International Journal Of Engineering Research & Management Technology

ISSN: 2348-4039

September- 2014 Volume 1, Issue-5

Email: editor@ijermt.org www.ijermt.org

A survey of Metaheuristic Algorithm for Job Scheduling

Sudhanshu Prakash Tiwari
Assistant Professor
Lovel Professional University
Jalandhar , Punjab

Satyendra Singh
Assistant Professor
Lovel Professional University
Jalandhar , Punjab

Dr.Kapil Kumar Bansal
Head Research & Publication
SRM University NCR Campus
Modinagar Ghaziabad

ABSTRACT :
Job scheduling is a NP –hard problem in which we have to minimize the makespan time. Scheduling is the process of assigning resources to the jobs in such a way that all jobs get required resource in fairly manner without affecting one another. In this paper we have analysed different – different optimization algorithm cuckoo search algorithm, genetic algorithm, particle swarm optimization algorithm and hybrid algorithm and hybrid algorithm for job scheduling.

Keywords-job scheduling, cuckoo search algorithm, genetic algorithm, particle optimization algorithm, hybrid algorithm;

INTRODUCTION

job scheduling is a NP- hard problem.NP –hard problem is a set of all decision problem for which no solution exist in polynomial time but solution can be verified in polynomial time. Job scheduling is the process of assigning resources to the multiple jobs. Emphasis has been on investigating machine scheduling problems where jobs represent activities and machines represent resources; each machine can process at most one job at a time.

Some of the assumptions for the job scheduling problem are,

1. Jobs should be limited.
2. Each and every job contains a set of operations that needs to be performed.
3. Machines should be limited.
4. Each machine is capable of handling only one operation at a time.

Some of the constraints are,

1. No job should visit the same machines second time.
2. No condition among operation of various jobs.
3. Preemption type of operation is not allowed.
4. A single machine is capable of handling individual operation of a job at a time.
5. No machines fail during its operation.

CUCKOO SEARCH ALGORITHM

Cuckoo search (CS) is an optimization algorithm developed by Xin-she Yang and Suash Deb in 2009. It is based upon the obligate brood parasitism of some cuckoo species by laying their eggs in the host bird's nest .Some host birds can recognize the eggs of cuckoo. For example, if a host bird identifies the eggs are not their own, it will either throw these alien eggs away or simply abandon its nest and build a new nest elsewhere.

Cuckoo search is mainly consists of three rules:

1. Each cuckoo lays only single egg at a time, and dumps its egg in a randomly chosen nest.
2. The best nests with high quality of eggs will carry over to the next generation.
3.The hosts nests is fixed, and the egg laid by a cuckoo is recognized by the host bird with a probability p which lies in range from 0 to 1.Discovering operate on some set of worst nests, and discovered solutions dumped from farther calculations.

Based on the other evolutionary algorithms a cuckoo search algorithm for solving job scheduling is developed. The cuckoo's egg represents the solution of the problem.

Steps of cuckoo search algorithm are as:
Step-1 Generate an initial population of n host nests;
Step-2 While (t<MaxGeneration) or (stop criterion)
Step-3 Get a cuckoo randomly (say, i) and replace its solution by performing Lévy flights;
Evaluate its quality/fitness
For maximization
Choose a nest among n (say, j) randomly;
if (fitness function of cuckoo i >fitness function of cuckoo j)
Replace j by the new solution;
end if
A fraction (p_a) of the worse nests are abandoned and new ones are built
Keep the best solutions/nests;
Rank the solutions/nests and find the current best;
Pass the current best solutions to the next generation;
End while

Working of cuckoo search algorithm for job scheduling is as:
Based on the other evolutionary algorithms a cuckoo search algorithm for solving job scheduling is constructed .the cuckoo's egg represents the solution of the problem. Generate initial population is starting phase of algorithm and Cuckoo search algorithm finds the optimal solution for jobs having machines where each job contains a set of operation. We assume the number of machines with the number of operations is equal. We use operation-based presentation to show cuckoo's egg and nests, because in this type of presentation each permutation of cuckoo's eggs is a candidate solution .A user enters a n no of jobs and m no of machines. We should select order of machines in such a way that each job require minimum execution time. each operation of job should be implemented by one machine, so we cannot assign a operation to two machines simultaneously. To satisfy this assumption, we should define a penalty coefficient that applies to fitness function (FF). Fitness function is defined by the following formula:
FF= summation from j=1 to n∑ min (execution time)+FL;
Which j is the number of jobs, FL is penalty coefficient. If two machines implement an operation of jobs at same time then fitness function will increase by certain value of FL then we will discard the fitness function. By cuckoo optimization algorithm. After calculating the execution time of jobs and find an optimal sequence of jobs on machines, turn is the allocate machines based on incoming sequence .A machine may idle during execution time so we define a range of time in between a machines can be idle. To ensure the assumption that two machines not be allocate at a time, we define the another fitness function to prevent the occurrence of two machines should not be allocate at a same time. Fitness function can be calculated as follow:
FF= max (finish times of machines) + FL;

Step-1 start
Step-2 initialization of population
Step-3 selects the optimal sequence of machines for completing the jobs.
Step-4 calculates the value of first fitness function.
Step-5 assigns a machine to a job based on input sequence of operation.
Step-6 calculate the idle time of machines
Step-7 calculate the value of another fitness function
Step-8 end

IJERMT

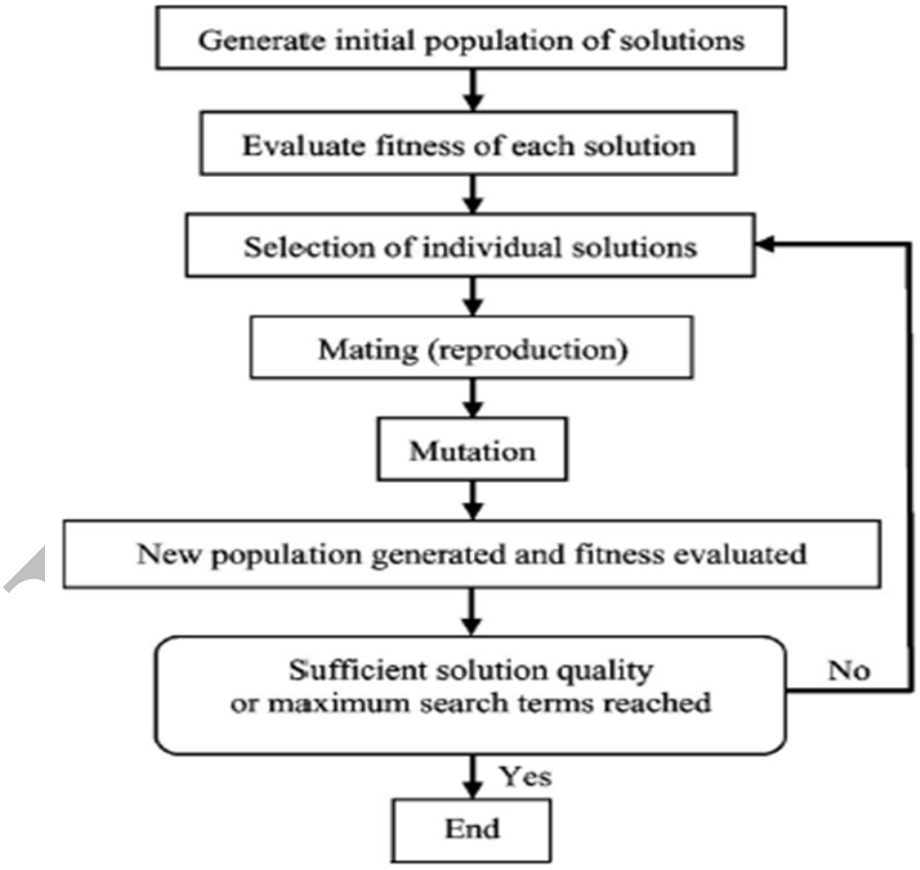
GENETIC ALGORITHM

The GA was firstly used by Holland (1975). It's mechanism is based on the simplifications of evolutionary process observed in nature. As opposed to many other optimization methods, GA starts working with a no of solutions instead of just a having a single solution. GA assigns a value to each individual in the population according to a problem-specific objective function. A selection step selects the fittest individual from the starting population. A crossover step is used to produce new child or offspring from selected parents. Mutation is used to maintain genetic diversity. GA is an optimization method of finding optimal solution based on evolutionary process. In applying GA, we have to analyse and understand the specific properties of problems and decide on a proper representation, an objective function, and a construction method of initial population, a genetic operators and a genetic parameter.

Steps of genetic algorithm are as follow:

Step 1- initialization
Step-2 selection
Step-3 cross over
Step-4 mutation
Step-5 condition satisfied goes to step 6 else go to step 2

Flowchart of genetic algorithm:



Working of genetic algorithm for job scheduling:

a. Chromosome representation: Genetic algorithm start with genetic representation. A good representation is necessary because it significantly affects the next steps of the GA. Many representations for the job scheduling problem have been developed. In this study, two representations based on working sequence and machine distribution are constructed. If the number

of the machine type t, the genes in each chromosome will be divided into t parts in turn. Each part represents one type of machine. Each operation can only be assigned to the machines which can handle it. For example suppose a chromosome is given as

[4 2 2 1 1 3 4 3 2 3 1 4 1 2 3 4] in 4 job×4 machines problem.

Here, 1 implies operation of job J1 , and 2 implies operation of job J2 and 3 implies operation of job J3 and 4 implies operation of job J4. Because there are four operations in each job, it appears the four times in a chromosome. Such as number 2 being repeated the forth in a chromosome, it implies four operations of job J2. The first number 2 represents the first operation of job J2 which processes on the machine 1. The second number 2 represents the second operation of job J2 which processes on the machine 2, and so on. The representation for such problem is based on two-row structure, as following:

Chromosome gene- 4 2 2 1 1 3 4 2 3 3 1 4 1 2 3 4

Operation machine- 1-1 1-2 1-3 1-4 2-1 2-2 2-3 2-4 3-1 3-2 3-3 3-4 4-1 4-2 4-3 4-4

b. Fitness function and selection: Fitness function is used to check the survival rate of an individual. It is relevant to the objective function to optimize. The higher the fitness of a chromosome is, the greater the probability to survive.

c. Crossover :

In crossover two parents are selected that are best fit for survival. parent are selected based on fitness value chances of selecting a parent is more having higher value of fitness value .higher value of fitness means it can survive to long time. Then parents are crossover to produce new children or offspring .these offspring goes to the next generation. Let assume that order of execution of operations are given. Parent has been selected from selection method.

Parent 1=(o11 ,o13 , o12 , o21 , o22 , o24 , o31 , o32 , o33 , o34 , o42);

Parent 2=(o14 ,o12 ,o13 ,o21 ,o23 ,o22 ,o31 ,o33 ,o32 , o41 ,o43 , o44);

A single point crossover is used to produce children.

offspring1=(011,o12 ,o13 ,o21 ,o23 ,o22 ,o31 ,o33 ,o32 , o41 ,o43 , o44);

offspring 2 =(o14, o13 , o12 , o21 , o22 , o24 , o31 , o32 , o33 , o34 , o42);

d. Mutation : Mutation is required to maintain genetic diversity.in mutation we change some value of genes in chromosome. After mutation solution may be different entirely from previous one.

PARTICLE SWARM OPTIMIZATION

Particle swarm optimization algorithm was introduced in 1995 by Kennedy and Eberhart. PSO algorithm is an adaptive method that can be used to solve optimization problem. Conducting search uses a population of particles .Each particle corresponds to individual in evolutionary algorithm. A flock or swarm of particles is randomly generated initially, each particle's position representing a possible solution point in the problem space. Each particle has an updating position vector X^i and updating velocity vector V^i moving through the problem space. Kennedy **and** Eberhart proposed the formula of updating position

$X^{i.}$

$$X^i_{k+1} = X^i_k + V^i_{k+1}$$

And formula for updating velocity vector V^i.

$$V^i_{k+1}=W_k V_k{}^i+C_1R_1(P_k{}^i-X_k{}^i) +C_2R_2(P_k{}^g -X_k{}^i)$$

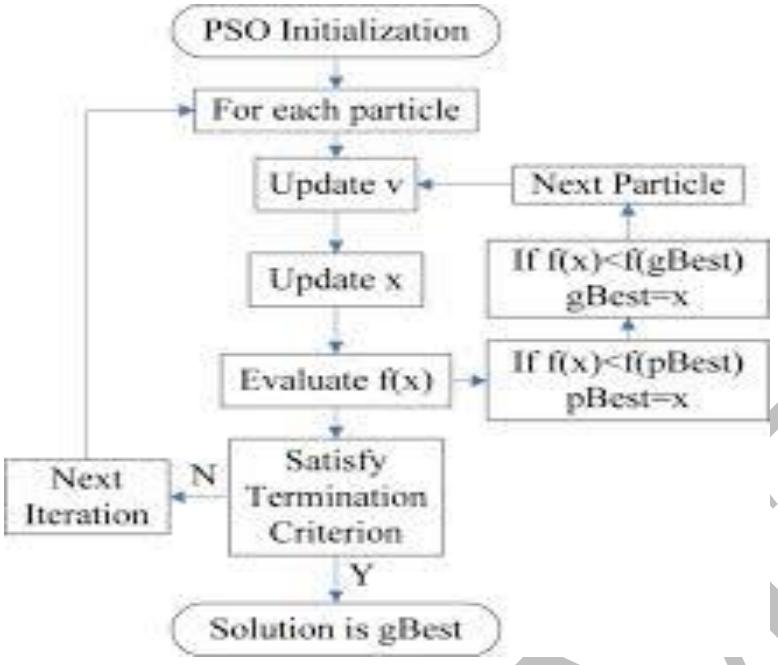
Where C_1 and C_2 are positive constant and R_1 and R2 are uniformly distributed random numbers between [0,1].

Velocity vector is of range [-Vmax ,Vmax]

At each iteration step, a function F_i is calculated by position vector X_i evaluating each particle's quality. The vector Pi represents ever the best position of each particle and **P**g represents the best position obtained

so far in the population. Changing velocity this way enables the particle to search around its individual best position Pi , and updating global best position Pg , until finds the best global position in the limited iteration.

Flowchart for pso algorithm:



Working of pso for job scheduling:
Initialize required parameters;
Initialize population randomly;
Initialize each particle position vector and velocity vector;
Find a permutation according to each particle's position;
Evaluate each particle and find the personal best and the
Global best
Do {
Update each particle's velocity and position;
Find a permutation based on updated position of each particle;
Evaluate each particle and update the personal best and the
Global best;
Apply the local search;
} While (Stop criterion)
}
END
For each particle personel best is defined as
$P^i_k = [P^i_1, P^i_2, \ldots\ldots\ldots\ldots P^i_n]$;
Where Pik is the position of Ith particle best with respect to the n dimension.
Global best is defined as
Gik=[Gi1 ,Gi2,……..,Gin];
Where Gik is the position value of the global best with respect to then n dimension

IJERMT

CONCLUSION

All optimization algorithms gives optimal solution for job scheduling problem. Cuckoo optimization algorithm provides better result compared to genetic algorithm and particle swarm optimization algorithm because of its fast local search characteristics. So in future more hybrid algorithms can be applied to job scheduling problem to obtain better result

REFERENCES:

1. Xin-SheYang,Suash Deb,"Cuckoo Search via Levy Flights" 2009 World Congress on Nature & Biologically Inspired Computing (NaBIC 2009)
2. M. Z. Rashad , A.E. Keshk,,M..A El-Douskey "Genetic Cuckoo Optimization Algorithm" International Journal of Computer Applications (0975 – 8887) Volume 90 – No.3, March 2014
3. Mitsuo Gen,Yasuhiro Tsujimura "solving job shop scheduling problem using genetic algorithm" 0-7803-2129-4/94 $3.00 Q 1994 IEEE
4. W.C.E.Lim ,G.Kangaraj,S.G Ponnambalam "hybrid algorithm using cuckoo search algorithm and genetic algorithm for hole making sequences"© Springer Science+Business Media New York 2014
5. Ren Qing-dao-er-ji,Yuping Wang "A new hybrid algorithm for job shop scheduling" Computers & Operations Research 39 (2012) 2291–2299
6. Ala'a Ahu-Srhan,Essan Al Daoud " a hybrid algorithm using cuckoo search algorithm and genetic algorithm for travelling salesman problem "International Journal of Advanced Science and Technology Vol.61, (2013), pp.29-38 http://dx.doi.org/10.14257/ijast.2013.61.04
7. Maryam Robiee ,Hedich Sajedi "job scheduling in grid computing using cuckoo optimization algorithm"
8. International Journal of Computer Applications (0975 – 8887) Volume 62– No.16, January 2013
9. Ye Li Yon Chen "a genetic algorithm for job shop scheduling"journal of software vol 5no 3 march 2010
10. Liang Sun,Xiaochum Cheng,Yanchum Liang "solving job shop scheduling problem using genetic algorithm with penalty function" International Journal of Intelligent Information Processing Volume 1, Number 2, December 2010
11. Leizhang,Yuehui chen,Runyuvan Sun,Shan Jing,Bo Yang "particle swarm optimization algorithm for task scheduling "International Journal of Computational Intelligence Research. ISSN 0973-1873 Vol.4, No.1 (2008), pp. 37–43 © Research India Publications
12. Hesam Izakian,Beh rouzTook Ladani,Kamran Zamanifar,Ajit Abraham"A Novel Particle Swarm Optimization Approach for Grid Job Scheduling" Norwegian Center of Excellence, Center of Excellence for Quantifiable Quality of Service, Norwegian University of Science and Technology, Trondheim, Norway
13. Vikas singh,Deepak Singh,Shyam Swarup "Grid scheduling using PSO with navie crossover" International Journal of Computer Applications (0975 – 8887)Volume 26– No.11, July 2011
14. Marayam Rabiee,Hedieh Sajedi "Job Scheduling in Grid Computing with Cuckoo Optimization Algorithm" International Journal of Computer Applications (0975 – 8887) Volume 62– No.16, January 2013
15. Ms. Hetal R. Soneji1andMr. Rajesh C. Sanghvi" Towards the Improvement of Cuckoo Search Algorithm" International Journal of Computer Information Systems and Industrial Management Applications.ISSN 2150-7988 Volume 6 (2014) pp.77 – 88 © MIR Labs,www.mirlabs.net/ijcisim/index.html
16. Kennedy, J.; Eberhart, R. (1995). "Particle Swarm Optimization". Proceedings of IEEE International Conference on Neural Networks **IV**. pp. 1942–1948. doi:10.1109/ICNN.1995.488968.
17. Shi, Y.; Eberhart, R.C. (1998). "A modified particle swarm optimizer". Proceedings of IEEE International Conference on Evolutionary Computation. pp. 69–73
18. Poli, R. (2007). "An analysis of publications on particle swarm optimisation applications". Technical Report CSM-469 (Department of Computer Science, University of Essex, UK).
19. Clerc, M. (2012). "Standard Particle Swarm Optimisation". HAL open access archive.