International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
SKRGC Since 2012
http://www.ijcsjournal.com Volume 5, Issue 1, No 22, 2017 ISSN: 2348-6600
Reference ID: IJCS-264 PAGE NO: 1684-1690

Alagappa University, Karaikudi, India 15th -16th February 2017
IT Skills Show & International Conference on Advancements in Computing Resources (SSICACR-2017)
http://aisdau.in/ssicacr ssicacr2017@gmail.com

ENHANCED PARTICLE SWARM OPTIMIZATION ALGORITHM FOR JOB SCHEDULING PROBLEM

V.Selvi
Assistant professor of computer Science Department
Mother Teresa Women's University
Kodaikanal.Tamilnadu,India.
selvigiri.s@gmail.com

Abstract – This paper presents the hybrid approach of two natures inspired metaheuristic algorithms; simulated annealing and Particle Swarm Optimization (PSO) is used for solving optimization problems. The population-based stochastic global search algorithm is known as Cuckoo Search. The job scheduling (JS) is one of the most studied operational research and computer science. Research is produced to a large number of techniques to resolve this problem, the results obtained by is when compared to other techniques. This paper propose a hybrid algorithm, namely PSO-SA, based on Particle Swarm Optimization (PSO) and Simulated Annealing (SA) algorithms. The hybrid PSO algorithm is not only in the structure of the algorithm, but also the search mechanism provides a powerful way to solve JSSP. Experimental results are examined with the job scheduling problem and the results show a promising performance of this algorithm. The outcomes prove that the proposed hybrid algorithm is an efficient and effective tool to solve the JSSP.

Keywords: Particle Swarm Optimization, Simulated annealing, Job Scheduling, Swarm Intelligence, Enhanced Particle swarm optimization.

I. INTRODUCTION

An Evolutionary swarm intelligence technique is well-designed in this paper for solving job scheduling problem. It is a combinatorial optimization problem, where the jobs are arranged, so that they may or may not be processed in every machine sequence [2]. Then each machine consist of an unique sequencing of jobs[5]. An extended version of flow scheduling is known as job scheduling. The main purpose of both kinds of problems is jobs in a sequence that gives a minimum value of makespan.

In this paper PSO is used to solve job scheduling problem. The next most common population based algorithm is used Simulated Annealing. Then the algorithm is extended after understanding particle swarm optimization algorithm and is applied in job scheduling problem for n jobs and m machines.

Efficiency plays an important role if jobs and machines are considered in large numbers. Set of jobs and machines are there and each job consists of chain and is processed during an uninterrupted time period of that particular length on a given machine. Only one process can be done at a time. The main aim of this algorithm is to find the

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 5, Issue 1, No 22, 2017 ISSN: 2348-6600
Reference ID: IJCS-264 PAGE NO: 1684-1690
Alagappa University, Karaikudi, India 15th -16th February 2017
IT Skills Show & International Conference on Advancements in Computing Resources (SSICACR-2017)
http://aisdau.in/ssicacr ssicacr2017@gmail.com

schedule of minimum length where a schedule is an allocation of operations to time intervals of the machines.

Efficient solutions in a polynomial time can be obtained using heuristic algorithms. Individual task scheduling of heuristic algorithms is simple and suitable for sequential workflow scheduling of tasks. A list scheduling algorithm is used to set task priority and form an efficient schedule, as many tasks compete for limited resources. Priority of tasks determines tasks execution order. Batch mode scheduling algorithms arrange tasks to be executed in parallel based on execution time. Dependency mode scheduling algorithms are for other tasks based on their critical path length. Interdependent tasks with longer execution time are scheduled first to reduce overall execution time in dependency mode scheduling algorithms. Its complexity is higher due to computation of tasks critical path. If a workflow has many parallel tasks with same critical path length, it cannot solve resource competition problem in the dependency mode scheduling algorithm. Dependency batch mode shows better performances than dependency mode as it uses the advantages of both scheduling algorithms (Sakellariou & Zhao 2004). It takes higher scheduling time for computation of task priorities as it uses both batch mode and dependency mode approaches. List scheduling will not be given an efficient schedule for data intensive workflow applications due to the transfer of data between interdependent tasks during execution time. The objective of list scheduling is to find the effective execution order for available parallel tasks and identification of the best resource for tasks based on details of current task. Communication delay reduction among interdependent tasks is the main objective of cluster based scheduling as well as duplication based scheduling. All interdependent tasks with heavy data communication requirements are grouped into the same task clusters to minimize data transmission time and tasks in the same task cluster are scheduled to the same resource for execution, to reduce data transmission time. In duplication based scheduling, interdependent tasks are duplicated to minimize data transmission time. These methods may create problems in grid applications due to the condition of scheduling tasks with high communication requirements on the same resource. As the workflow applications in grid services are heterogeneous, different types of resources are needed for their execution. Meta-heuristic scheduling produces an optimal solution based on the entire workflow performance, but a heuristic based approach considers only partial workflow performance while scheduling a workflow. Heuristic workflow scheduling is designed for a specific type of workflow application, but meta heuristic scheduling algorithms produce good solutions for a different workflow applications. But scheduling time required to schedule meta-heuristic scheduling algorithms is significantly high compared to other scheduling algorithms. Heuristic workflow scheduling algorithms suit simple workflow scheduling applications whereas meta-heuristic workflow scheduling methods can schedule large and complex workflows.

The methods of Particle swarm optimization (PSO) and simulated annealing (SA) have been

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 5, Issue 1, No 22, 2017 ISSN: 2348-6600
Reference ID: IJCS-264 PAGE NO: 1684-1690
Alagappa University, Karaikudi, India 15th -16th February 2017
IT Skills Show & International Conference on Advancements in Computing Resources (SSICACR-2017)
http://aisdau.in/ssicacr ssicacr2017@gmail.com

studied in this paper and are being applied to the job scheduling case so as to get an optimum processing time. Recent studies illustrate that SA is potentially far more efficient than PSO, genetic algorithms, and other algorithms. Many of the recent research studies focused on metaheuristic approaches such as ACO_PSO, PSO_GA and ABC_GA with job scheduling. In this paper the above hybrid techniques are completed with minimum make span and completion time. In this paper an Improved Particle Swarm Optimization (EPSO) algorithm is used to solve JSP. This is also hybrid technique which is better than the above existing algorithm makespan and completion time is minimized when compared to other obtainable algorithm.
The rest of this report is coordinated as follows: Section 2 briefly describes about the Swarm intelligence techniques. Section 3 discusses about Experimental results and discussion and finally some conclusion are made in Section 4.

II. SWARM INTELLIGENCE

Swarm intelligence (SI) is that the collective behavior of decentralized, self-fashioned systems, natural or artificial. The conception is utilized in work on computing. Swarm Intelligence principles have been successfully applied in a variety of problem domains together with function optimization problems, finding optimal routes and scheduling [10][11].

A. Particle Swarm Optimization

Particle swarm optimization is a population based stochastic optimization technique for the resolution of continuous optimization problems (Kennedy and Eberhart -1995)[2]. In particle swarm optimization (PSO), a set of software agents called particles search for best solutions to a specified continuous optimization problem. In practice, an initialization phase of each particle is thrown a random initial position and an initial speed (velocity)[3]. The position of the particle represents an elucidation of the problem and therefore has a value, interpreted by the objective intent. At every iteration of the algorithm, every constituent part moves with a speed that is a weighted summation of three components: the old velocity, a velocity component that forces the particle towards the positioning in the search space where the best result is found so far, and a velocity component that forces the particle towards the positioning in the search space where the neighbour particles found the best answer so far. In PSO, each single solution is a "bird" in the search space Called "particle". The operation of a particle is precise by a fitness value.
The particles fly through the problem space by falling out the current optimum particles. PSO is initialized with a group of arbitrary particles and then searches for optima by updating generations. The situation of a particle is determined by the best position visited by it. Each particle knows its best position pbest and the best position so far among the full group of particles gbest.
After finding the two best values, the particle updates its velocity and positions with the following equation (1) and (2).
v[] = v[] + c1 * rand() * (pbest[] - present[]) + c2 * rand() * (gbest[] - present[])- (1)
present[] = present[] + v[] ---------------(2)

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 5, Issue 1, No 22, 2017 ISSN: 2348-6600
Reference ID: IJCS-264 PAGE NO: 1684-1690
Alagappa University, Karaikudi, India 15th -16th February 2017
IT Skills Show & International Conference on Advancements in Computing Resources (SSICACR-2017)
http://aisdau.in/ssicacr ssicacr2017@gmail.com

v[] is the particle velocity, present[] is the current particle. pbest[] and gbest[] are defined as stated before rand () is a random number between (0,1). c1, c2 are learning factors. Usually c1 = c2 = 2.

The pseudo code for PSO is as follows

```
For each particle
   Initialize particle values
END
Do
   For each particle
      Compute  the fitness value
      If the fitness value is best than pBest value
        Current value= pBest
   End
  Work out the particle velocity according the
particle with the best fitness value of all the
particles as the gbest
   For every particle
      Work out the particle velocity according
equation ------ (1)
Change particle position according equation--(2)
                End
  While  Minimum error criteria or Maximum
          iterations are not achieved.
```

B. Simulated annealing (SA)

Simulated annealing (SA) (Metropolis et al 1953) is a method to search global space statistically using the Monte Carlo method. The core idea is from the Annealing Process of crystalline structures to form a more ordered state by repeated heating and slow cooling. Performance of SA algorithms in scheduling, workflow applications in grid services is evaluated in Young et al (2003). A general Simulated Annealing algorithm is explained in Figure 2.1.

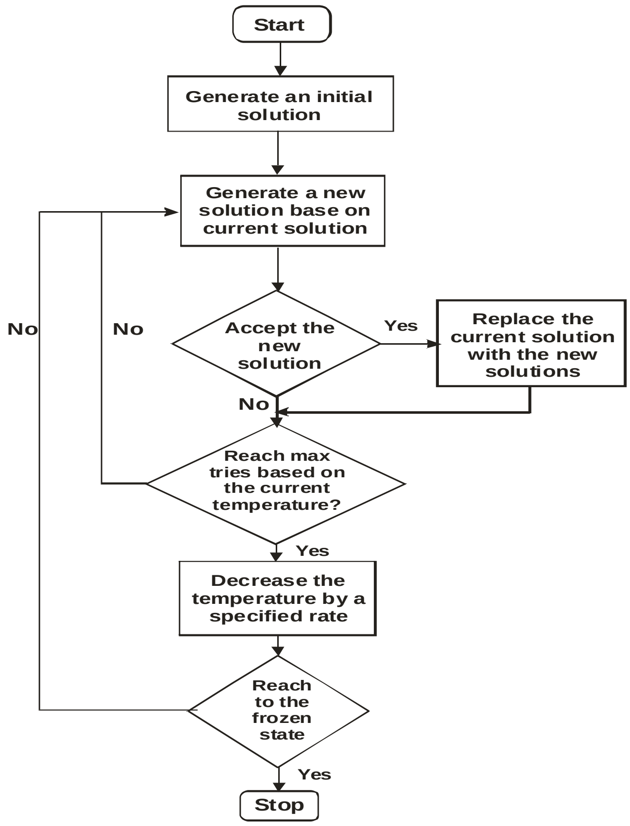

Figure 2.1 Simulated annealing

The initial input of SA algorithm is formed by the random allocation of resource to a task. Each time the temperature is lowered in simulated annealing algorithms, many steps have to be performed. A number of iterations are needed to sample search space in an annealing process. A new solution is formed in each cycle by making random changes in the current solution. Randomization can be introduced by a task to be scheduled on different resource (Young et al 2003). Metropolis algorithm (Metropolis

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
SKRGC Since 2012
http://www.ijcsjournal.com Volume 5, Issue 1, No 22, 2017 ISSN: 2348-6600
Reference ID: IJCS-264 PAGE NO: 1684-1690
Alagappa University, Karaikudi, India 15th -16th February 2017
IT Skills Show & International Conference on Advancements in Computing Resources (SSICACR-2017)
http://aisdau.in/ssicacr ssicacr2017@gmail.com

et al 1953, Young et al 2003) is used to determine the current solution by comparing the new solution and the existing one. If the new solution is better than the current one, the new solution is accepted as the current solution. Lower execution time is considered the best solution in workflow scheduling minimization problems. The temperature decreases, once the defined number of cycles are completed. The temperature decreases till the lowest permissible temperature. During

S. No	Number of Task	PSO	SA	IPSO
1	50	2510	2480	2310
2	100	2930	2749	2230
3	150	2992	2900	2450
4	200	3230	3150	2920
5	250	3570	3412	3090
6	300	3856	3670	3450
7	350	4010	3850	3410
8	400	4230	4089	3700
9	450	4570	4150	3830
10	500	5010	4715	4050

termination, simulated annealing algorithm gives the best solution obtained from the beginning.

III. EXPERIMENTAL RESULTS AND DISCUSSION

A. Implementation and Results

In this paper, the result obtained by the proposed algorithm is analyzed. To test the efficiency of the algorithm results of IPSO is compared with the results of the PSO and SA algorithms. The experiments by varying the number of jobs with 50 resources are conducted and then the results with that of two existing algorithms are compared. The comparative results are shown in the following Table 1 and Fig. 3.1.

Table 1: Mean Makespan time for 50 to 500 Jobs with 50 resources for PSO, SA and IPSO algorithm

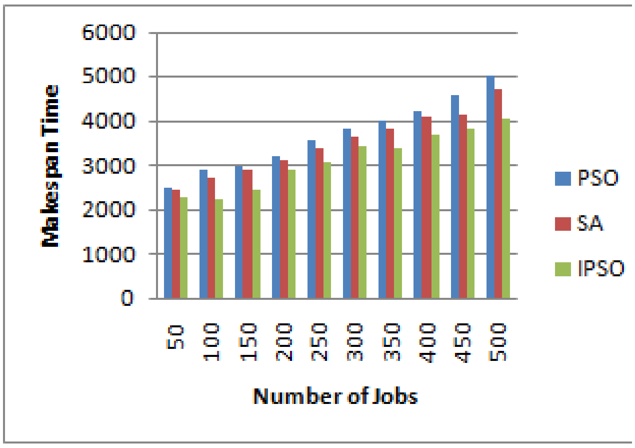

Fig.3.1. Mean Makespan time for 50 to 500 Jobs with 50 resources for PSO, SA and IPSO algorithm.

IV. CONCLUSION AND FUTURE WORK

The proposed method EPSO has achieved the minimum makes span time. In this paper work, an Enhanced Particle swarm Optimization technique is proposed to allocate the available jobs to the exact resources. The experimental result shows that the proposed job scheduling technique has attained high accuracy and efficiency than the two existing techniques PSO, SA. In the PSO optimization algorithm, a velocity value is utilized for

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 5, Issue 1, No 22, 2017 ISSN: 2348-6600
Reference ID: IJCS-264 PAGE NO: 1684-1690
Alagappa University, Karaikudi, India 15th -16th February 2017
IT Skills Show & International Conference on Advancements in Computing Resources (SSICACR-2017)
http://aisdau.in/ssicacr ssicacr2017@gmail.com

generating the new solutions. Hence, the Enhanced Particle Swarm Optimization algorithm for job scheduling technique is capable of finding the optimal jobs to the resources and also in achieving the minimum makespan time.

REFERENCES:

1. Gandomi, A.H., Yang, X.S., Alavi, A.H.: Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems. Eng. Comput. 29, 17–35 (2013).

2. Tsung- Lieh Lin et al., An efficient job-shop scheduling algorithm based on particle swarm optimization, Expert Systems with Applications, Volume 37, Issue 3, 15 March 2010, Pages 2629-2636.

3. Xinyu Shao et al., Hybrid discrete particle swarm optimization for multi-objective flexible job-shop scheduling problem,The International Journal of Advanced Manufacturing Technology ,August 2013, Volume 67, Issue 9-12, pp 2885-2901.

4. Xin-She Yang, Xingshi He , Swarm Intelligence and Evolutionary Computation: Overview and Analysis, Recent Advances in Swarm Intelligence and Evolutionary Computation Studies in Computational Intelligence Volume 585, 2015, pp 1-23 Dec 2014.

5. Albodour, R, James, A & Yaacob, N 2012, ‘High level qos-driven model for grid applications in a simulated environment’, Future Generation Computer Systems vol. 28, no. 7, pp. 1133-1144.

6. D.Y. Sha, Hsing-Hung Lin ,” A Multi-Objective PSO for Job-Shop Scheduling Problems. Expert Systems With Applications”, Volume 37, Issue 2, March 2010, Pages 1065–1070.

7. Hazem Ahmed,Janice Glasgow,”Swarm Intelligence: Concepts, Models and Applications”, School of Computing,Queen's University, Kingston, Ontario, Canada K7L3N6, February 2012.

8. P. Lim, L. C. Jain, S. Dehuri, ”Innovations in Swarm Intelligence. Studies in Computational Intelligence”, Vol. 248, Springer, 2009.

9. Xin-She Yang,Suash Deb, Cuckoo search: recent advances and applications, Neural Computing and Applications, January 2014, Volume 24, Issue 1, pp 169-174,Mar 2013.

10. Derrac, J., Garcia, S., Molina, D., Herrera, F.: A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm Evol. Comput. 1, 3–18 (2011).

11. Xin-She Yang, Swarm intelligence based algorithms: a critical analysis, Evolutionary Intelligence April 2014, Volume 7, Issue 1, pp 17-28 Date: 17 Dec 2013.

12. Lixin Tang; Xianpeng Wang, An Improved Particle Swarm Optimization Algorithm for the Hybrid Flowshop Scheduling to Minimize Total Weighted Completion Time in Process Industry, Control Systems Technology, IEEE Transactions

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 5, Issue 1, No 22, 2017 ISSN: 2348-6600
Reference ID: IJCS-264 PAGE NO: 1684-1690
Alagappa University, Karaikudi, India 15th -16th February 2017
IT Skills Show & International Conference on Advancements in Computing Resources (SSICACR-2017)
http://aisdau.in/ssicacr ssicacr2017@gmail.com

on Year: 2010, Volume :18, Issue: 6, Pages : 1303 – 1314.

13. Yue-Jiao Gong et al, Optimizing the Vehicle Routing Problem With Time Windows: A Discrete Particle Swarm Optimization Approach Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on Year: 2012, Volume: 42, Issue: 2, Pages: 254 – 267.

14. Udayraj et al, Performance analysis and feasibility study of ant colony optimization, particle swarm optimization and cuckoo searchalgorithms for inverse heat transfer problems, International Journal of Heat and Mass Transfer, Volume 89, October 2015, Pages 359-378.

15. Thang Trung Nguyen, Dieu Ngoc Vo Modified cuckoo search algorithm for short-term hydrothermal scheduling ,International Journal of Electrical Power & Energy Systems, Volume 65, February 2015, Pages 271-281

16. Hao Gao; Sam Kwong; Baojie Fan; Ran Wang,A Hybrid Particle-Swarm Tabu Search Algorithm for Solving Job Shop Scheduling Problems Industrial Informatics, IEEE Transactions on Year: 2014, Volume: 10, Issue: 4,Pages: 2044 – 2054.