266.
35. M. S. Pilavare and A. Desai, "A novel approach towards improving performance of load balancing using Genetic Algorithm in cloud computing," *2015 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)*, Coimbatore, 2015, pp. 1-4.

36. T. Wang, Z. Liu, Y. Chen, Y. Xu and X. Dai, "Load Balancing Task Scheduling Based on Genetic Algorithm in Cloud Computing," *2014 IEEE 12th International Conference on Dependable, Autonomic and Secure Computing*, Dalian, 2014, pp. 146-152.
37. F. Zhao, G. Nian, H. Jin, L. T. Yang and Y. Zhu, "A Hybrid eBusiness Software Metrics Framework for Decision Making in Cloud Computing Environment," in *IEEE Systems Journal*, vol. 11, no. 2, pp. 1049-1059, June 2017.
38. T. Lin, T. Alpcan and K. Hinton, "A Game-Theoretic Analysis of Energy Efficiency and Performance for Cloud Computing in Communication Networks," in *IEEE Systems Journal*, vol. 11, no. 2, pp. 649-660, June 2017.
39. X. Wang, J. Wang, X. Wang and X. Chen, "Energy and Delay Tradeoff for Application Offloading in Mobile Cloud Computing," in *IEEE Systems Journal*, vol. 11, no. 2, pp. 858-867, June 2017.
40. F. Azimzadeh and F. Biabani, "Multi-objective job scheduling algorithm in cloud computing based on reliability and time," *2017 3th International Conference on Web Research (ICWR)*, Tehran, Iran, 2017, pp. 96-101.
41. R. Singh and M. Agnihotri, "Proactive framework for energy efficient job scheduling in cloud computing," *2016 2nd International Conference on Contemporary Computing and Informatics (IC3I)*, Noida, 2016, pp. 711-717.
42. K. S. Bahwaireth, L. Tawalbeh, A. Basalamah, Y. Jararweh and M. Tawalbeh, "Efficient techniques for energy optimization in Mobile Cloud Computing," *2015 IEEE/ACS 12th International Conference of Computer Systems and Applications (AICCSA)*, Marrakech, 2015, pp. 1-8.
43. Liang Luo, Wenjun Wu, Dichen Di, Fei Zhang, Yizhou Yan and Yaokuan Mao, "A resource scheduling algorithm of cloud computing based on energy efficient optimization methods," *2012 International Green Computing Conference (IGCC)*, San Jose, CA, 2012, pp. 1-6.
44. P. Arroba, J. M. Moya, J. L. Ayala and R. Buyya, "DVFS-Aware Consolidation for Energy-Efficient Clouds," *2015 International Conference on Parallel Architecture and Compilation (PACT)*, San Francisco, CA, 2015, pp. 494-495.
45. B. Moons, R. Uytterhoeven, W. Dehaene and M. Verhelst, "DVAFS: Trading computational accuracy for energy through dynamic-voltage-accuracy-

frequency-scaling," *Design, Automation & Test in Europe Conference & Exhibition (DATE), 2017*, Lausanne, 2017, pp. 488-493.

46. Abdul Razaque, Nikhileshwara Reddy Vennapusa, "Task Scheduling in Cloud Computing",IEEE2016.
47. Dr. Amit Agarwal, "Efficient Optimal Algorithm of Task Scheduling in Cloud Computing Environment", *International Journal of Computer Trends and Technology (IJCTT)* – volume 9 number 7– Mar 2014.
48. Raja Manish Singh, "Task Scheduling in Cloud Computing: Review," *International Journal of Computer Science and Information Technologies,* Vol. 5 (6) , 2014, 7940-7944.
49. R. Durga Lakshmi, "A dynamic approach to task scheduling in cloud computing using genetic algorithm", *Journal of Theoretical and Applied Information Technology* 20th March 2016. Vol.85. No.2.
50. B. Saovapakhiran, G. Michailidis and M. Devetsikiotis, "Aggregated-DAG Scheduling for Job Flow Maximization in Heterogeneous Cloud Computing," *2011 IEEE Global Telecommunications Conference - GLOBECOM 2011*, Houston, TX, USA, 2011, pp. 1-6.
51. Y. Wang, C. Jia and Y. Xu, "Multiple DAGs Dynamic Workflow Scheduling Based on the Primary Backup Algorithm in Cloud Computing System," *2014 Ninth International Conference on Broadband and Wireless Computing, Communication and Applications*, Guangdong, 2014, pp. 177-182.
52. D. Kliazovich, J. E. Pecero, A. Tchernykh, P. Bouvry, S. U. Khan and A. Y. Zomaya, "CA-DAG: Communication-Aware Directed Acyclic Graphs for Modeling Cloud Computing Applications," *2013 IEEE Sixth International Conference on Cloud Computing*, Santa Clara, CA, 2013, pp. 277-284.
53. Haijun Cao, Hai Jin, Xiaoxin Wu, Song Wu and Xuanhua Shi, "DAGMap: Efficient scheduling for DAG grid workflow job," *2008 9th IEEE/ACM International Conference on Grid Computing*, Tsukuba, 2008, pp. 17-24.
54. Y. Liu, H. Shao, W. Jing and Z. Qiu, "Multi-DAGs Scheduling Integrating with Security and Availability in Cloud Environment," in *Chinese Journal of Electronics*, vol. 24, no. 4, pp. 709-716, 10 2015.
55. H. Arabnejad and J. G. Barbosa, "List Scheduling Algorithm for Heterogeneous Systems by an Optimistic Cost Table," in *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 3, pp. 682-694, March 2014.

56. T. Hagras and J. Janecek, “A Simple Scheduling Heuristic for ˇ Heterogeneous Computing Environments”, Proceedings Second International Symposium on Parallel and Distributed Computing, pp. 104-110, 2003.
57. A. K. Amoura, E. Bampis and J. C. Konig, “Scheduling algorithms for parallel Gaussian elimination with communication costs,” in *IEEE Transactions on Parallel and Distributed Systems*, vol. 9, no. 7, pp. 679-686, Jul 1998.
58. L.F. Bittencourt, R. Sakellariou and E.R.M. Madeira, “DAG Scheduling Using a Lookahead Variant of the Heterogeneous Earliest Finish Time Algorithm”, *18th Euromicro International Conference on Parallel, Distributed and Network-Based Processing* (PDP’10), pp. 27-34, 2010.
59. C. Boeres, J.V. Filho and V.E.F. Rebello, “A Cluster-based Strategy for Scheduling Task on Heterogeneous Processors” *16th Symposium on Computer Architecture and High Performance Computing*, pp. 214-221, 2004.
60. Sinnen O, Sousa L, “List scheduling: extension for contention awareness and evaluation of node priorities for heterogeneous cluster architectures”, Parallel Computing, Vol. 30, Issue 1, pp.81-101, 2004.

Publications

Khushleen, Ravinder Kumar "An Analytical review of task scheduling on mobile cloud computing " **accepted** at International Journal of Advanced Computronics and Management Studies (IJACMS) ISSN – 2456-1835

Khushleen, Ravinder Kumar " Energy-aware task scheduling based on the DAG applications using Genetic Algorithm in Cloud computing" **accepted** at International Journal of Advanced Computronics and Management Studies (IJACMS) ISSN – 2456-1835

Video Presentation Link

https://youtu.be/vEp0GPkvKIk