

Bio Inspired Algorithms: An Efficient Approach for Resource Scheduling in Cloud Computing

Gurtej Singh,
Department of CSE,
SGGS World University,
Fatehgarh Sahib, Punjab, India.

Amritpal Kaur
Department of CSE,
SGGS World University,
Fatehgarh Sahib, Punjab, India.

ABSTRACT

Nature plays a vital role in solving complicated problems in computer science. It helps us in finding the optimal desired way to solve extremely dynamic, difficult and robust problems. Bio inspired algorithm help us to cope with the technological need of a new era. Many researchers did enormous work in this area from the past few decades. However, still there is a large more scope for bio inspired algorithm (BIA) in exploring new application and opportunities in cloud computing. This paper presents a broad, detailed in of some Bio inspired algorithm, which was used in order to tackle various challenges faced in Cloud Computing Resource management environment.

General Terms

Cloud Computing, Scheduling Algorithm, Virtual Machine.

Keywords

Genetic Algorithm (GA), Genetic Programming (GP), Ant Colony (AC), Firefly (FF), Flower Pollination (FP), Cuckoo Search (CS), Honey Bee (HB).

1. INTRODUCTION TO BIO INSPIRED ALGORITHM

Optimization plays a very important role in solving complex mathematical problem of engineering. In a general sense, it means finding the best solution for a given problem. The field of optimization algorithms [17] studies models derived from the observation and these models as a source of inspiration for the design of novel algorithms for the solution of optimization and distributed control problems. Traditional methods require an enormous amount of computation efforts which tend to fail as the problem size increases. Using Bio inspired algorithm is a motivational approach for employing computationally efficient alternative models for deterministic approach. Meta-heuristics approaches are based on the iterative improvement of either a population of solutions (for example, in Evolutionary algorithms, Swarm based algorithms) and mostly used randomization and local search to solve a given optimization problem.

The organization of the paper is as follows: Section 2 provides a classification tree of Bio inspired scheduling algorithm. Section 3 provides an overview of Swarm intelligence scheduling algorithm. Section 4 provides an overview of Evolutionary scheduling algorithms. Section 5 provides the conclusion and future scope of work.

2. CLASSIFICATION TREE OF BIO INSPIRED SCHEDULING ALGORITHM

The division of bio inspired scheduling algorithm in two categories is broadly elaborated in Fig 1. We can divide the

scheduling algorithm of cloud computing in two categories which are Evolutionary scheduling algorithm and swarm intelligence based scheduling algorithm. Traditional scheduling algorithms [1] are used in late 1960 in order to solve resource and process management in operating systems. With the passage of time these traditional scheduling was used in cloud computing as well, but somehow they were unable to match the growing needs of cloud computing resource management. Moreover, this gives birth to bio inspired algorithm. There are two types of bio inspired algorithm these are evolutionary scheduling and swarm intelligence based scheduling algorithm.

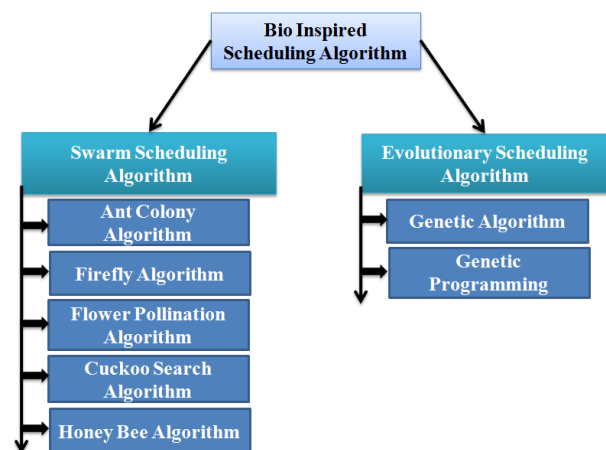


Fig 1: Hierarchy of different Scheduling Algorithm

Evolutionary algorithms provide a framework in cloud computing [2] for effectively sampling large search spaces and the basic technique is both broadly applicable and easily used to a specific cloud computing problem. Evolutionary Algorithm is computer programs that attempt to solve complex problems by mimicking the processes in a cloud computing environment by using Darwinian evolution theory. There are many well known evolutionary algorithms which are Genetic algorithm (GA), Genetic programming (GP) and Bee genetic (BG) algorithm. In many difficult cloud computing problems either, evolutionary algorithms [4] are unable to solve or required help from swarm intelligence algorithms in order to solve these complex problems efficiently. Swarm intelligence (SI) has received great interest and attention in cloud computing different branched like resource management, job scheduling, fault tolerant, etc. Swarm based algorithms are inspired by the behavior of some social living beings, such as ants, birds, and fishes. In recent years, new swarm intelligence algorithms have appeared, inspired by fish schools, gravity and mass interactions, as well as different aspects of the behavior of bees, bacteria, glowworms, fireflies, cockroaches, bats, and cuckoo birds.

In the next section, we are going to discuss one by one each Swarm intelligence scheduling algorithm used in a cloud computing environment in detail.

3. SWARM INTELLIGENCE SCHEDULING ALGORITHM

Swarm intelligence (SI) has received great interest and attention in the communities of optimization, computational intelligence, and computer science, bio-inspired algorithms especially those Swarm intelligence based algorithms, has become very popular in cloud computing environment. In fact, these nature-inspired Meta heuristic algorithms are now among the most widely used algorithms for optimization and computational intelligence in cloud computing systems.

Swarm intelligence [6] based algorithms are inspired by the behavior of some social living beings, such as ants, termites, birds, and fishes. In recent years, new swarm intelligence algorithms have appeared, inspired by fish schools, gravity and mass interactions, as well as different aspects of the behavior of bees, bacteria, glow-worms, fireflies, cockroaches, bats and cuckoo birds. Despite the swarm inspiration common to these approaches, they have their own particular way to exploit and explore the search space of the problem.

In Swarm algorithm our main concentration is on the implementation of these algorithms in order to solve complex cloud computing problems. Some most well known swarm based algorithms used in cloud computing system are discussed below in detail. These algorithms are being inspired by the collective behavior of animals, exhibit decentralized, self-organized patterns in the foraging process.

3.1 Ant Colony Algorithm

Ant colony algorithm [10] model had been derived from the observation of real ant's behavior, and this model act as a source of inspiration in design of novel algorithms in order to find the solution of optimization and distributed control problems in cloud computing environment. The main idea behind the ant colony algorithm is self-organizing principles which allow the highly coordinated behavior of real ants and which can be exploited to coordinate populations of artificial agents and it help us in solving computational problems of cloud computing environment.

An important finding of the researcher about ant behavior is that most of the communication among ants and the environment is based on the use of chemical produced by the ants. This chemical is known as pheromones. Ants with the help of their smelling power are going to smell the pheromone choose that path which has high pheromone concentration

In the Fig 2(a) the bridge had two branches of equal length. At the start, ants (numbered by 1, 2, and 3) were left free to move between the starting point and the food source and the selection of one path out of two is chosen randomly. Which we can see clearly in Fig 2(b). While moving through those two path ants had deposited the pheromones the percentage of ant's pheromones directly depends on the number of ants moved over that path. Furthermore, all the three ants went away with the fraction of food (represented by OD in Fig 2(c)). Ants has come back to the initial state this time ant's had chosen the path which had high pheromone density as this process can be seen clearly in fig 2(d) and fig 2(e). At end, the ants had moved away with the leftover food as well as also increase the pheromone density from the path which they had chosen. This whole algorithm can help us in solving the cloud

computing resource management and job scheduling problem. This approach can be easily understood by analyzing the below listed simple ant colony algorithm.



Fig 2(a) : Shows initial path chosen by ants randomly in to reach the food source

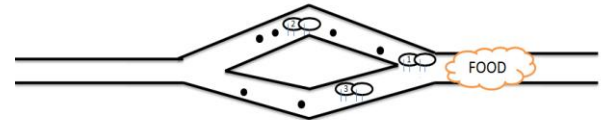


Fig 2(b) : Ants had chosen different path depending on their interest

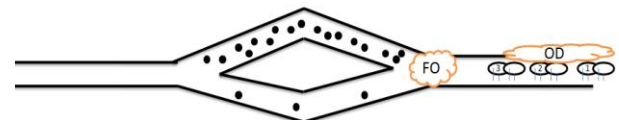


Fig 2(c) : increase in the pheromones density at the path which is chosen by ants

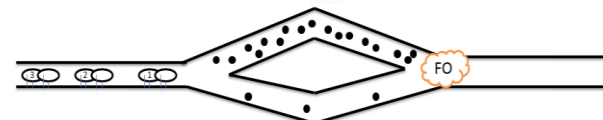


Fig 2(d) : Ants have come again in order to take away the left food

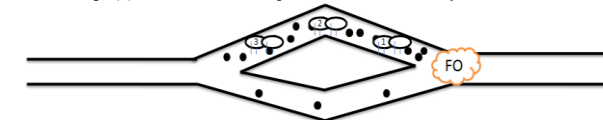


Fig 2(e) : All the ant's had chosen the path which has high density of pheromones

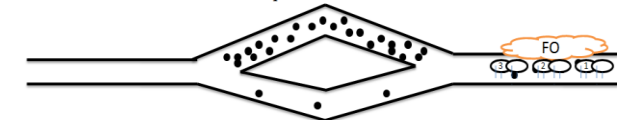


Fig 2(f) : All the ant's at end move away by taking away the left food.

	Ant representation (1 represent serial number of ant)
	Pheromone Representation
	Food Source (Act As Destination)

3.1.1 Simple Ant Colony Algorithm

Step1) Job arrived on the cloud server is inserted in the job queue. Pheromone count for all the virtual machine available to cloud server is equated to zero.

Step2) Jobs are moved to different resource pool queue RP[j] depending upon the characteristics mapping between the resource pool queue RP[j] and job selected by the scheduler from the job queue.

Step3) For every single resource pool queue RP[j] a single ant AN[j] is allocated.

Step4) An ant AN[j] select the job from the resource pool queue RP[j] and allocate the virtual machine VM[i] (where $i < j$) depending on the characteristics mapping between the job and the virtual machine and also depending on the pheromone concentration on the virtual machine.

Step5) Job starts executing on the selected virtual machine.

Step6) A pheromone is deposited at the virtual machine with

the passage of time the amount of pheromone concentration start increasing with the help of a certain formula depending on the no jobs arrived at a desired virtual machine.

Step7) The concentration of pheromone also decreased on the virtual machine with the help of same formula which we used in order to increase the pheromone concentration.

Step8) Go to step 4 until the last resource pool queue is emptied.

3.2 Firefly Algorithms

The fire flies [5] are a amazing natural specie which coagulates and produces immense flash light. There are numerous firefly species and each firefly species has a unique pattern of flashes. The flashing light is produced by a process of bio luminescence.

There are generally two functions of such flashes firstly in order to attract the mating partner and to attract to potential prey. There are numerous factors which bring both the sex together those are rate of flashing, rhythmic flash and amount of time between flashes. The response of the female depends on the unique pattern generated by male specie. This natural phenomenon can help us in solving a large amount of complex cloud computing problem in scheduling and managing the resources. Depending on firefly [8] behavior below listed simple algorithm shows how this natural phenomenon can help us in managing resources in cloud computing environment.

3.2.1 Simple Fire Flies Algorithm

Step1) Job arrived on the cloud server is inserted in the job queue. Define the light absorption coefficient for all the virtual machine available to cloud server and initialize it to zero.

Step2) Jobs are moved to different resource pool queue $RP[j]$ depending upon the characteristics mapping between the resource pool queue $RP[j]$ and job selected by the scheduler from the job queue.

Step3) for every single resource pool queue $RP[j]$ a single firefly $FF[j]$ is allocated.

Step4) A firefly $FF[j]$ select the job from the resource pool queue $RP[j]$ and allocate the virtual machine $VM[i]$ (where $i < j$) depending on the characteristics mapping between the job and the virtual machine and also depending on the light absorption coefficient value on the virtual machine.

Step5) Job starts executing on the selected virtual machine.

Step6) Light absorption coefficients value at the virtual machine start increasing with the passage of time depending on the number of jobs arrived at that virtual machine.

Step7) Light absorption coefficients value of firefly decreased on the virtual machine with the passage of time depending on the number of jobs completed their execution at that virtual machine.

Step8) Go to step 4 until the last resource pool queue is emptied.

3.3 Flower Pollination Algorithm

Flower pollination is a process which acts inside almost 80 percent of the plant species on the earth. Flower pollination [9] is a process in which transfer of pollen grain plays an important role and such transfer is taken care by pollinators such as insects, birds, bats, other animals. Some flowers can only attract a specific species of insect or bird for a successful

pollination process there are only two major forms of the pollination process biotic and a-biotic pollination. In biotic pollination process, pollen grains are transferred by pollinator's insects and animals. In a-biotic pollination process which does not require pollinator. In addition to it find help in pollination of such plants flower (for e.g. Grass). This flower pollination [11] phenomenon can use in order to solve many complex computational and distributed problems of cloud computing environment. In addition to it also help us in solving scheduling, optimization and resource management problem of cloud computing system. The below listed, figure shows how the whole pollination process happens in the natural environment.

In fig 3(a) which clearly shows the different parts of the plant for which a plant is composed of at the top of the plant we can see the pollen grain. In addition to it, this is the part of the plant where the pollinator is going to sit and take pollens along with him and these pollens act as an important part in the pollination process.

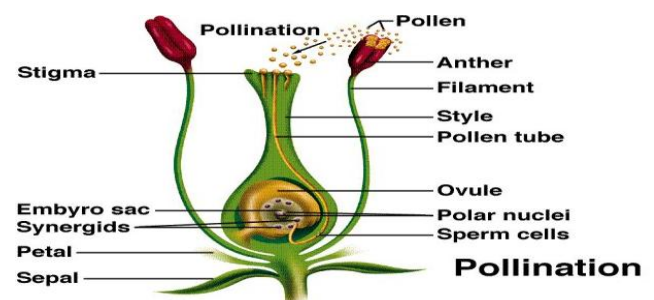


Fig 3 (a): Pollen grains of plants [15]

This natural phenomenon can help us in solving a large amount of complex cloud computing problem in scheduling and managing the resources. Depending on the flower pollinator behavior a below listed simple algorithm shows how this natural phenomenon can help us in managing resources in a cloud computing environment.

3.3.1 Simple Flower Pollination Algorithm

Step1) all submitted job reside in the job queue of cloud server. Calculate the queue depth or length and assign its value to variable called queue_length (QL).

Step2) Generate the initial bee required by applying a random function on the queue_length (QL) and assign its value to variable bee_required(BR).

Step3) for each bee generated extracts job from the job queue of cloud server and assign the job to the virtual machine. Start executing job.

Step4) Go to step1 until the job queue is emptied.

3.4 Cuckoo Search (CS)

The Cuckoo search [3] was inspired from a natural phenomenon in which there is cuckoo specie which lay eggs in the nest of host birds. In addition to it there are some female cuckoos which change the color and patterns of the host birds. This mechanism helps in bluffing the host bird of the cuckoo bird. But sometimes host birds the eggs are not their own and they will throw their whole nest and start building a new one at a different place. The cuckoo eggs hatch a little earlier than the host eggs. Once the first cuckoo chick is hatched his first instinct action is to evict the host eggs by blindly. Propelling the eggs out of the nest. This action results in increasing the cuckoo chick's share of food provided by its host bird.

This natural phenomenon can be used in order to solve a large number of complex cloud computing problems in scheduling and managing the resources. Depending on the cuckoo search [13] the below listed simple algorithm shows how this natural phenomenon can help us in managing resources in cloud computing environment.

3.4.1 Simple Cuckoo Search Algorithm

Step1) Job arrived on the cloud server is inserted in the job queue.

Step2) Jobs are moved to different resource pool queue RP[j] depending upon the characteristics mapping between the resource pool queue RP[j] and job selected by the cuckoo scheduler from the job queue.

Step3) for every single resource pool queue RP[j] a single host bird HB[j] is allocated.

Step4) host birds HB[j] select the job from the resource pool queue RP[j] and allocate the virtual machine VM[i] (where $i < j$) depending on the characteristics mapping between the job and the virtual machine.

Step5) Job starts executing on the selected virtual machine.

Step6) Go to step 4 until the last resource pool queue is emptied.

3.5 Honey Bee Algorithm

Honey bee [7] travel over long distances and simultaneously in multiple directions in search of food sources. Flower with large amount of pollen receives a large amount of honey bees in comparison with the flower which has a less amount of pollen. Scout bees begin their search in order to find for promising flower patches. Scout bees move randomly from one patch to another. When scout bee return to the hive and those scout bees which had found a patch greater than a threshold deposit their pollen and a waggle dance is performed on the dance floor by a bee. While extracting the food from the hive bee monitors the food level of the patch. This is very crucial thing as the next waggle dances performed by the bee depend on the current amount food available in the hive to a large extent. Furthermore, if the patch is still rich in food source than a waggle dance is performed by the bee on the dance floor and more bees will be recruited to that source. Scout bee main purpose is to discover a new food source which employed bees were unable to find.

In cloud computing, we can usually implement this algorithm while dealing with job fault tolerant. Apart from this algorithm also help us in dealing with difficult cloud computing problems. These bees perform their operation by following the random search approach. There are different types of dance performed by Scout Bee which are listed below:

3.5.1 Waggle Dance

Scout bees to communicate information about food source to the rest of the colony perform this dance on the dance floor. Scout bees provide the following information by means of waggle dance .

- Quality of food source.
- Distance of the food source from the hive.
- Direction of the food source.

3.5.2 Tremble Dance

Scout bee performs this dance from ages in order to recruit more honey bees to collect nectar from the food source. This dance is performed on the dance floor in order to communicate a long delay in unloading the nectar or shortage of receiver bees.

3.5.3 Simple Honey Bee Algorithm

Step1) all submitted job resides in the job queue of cloud server. Calculate the queue depth or length and assign its value to variable called queue_length (QL).

Step2) If the queue_length is greater than zero than bee is going to perform a waggle dance at the job_queue_hive.

Step3) Generate the initial bee required by applying a random function on the queue_length (QL) and assign its value to variable bee_required(BR).

Step3) for each bee generated extracts job from the job queue of cloud server and assign the job to the virtual machines. Start executing job.

Step4) Go to step1 until the job queue is emptied.

In this section, we had discussed the Swarm scheduling algorithm used in cloud computing environment in detailed. Evolutionary scheduling algorithms were unable to meet the needs of cloud computing complex problems. Evolutionary algorithm seeks help from swarm intelligence algorithms in order to solve these complex problems efficiently. In next section we are going to discuss evolutionary scheduling algorithm one by one.

4. EVOLUTIONARY SCHEDULING ALGORITHM

Evolutionary Algorithm [12] is a term which can be defined with the help of natural mechanisms of selection and evolution. This term is mainly divided into three sub categories evolutionary strategies, genetic algorithm and genetic programming.

4.1 Genetic algorithm

In genetic algorithm [18], mainly we use the four terms which are selection, crossover population and mutation. In this algorithm, we use a long binary string instead of using advanced data structures like trees or graphs in order to implement this algorithm in cloud computing. In genetic algorithm, we used Charles Darwin's theory ("survival of the fittest") of biological evolution in natural science. There are three operators [16] used in the genetic algorithm these are

- Crossover
- Mutation
- Selection of the fittest.

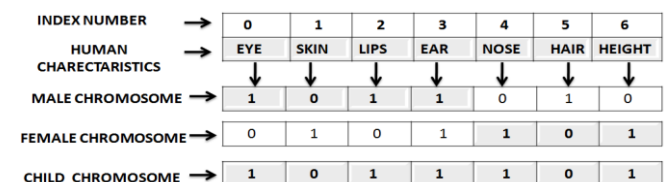


Fig 4(a): Genetic Algorithm Operation

There are three main genetic operators used in order to implement this algorithm in cloud computing system which is crossover, mutation, and selection.

- a) **Crossover:** It is a process in which we are going to find an index location among chromosomes long binary strings from that location we are going to perform the next operation called mutation in cloud computing environment. For example, in Fig 4(a) we can see that the male chromosome index number are 3 (0 to 3) and for a female chromosome index number are 4 (4 to 6). Both of these two index show a virtual boundary in two chromosomes from the crossover operation is going to be executed.
- b) **Mutation:** It is processes in which we randomly select the parts of one chromosome (male or female) this help us in increasing the diversity of the population. For example in fig 4(a) Male chromosome dominate female chromosomes from characteristics index 0 to 3 and female chromosomes dominate male chromosomes from characteristics index 4 to 6 as a result of the mutation operation is performed on this dominating characteristics this give rise to child's chromosomes.
- c) **Selection of the fittest:** It is a process in which selects the chromosomes with high fitness factor and we pass these chromosomes to the next generation. This whole process will help us in reaching the best chromosomes in the near future. For example in fig 4(a) we can say that the child chromosome is better in each and every sense than male or female chromosomes.

In this algorithm character strings are represented by an array of bits in order to represent the chromosomes. On this set of chromosomes we perform operations like crossover, mutation and selection of the fittest. Our aim in this is to reach a chromosome, which is optimal. In genetic algorithm [12], we select or create an initial random population of size 'X' and every single element of this population has a fixed size of 'L' length. we select the first individual string (FIS) from the given population set. In this phase only we set an optimal fitness value which is given as an input to this program.

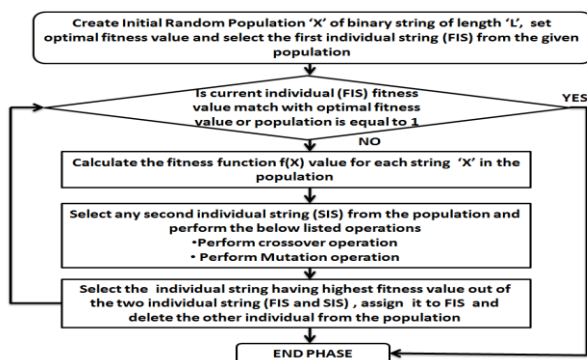


Fig 4(b): Genetic Algorithm Flow chart

We are going to calculate the fitness function $F(X)$ value for each string 'X' of a given population and then we select second individual string (SIS) out of the whole population and perform the crossover and mutation operation on these two binary string's. We are going to calculate the fitness factor value of the two strings after performing crossover and mutation operation and then the individual string that has the higher fitness factor value will be assigned its value to FIS, the other individual string will be deleted from the population set. Next iteration is performed which contain above listed

same operation is going to be performed if two conditions are satisfied which are current individual (FIS) fitness value matched with an optimal fitness value fixed and entered by the user initially or population size goes down equal to one.

4.1.1 Simple Genetic Algorithm

The below listed algorithm gives a clear way to solve the genetic algorithm problem, a simple GA works as follows:

- Step1) In the initial state we start with a random population of n binary strings of length "L".
- Step2) Calculate the value of the fitness function $F(x)$ of each chromosome string x in the population.
- Step3) Cross over the two parent's chromosome strings with probability P_c . Select one of the chromosome string at random and discard the other.
- Step4) Mutate each bit in the selected chromosome strings with the probability P_m and place it in the new population.
- Step5) Go to step2 until a new population is complete.
- Step6) Go to Step1.

4.2 Genetic Programming

It is designed to store computer programs in such a way that it can be optimized using an evolutionary approach. In evolutionary algorithm various operation likes crossover, mutation and calculating the fitness value of individual. Computer programming languages like LISP was used to code computer program into an individual. LISP code was also translated into program trees. These program tresses were easier to manipulate and they can be implemented in any computer programming language. Nodes act as an important building block for program tree and these nodes can point to other nodes. If a node does not point to any node then that node is known as a leaf node. For example, a subtract operator will take two attributes as an input and one attribute as an output. This operation can also be represented in the tree like structure subtracts operation at a root node of the tree and the two input variable act as a leaf node to the root node.

4.2.1 Genetic Programming selection

In Genetic programming [14] selection method is based on the tournament selection approach. In fig 4(c), we can see that there are eight parent individuals among them three tournaments were played in order to get the best individual of the tournament. In tournament C, was played among the eight individual I1,I2,I3,I4,I5,I6,I7 and I8 and winner of this tournament are I1,I4,I5 and I8. Among these individual tournament B is going to be played winner of these two tournaments are I1 and I5 were given a chance to get to have a face to face in the tournament A at end of I5 has won this tournament.

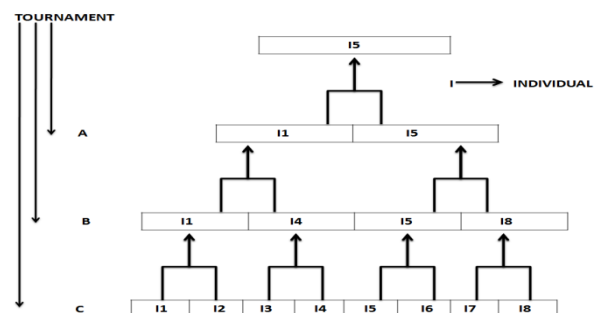


Fig 4(c): Genetic programming selection tournament

4.2.2 The Genetic Programming crossover

In this operation, we are going to select the parent node of the tree around which we will be performing mutation operation.

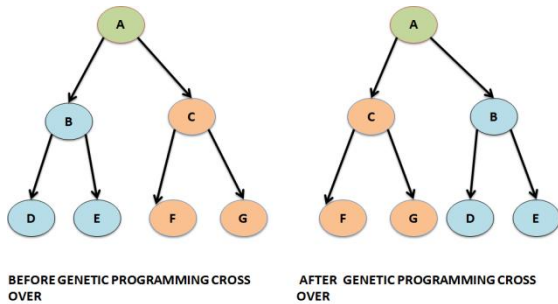


Fig 4(d): Genetic programming mutation

4.2.3 Genetic Programming mutation

This operation is performed on the two sub trees by chance from the parent where the parent swaps the left sub tree with the right sub tree. In Fig 4(d), we can see that Node A represented in green color act as a root node for the tree. The left sub tree contains node B,D,E represented by blue color and right sub tree contains node C,F and G represented by orange color. On this tree once we have performed the crossover operation then the right sub tree gets swapped with left sub tree which we can see clearly in below listed figure to be evaluated become more complex.

In this section, we had discussed the evolutionary scheduling algorithm used in cloud computing environment in detailed. But evolutionary scheduling algorithms were unable to meet the needs of cloud computing complex problems. Evolutionary algorithm seeks help from swarm intelligence algorithms in order to solve these complex problems efficiently. These evolutionary scheduling algorithms we are going to discuss in the next section in detail.

5. CONCLUSION

Bio Inspired algorithm in this new era of technology plays a very important role in various education fields. One of the important educational fields of technology is computer science which has started using this idea of bio inspired algorithm in various real time applications. The scope of bio inspired algorithm is very vast in computer science as only a fraction of bio inspired algorithm is used in this field. This paper presents an overview of evolutionary (GA, GP) and swarm (AC, FF, FP, CS, HB) based scheduling algorithm. Nowadays Bio inspired algorithm plays very important role in computer networks, data mining, power system, economics, robotics, information security, control system, image processing etc. There are great opportunities of exploring or enhancing this field algorithm with the help of innovative ideas or thoughts. Since this field of Bio inspired algorithm bridge a knowledge bond between different communities like computer science, biology, economics, artificial intelligence etc. At end we can conclude that the nature inspired algorithm are next era most power optimization algorithm which is going to play a key stealing role in next generation computing.

6. REFERENCES

[1] Dr. Amit Agarwal, Saloni Jain, Mar 2014: Efficient Optimal Algorithm of Task Scheduling in Cloud Computing Environment International Journal of Computer Trends and Technology (IJCTT) – volume 9 number 7.

[2] Anthony T.Velte ,Toby J.Velte ,Robert Elsenpeter ,Cloud computing a practical approach.

[3] Ehsan Valian, Shahram Mohanna and Saeed Tavakoli December 2011 : Improved Cuckoo Search Algorithm for Global Optimization, IJCIT-2011-Vol.1-No.1.

[4] Felix Streichert, University of Tuebingen, Introduction to Evolutionary Algorithms.

[5] Ms. D. Thilagavathi and Dr. Antony Selvadoss Thanamani August 2014, Scheduling in High Performance Computing Environment using Firefly Algorithm and Intelligent Water Drop Algorithm, International Journal of Engineering Trends and Technology (IJETT) – Volume 14 Number 1.

[6] Nitesh Sureja , November 2012: New Inspirations in Nature: A Survey International Journal of Computer Applications & Information Technology Vol. I, Issue III.

[7] Baris Yuce, Michael S. Packianather , Ernesto Mastrocinque , Duc Truong Pham November 2013, Alfredo Lambiasi 3Honey Bees Inspired Optimization Method: The Bees Algorithm, ISSN 2075-4450.

[8] Sankalp Arora, Satvir Singh, May 2013 : The Firefly Optimization Algorithm: Convergence Analysis and Parameter Selection, International Journal of Computer Applications (0975 – 8887) Volume 69– No.3.

[9] Xin-She Yanga, Mehmet Karamanoglu, Xingshi Heb , Year 2013: Multi-objective Flower Algorithm for Optimization, International Conference on Computational Science, ICCS 2013.

[10] S. D. Shtovba, March 2004 : Ant Algorithms Theory and Applications, Vol. 31, No. 4, 2005.

[11] O. Abdel-Raouf, M. Abdel-Baset, I. El-henawy, February 2014: A New Hybrid Flower Pollination Algorithm for Solving Constrained Global Optimization Problems, International Journal of Applied Operational Research Vol. 4, No. 2, pp. 1-13.

[12] Ashish Gosh , Satchidananda Dehuri , April 2004: Evolutionary Algorithm for Multi Criterion Optimization : A Survey , Volume 2, International Journals of Computing and Information Science.

[13] Sangita Roy, Sheli Sinha Chaudhuri, Year 2013: Cuckoo Search Algorithm using Lévy Flight: A Review, IJ. Modern Education and Computer Science.

[14] Koza, J.R. (1992). "Genetic Programming: on the programming of Computers by means of natural selection" MIT Press.

[15] http://www.mhhe.com/biosci/pae/botany/botany_map/articles/article_01.html.

[16] Xin-She Yang, Nature-Inspired Optimization Algorithms, School of Science and Technology, Middlesex University London, London.

[17] David Alejandro Pelta, Natalio Krasnogor , Dan Dumitrescu , Camalia Chira , Rodica Lung , Natural Inspired Cooperative Strategies for Optimization (NICSO2011).

[18] Melanie Mitchell, An introduction to genetic algorithm.