

# Comparative Analysis of Metaheuristic Approaches for Makespan Minimization for No Wait Flow Shop Scheduling Problem

Laxmi A. Bewoor<sup>1</sup>, V. Chandra Prakash<sup>2</sup>, Sagar U. Sapkal<sup>3</sup>

<sup>1,2</sup>Computer Science & Engg. Dept., K. L. University, Guntur-500002, India

<sup>3</sup>Mechanical Engg. Dept., W. C. o. E., Shivaji University, Sangali-416415, India

---

## Article Info

### Article history:

Received Jun 22, 2016

Revised Aug 29, 2016

Accepted Sep 13, 2016

---

### Keyword:

Makespan

Metaheuristic

No wait flow shop

NP hard

---

## ABSTRACT

This paper provides comparative analysis of various metaheuristic approaches for m-machine no wait flow shop scheduling (NWFSS) problem with makespan as an optimality criterion. NWFSS problem is NP hard and brute force method unable to find the solutions so approximate solutions are found with metaheuristic algorithms. The objective is to find out the scheduling sequence of jobs to minimize total completion time. In order to meet the objective criterion, existing metaheuristic techniques viz. Tabu Search (TS), Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) are implemented for small and large sized problems and effectiveness of these techniques are measured with statistical metric.

Copyright © 2017 Institute of Advanced Engineering and Science.  
All rights reserved.

---

## Corresponding Author:

Laxmi A. Bewoor,  
Computer Science & Engg. Dept.,  
K. L. University,  
Guntur-500002, India.  
Email: laxmiabewoor@gmail.com

---

## 1. INTRODUCTION

Manufacturing scheduling is concerned with setting the timetable for the processing of given set of jobs on the set of machines in order to optimize given measure of performance. Efficient scheduling is need of an hour for manufacturing industry to survive in today's intensely competitive business environment. Manufacturing scheduling is broadly classified into flowshop scheduling, job shop scheduling and Openshop scheduling. In recent years, a considerable amount of interest has seen in no wait scheduling problem not only from research aspect but also because of wide variety of industrial applications. No wait flow shop scheduling is one of the variant of flow shop scheduling, in which job must be processed from start to finish, without any interruption and pre-emption on or between machines.

Applications of no-wait flow shop scheduling (NWFSS) can be found in many industries such as steel industry, plastic moulding industry, process industries, chemical and pharmaceutical industries, concrete ware production, electronic industry, Food industry etc. Hall and Sriskandarajah [1] provided a detailed survey of applications and related research problems of the no-wait flow shop scheduling. Subsequently, NWFSS problem is addressed from research point as well because problem is NP hard in nature and treated as an example of combinatorial optimization. This type of class of problems finds an optimal solution from a diverse search-space. In this case, the search-space of candidate solutions grows exponentially as the size of the problem increases, and it is very difficult to make an extensive search for the optimal solution with traditional enumerative methods. In case of NWFSS, decision about optimal sequence of given n jobs to be processed on given m machines should be taken from n! combinations of job sequences for minimizing (or

maximizing) the optimality criterion. Makespan, total flow time, mean flow time, idle machine time, total tardiness, number of tardy jobs, in-process inventory cost, and cost of being late are some of the optimality criterion in context with NWFSS scheduling problem. Makespan is one of the most important performance measures because it decides total length of schedule. In practice a product can be delivered if all the subsequent jobs required to make that product get executed within given schedule. Therefore, minimizing makespan is a very important objective for scheduling in many NWFSS systems.

In recent years, Metaheuristics are preferred more than heuristic for solving combinatorial optimization problems because they guide and modify heuristics and less pruned to get stuck in local optima. The classical metaheuristic algorithms of this type include Ant Colony Optimization (ACO), Evolutionary Algorithms (EA), Particle Swarm Optimization (PSO), and Genetic Algorithms (GA) etc. [2]. Researchers made few attempts for comparative analysis of metaheuristic approaches for various optimization problems. Hassan et al. [3] compared performance of GA and PSO for set of benchmark test problems and design optimization problems of telescope array configuration and spacecraft reliability-based design problem. The authors claimed that PSO has the same effectiveness as that of GA for finding the true global optimal solution but with significantly better computational efficiency. Statistical analysis and formal hypothesis testing were applied to validate number of comparison of function evaluations of GA and PSO. Jamian et al. [4] studied that the size of Distributed Generation (DG) is crucial in order to reduce the impact of installing a DG in the distribution Network. They implemented PSO and evolutionary PSO (EPSO) for formulating optimization technique to regulate the DG's output to compute its optimal size so that power loss gets reduced and voltage get regulated.

The main objective of this paper is to compare effectiveness of some metaheuristic techniques for minimizing makespan for NWFSS problem. The remaining paper is organized as follows: Section 2 briefs about literature review, Section 3 defines NWFSS problem precisely, Section 4 discusses computational experiences and Section 5 discusses meaningful conclusions.

## 2. LITERATURE REVIEW

Enumerative methods got failure when the job size increases which generates the necessity of developing approximation algorithms. So attempts were made for developing heuristic algorithms for finding near optimal solution but still the heuristic approaches suffered with costly development of specialized algorithm which hinder when applying to real life problems. This major problem can be solved with metaheuristic approaches which are widely generic with respect to type of problem. The paper provides a decade literature study of contribution by various researchers for solving scheduling problems. Production scheduling problems were studied majorly with two optimality criteria total flow time (TFT) and total completion time (makespan). So the review of all the literature pertinent to the problem under study with makespan is presented in this section.

Lu and Gu [3] considered the real-world problems in the Batch Plant Industry and introduced fuzzy processing time and due date window into the Flow shop scheduling problems. The fuzzy Flow shop scheduling model with minimizing the makespan and minimizing the penalties of the Em was set up based on the theory of fuzzy programming, by applying the Maximum Membership Functions of Mean Value (MMFMV). The proposed model convert the fuzzy optimization problem to the general optimization problem and further solved with improved simulated annealing (SA) algorithm.

Xia and Wu [4] developed approximation algorithm for the problem of finding the minimum makespan in the job-shop scheduling environment. The new algorithm was based on the principle of particle swarm optimization (PSO) and for high search efficiency and Simulated Annealing (SA) to avoid trap in local optima. By reasonably combining these two different search algorithms, a general, fast and easily implemented hybrid optimization algorithm HPSO was developed.

Liu et al. [5] proposed an effective hybrid algorithm based on particle swarm optimization (PSO) for no-wait flow shop scheduling with the criterion to minimize the maximum completion time. The algorithm uses random key encoding scheme along with local search based on the Nawaz-Enscore-Ham (NEH) heuristic and simulated annealing (SA) with an adaptive meta-Lamarckian learning strategy.

Li et al. [6] proposed an objective increment heuristic method no-wait flow shops with makespan minimization. The proposed model can calculate makespan of a new schedule directly from that of its parent and decision could be made about quality of new schedule in comparison with its parent. Additionally composite heuristic based on makespan increment was proposed which needs minimal CPU time.

Pan et al. [7] developed a discrete particle swarm optimization (DPSO) algorithm is presented to solve the no-wait flowshop scheduling problem with both makespan and total flow time criteria. The particles in this algorithm were represented as discrete job permutations and a new position update method is

developed based on the discrete domain. Further DPSO algorithm was hybridized with the variable neighbourhood descent (VND) algorithm for improving the solution quality.

Sha and Shu [8] presented a new particle swarm optimization (PSO) for the open shop scheduling problem to minimise makespan. In the proposed method the particle position were represented using priorities, and the particle movement using an insert operator. A modified parameterized active schedule generation algorithm (mP-ASG) was used to decode a particle position into a schedule and algorithm was hybridized with beam search.

Pan et al. [9] developed a novel hybrid discrete particle swarm optimization (HDPSO) algorithm to solve the no-wait flow shop scheduling problems for minimization of the maximum completion time. Discrete particle swarm optimization (DPSO) algorithm based on permutation representation and a local search algorithm based on the insert neighbourhood were integrated for enhance the searching ability and balancing the exploration and exploitation.

Pan et al. [10] proposed improved iterative greedy algorithm for minimizing makespan. The proposed method uses an improved Nawaz-Enscore-Ham (NEH) heuristic for constructing solutions initially and for searching sequences with a simple local search algorithm was incorporated into the iterated greedy algorithm to perform exploitation.

Laha and Chakraborty [11] presented a new constructive heuristic, based on the principle of job insertion, for minimizing makespan in no-wait permutation flow shop scheduling problems. The proposed algorithm constructs n-job sequences incrementally, by using shift neighbourhood mechanism in generating the initial sequence. Analytical expressions for the total number of partial and complete sequences generated by the algorithms are derived.

Kuo et al. [12] developed a new hybrid particle swarm optimization model (HPSO) that was combining random-key (RK) encoding scheme, for exploiting the global search ability of PSO and individual enhancement (IE) scheme for enhancing the local search ability of particles, and particle swarm optimization (PSO) for solving the flow-shop scheduling problem (FSSP) with makespan as an objective. The objective of FSSP is to find an appropriate sequence of jobs in order to minimize makespan.

Gao et al. [13] presented a discrete harmony search (DHS) algorithm for solving no-wait flow shop scheduling problems with makespan criterion. The proposed model builds harmony that was represented as a discrete job permutation and heuristic methods were used to initialize the harmony memory later with dynamically regrouping mechanism, the harmony memory was divided into several groups for sharing information reciprocally. Variable neighbourhood search algorithm was developed and embedded in the DHS for providing a balance between the global exploration and local exploration.

Chakaravarthy et al. [14] derived solution for n-job,  $m$ -machine lot streaming problem in a flow shop with equal size sub lots for minimizing the makespan and total flow time as objective functions. They proposed a Differential Evolution Algorithm (DEA) and Particle Swarm Optimization (PSO) to evolve best sequence for makespan/total flow time criterion. The proposed method was using the DEA and PSO algorithms for discrete lot streaming with equal sub lots.

Donald et al. [15] proposed Discrete Self-Organizing Migrating Algorithm (SOMA) which is a class of swarm heuristic based on the competitive-cooperative behaviour of intelligent creatures solving a common problem. The proposed model was discrete variant of SOMA for solving permutative variants of scheduling problems. The method looks for the best individual from the optimized model's hyperspace for minimizing makespan.

Shao et al. [16] developed a hybrid discrete particle swarm optimization (DPSO) and simulated annealing (SA) algorithm to identify an approximation of the Pareto front for flexible job-shop scheduling problem (FJSP). The proposed method addressed FJSP with makespan, maximal machine workload and total workload minimization. In the proposed hybrid algorithm, DPSO was significant for global search and SA was used for local search and Pareto ranking and crowding distance method were incorporated for finding the fitness of particles. The local best positions of particles were used to store the fixed number of non-dominated solutions.

The review of literature reveals that, though the scheduling problem has many practical applications, the literature related with this is very limited. The major literature focuses on variety of flow shop scheduling problem but NWFSS has comparatively less addressed scheduling problem. It is also observed that, limited efforts were attempted to compare different metaheuristic for NWFSS problem with makespan as an optimality criteria. Additionally attempts should be made for investigation of efficient algorithm for solving NWFSS problem with makespan as an objective function for large size problem.

In line with the said objective, this paper presents a comparative analysis of metaheuristic algorithms for  $m$ -machine NWFSS for finding optimal scheduling sequence considering minimization of makespan as an objective criterion. In order to find the near optimal solution, for NWFSS problem, population based technique viz. Tabu Search (TS), evolutionary technique viz. Genetic Algorithm(GA) and

population based evolutionary technique viz. Particle Swarm Optimization (PSO) are evaluated for small and large problem size.

### 3. NO WAIT FLOW SHOP SCHEDULING PROBLEM (NWFSSP)

The no-wait flow shop scheduling problem (NWFSSP) can be described as follows. Given  $n$  jobs are to be processed sequentially on machines  $1, 2, \dots, m$ . Processing time ( $PT_{ij}$ ) of  $i^{\text{th}}$  job on  $j^{\text{th}}$  machines is given. Every machine can process at most one job at given instance. The processing sequence of jobs is same on every machine. A job cannot be started on next machine unless it has completed its processing on earlier machine. In addition to that, no wait adds the constraint on FSSP that once the jobs get started on one machine it should complete its processing on every machine without any interruption or pre-emption. In order to satisfy this constraint, processing of first job is delayed at the start so that there must be no waiting time between processing of any consecutive operations of each of  $n$  jobs. So, a delay matrix (delmat) needs to be calculated as per stated in Equation 1 to solve NWFSS problem.

#### Nomenclature

$n$	Number of jobs
$m$	Number of machines
$PT_{ij}$	Processing time of $i^{\text{th}}$ job on $j^{\text{th}}$ machine
delmat	delay matrix
$\sigma(k-1, i')$	sequence order of jobs on $m$ machines $\{j_1, j_2, \dots, j_n\}$
$C_{\max}$	makespan

$$delmat = \max \left( \left( \sum_{i'=1}^i PT \sigma(k-1, i') - \sum_{i'=2}^i PT \sigma(k, i' - 1) \right) \mid 2 \leq k \leq n \right) \mid 1 \leq i \leq m \quad (1)$$

where,

$\sigma(k-1, i')$  is sequence order of job on  $m$  machines.

$delmat(i, k)$  is minimum delay time between start of job  $i$  and start of job  $k$  and job  $k$  follows job  $i$ . ( $1 \leq i \leq n$ ,  $1 \leq k \leq n$ ,  $i \neq k$ ). Delay matrix represents delay time between current job and next job to be submitted.

$delmat(j_{(i-1)}, j_i)$  denote minimum delay on the first machine between the start of two consecutive jobs  $i$  and  $i-1$ .

For the given matrix of size  $n(\text{jobs}) \times m(\text{machines})$  with processing time  $PT_{ij}$  generates  $(n!)$  number of feasible sequence of solutions from which optimal sequence is to be chosen. The problem is to determine a sequence of  $n$  jobs which gives minimum makespan. The formula makespan is given as per Equation 2.

$$C_{\max}(\sigma) = \sum_{k=2}^n delmat(\sigma(k-1), \sigma k) + \sum_{i=1}^m PT \sigma(n, i) \quad (2)$$

### 4. COMPUTATIONAL EXPERIENCE

The Metaheuristics under consideration are coded in Java and run on Intel Core i5, 8 GB RAM, 2.20 GHz PC. The experimentation is carried out in two phases. In first phase, small problem sizes with number of jobs ( $n$ )=6,8,10,12 and number of machines ( $m$ )=5,10,15,20,25 are considered. Second phase comprises of large problem sizes with  $n$ =20, 40, 60, 80,100 and  $m$ =5, 10,15,20,25. Each problem size in both phases have thirty independent problem instances. Each problem instance corresponds to a new processing time matrix ( $PT_{ij}$ ) generated randomly using uniform distribution  $u(1,99)$  which is used more often in research community[17]. The experimentation measures the performance by average relative percentage deviation (ARPD) for small problem size where deviation is calculated with respect to results obtained from metaheuristics and results obtained from enumerative approach. The performance for large problem size is measured with ARPD where deviation is calculated from results obtained from metaheuristic and result of best metaheuristic, which is metaheuristic giving less value of makespan at that problem instance. Equation 3 explains formula used for ARPD for small problem size.

$$ARPD = (100 / k) \sum_{i=1}^k \frac{\text{Metaheuristic-Optimal}}{\text{Optimal}} \quad (3)$$

Where, metaheuristic denotes the objective function value obtained for  $i^{\text{th}}$  instance by a Metaheuristics (PSO, GA and TS in this case). Optimal is the optimal solution value with enumerative approach and  $k$  is number of problem instances. Table 1 gives comparative analysis using ARPD values of metaheuristic algorithms with enumerative approach applied for small sized problem.

Table 2. Comparison of ARPD Values of Metaheuristic Algorithms with Enumerative Approach

n(jobs)	Problem Size		No. of instances	ARPD		
	m(machines)			PSO	TS	GA
6	5		30	6.50	27.09	11.33
	10		30	4.54	22.14	7.31
	15		30	4.72	22.35	7.22
	20		30	3.37	19.00	5.45
	25		30	4.21	17.66	5.76
8	5		30	10.08	29.11	11.10
	10		30	8.47	28.13	10.45
	15		30	8.92	29.00	11.36
	20		30	7.85	23.85	10.09
	25		30	7.05	23.27	7.79
10	5		30	14.30	32.03	13.19
	10		30	12.79	33.40	15.10
	15		30	14.07	31.99	13.42
	20		30	11.40	28.03	15.76
	25		30	10.71	25.99	12.91
12	5		30	15.65	34.02	19.52
	10		30	16.88	37.71	18.87
	15		30	16.62	36.95	19.71
	20		30	13.20	31.40	31.94
	25		30	14.66	31.37	16.89

Equation 4 provides formula for ARPD for large problem size.

$$\text{ARPD} = (100 / k) \sum_{i=1}^k \frac{\text{Metaheuristic-Best}}{\text{Best}} \quad (4)$$

Where, metaheuristic denotes the objective function value obtained for  $i^{\text{th}}$  instance by a Metaheuristics (PSO, GA and TS in this case). Best is optimal solution value obtained for that instance (PSO in majority of cases) and  $k$  is the number of problem instances for a problem size. Value of  $k$  is 30 for experimentation purpose. Table 2 gives comparative analysis of metaheuristic algorithms with best metaheuristic approach using ARPD values applied for large sized problem.

Clearly, the best possible performance is achieved when  $\text{ARPD} = 0$ . The “best” used in the ARPD calculations is better of the two competitors for the problem under consideration.

The results tabulated using ARPD metric in Table 1 shows that, in case of 600 problem instances, PSO outperforms TS and GA in 100% and 90% of the cases respectively. Table 2 reveals that, in case of 750 problem instances, PSO performs best than TS in 100% of the cases and better than GA in 80% of the cases. By observing the ARPD values tabulated in Table 1 and 2, it can be concluded that, PSO algorithm converges to near optimal solution as compared to GA and TS, which corroborates with the conclusions made by earlier researchers [20-22]. Programming efforts experience that PSO converges to optimal solution with less efforts of tuning parameters.

Table 3. Comparison of ARPD values of metaheuristic with best algorithm

n(jobs)	Problem Size		No. of instances	ARPD		
	m(machines)			PSO	TS	GA
20	5		30	0	11.63	2.12
	10		30	0	15.14	2.14
	15		30	0	16.02	2.09
	20		30	0	13.68	1.98
	25		30	0	12.52	1.54
40	5		30	0	9.99	0.91
	10		30	0	10.83	0.78
	15		30	0	11.30	1.07
	20		30	0	10.70	0.93
	25		30	0	9.87	1.22
60	5		30	0.46	8.52	0
	10		30	0	9.02	0.22
	15		30	0	9.68	0.89
	20		30	0	9.19	0.33
	25		30	0	8.25	0.01
80	5		30	0	7.33	0.15
	10		30	0	8.38	0.69
	15		30	0	8.31	0.32
	20		30	0	8.38	0.14
	25		30	0.07	7.65	0
100	5		30	0.57	6.36	0
	10		30	0	7.55	0.00
	15		30	0.03	7.18	0
	20		30	0	7.98	0.53
	25		30	0	7.06	0.27

## 5. CONCLUSION

In this paper, performances in terms of minimization of the makespan of NWFSS problem using three metaheuristic algorithms were examined. Empirical results on a large number of test problems showed that the solutions offered by use of PSO are found significantly better than GA and TS. Moreover, PSO converges more to the optimal solution with fairly less computation time and less overhead of parameter setting as compared with other metaheuristic techniques viz. GA and TS.

Also, relevant published literature is silent about the use of PSO as combinatorial algorithms for optimization in general and in case of no wait flow shop in particular. The results of comparative analysis presented in this paper advocates the said research gap of use of PSO as a future scope for developing a new variant of PSO to improve the solution quality in case of NWFSS.

## REFERENCES

- [1] Hall NG, Sriskandarajah C. "A survey of machine scheduling problems with blocking and no-wait in process", *Operation Research*, vol. 44, pp 510-525, 1996
- [2] Blum, C., Roli, "A. Metaheuristics in combinatorial optimization: Overview and conceptual comparison", *ACM Computing Surveys*, pp 268-308, 2003.
- [3] R. Hassan, B. Cohanin, O. Weck, G. Venter, "A Comparison of Particle Swarm Optimization and the Genetic Algorithm", *Proceedings of the structural dynamics and material conference*; vol. 46, 2005.
- [4] Jamian J. Mustafa M. Mokhlis H. Baharudin M. "Implimentation of Evolutionary Particle Swarm Optimization in Distributed Generation Sizing", *Int. J. of Electrical Engg. and Computer Sci*; Vol. 2, No. 1, pp-137-146. 2012.
- [5] Lu Zheng, Xingsheng Gu. "Fuzzy Production Scheduling in No-wait Flow shop to Minimize the makespan with E/T Constraints using SA", *Proceedings of the 5<sup>th</sup> World Congress on Intelligent Control and Automation*, vol 4, pp 2909-2913, 2004
- [6] W. Xia and Z. Wu. "A hybrid particle swarm optimization approach for the job-shop scheduling problem", *Int. J. Adv. Manuf. Technol.*, vol. 29 pp. 360-366, 2005.
- [7] B. Liu, L. Wang, and Y.H. Jin. "An effective hybrid particle swarm optimization for no-wait flow shop scheduling", *Int. J. Adv. Manuf. Technol.*, vol 31, pp 1001-1011, 2006.
- [8] Xiaoping Li, Qian Wang, Cheng Wu, Shih-Wei Lin, Kuo-Ching Ying, "An Efficient Method for No-Wait Flow Shop Scheduling to Minimize Makespan", *Proceedings of the 10th International Conference on Computer Supported Cooperative Work in Design*, pp. 1-6. 2006.



- [9] Q.K. Pan, M. Fatih Tasgetiren, and Y.C. Liang. "A discrete particle swarm optimization algorithm for the no-wait flowshop scheduling problem", *Comput. Oper. Res.* vol. 35 (9) pp. 2807–2839. 2008
- [10] D.Y. Sha and C.Y. Hsu. "A new particle swarm optimization for the open shop scheduling problem", *Comput. Oper. Res.*; vol. 35 (10) pp 3243–3261, 2008
- [11] Quan-Ke Pan, Ling Wang, M. Fatih Tasgetiren, Bao-Hua Zhao. "A hybrid discrete particle swarm optimization algorithm for the no-wait flow shop scheduling problem with makespan criterion", *Int. J. Advanced Manufacturing Technology*; vol. 38 pp. 337-347. 2008.
- [12] Quan-Ke Pan, Ling Wang, Bao-Hua Zhao, "An improved iterated greedy algorithm for the no-wait flow shop scheduling problem with makespan criterion", *Int. J. Advanced Manufacturing Technology*; vol. 38. Pp 778-786, 2008.
- [13] Dipak Laha and Uday K. Chakraborty, "A constructive heuristic for minimizing makespan in no-wait flow shop scheduling", *Int. J. Advanced Manufacturing Technolog.*, vol.; 44 pp. 97–109, 2009.
- [14] I.H. Kuo, S.J. Horng, T.W. Kao, T.L. Lin, C.L. Lee, T. Terano, and Y. Pan, "An efficient flow-shop scheduling algorithm based on a hybrid particle swarm optimization model", *Expert Syst. Appl.* vol. 36 (3) pp. 7027–7032, 2009.
- [15] Kaizhou Gao, Shengxian Xie, Hua Jiang, Junqing Li. "Discrete Harmony Search Algorithm for the No Wait Flow Shop Scheduling Problem with Makespan Criterion", LNCS 6838 ICIC, pp. 592–599, 2011.
- [16] G. Vijay chakaravarthy, S. Marimuthu, and a. Naveen Sait, "Performance evaluation of proposed Differential Evolution and Particle Swarm Optimization algorithms for scheduling m-machine flow shops with lot streaming", *J. Intell. Manuf.* vol. 24 (1), pp. 175–191. 2011.
- [17] Donald D., Ivan Z, Magdalena B, Davendra, Roman S, Roman J. "Discrete Self-Organizing Migrating Algorithm for flow-shop scheduling with no-wait makespan", *Mathematical and Computer Modeling*, vol. 57 pp. 100–110, 2013.
- [18] X. Shao, W. Liu, Q. Liu, and C. Zhang, "Hybrid discrete particle swarm optimization for multi-objective flexible job-shop scheduling problem", *Int. J. Adv. Manuf. Technol.*, vol. 67. Pp 2885–2901. 2013.
- [19] S.U. Sapkal and Dipak Laha, "An Efficient Heuristic Algorithm for m-Machine No-Wait Flow Shops", *Proceedings of the International Multiconference of Engineers and Computer scientist*; vol. 1. 2011.
- [20] Nasser A. Mahmoud S. Horbaty M. "A Comparative Study of Meta-heuristic Algorithms for Solving Quadratic assignment Problem", *Int. J. of Advanced Computer Science and Applications*, Vol. 5, No. 1, pp-1-6. 2014.
- [21] Arora T. Gigras Y. "A Survey of Comparison between Various Metaheuristic Techniques for Path Planning Problem", *Int. J. of Computer Engineering & Science*, Vol. 3 No. 2, pp 62-66. 2013.
- [22] Adhani G, Buono A, Faqih A. "Optimization of Support Vector Regression using Genetic Algorithm and Particle Swarm Optimization for Rainfall Prediction in Dry Season", *Int. J. of Electrical Engg. and Computer Sci.* Vol. 12, No. 11, pp. 7212-7219. 2014

## BIOGRAPHIES OF AUTHORS



**Mrs. Laxmi A. Bewoor** is research scholar at K.L. University, Guntur. She is working as Assistant professor at Vishwakarma Inst.of Info. Tech. from last 9 years. She has total 14 years of teaching experience. Her research interest includes Algorithmic Analysis and Artificial Intelligence.



**Dr. V. Chandra Prakash** is working as a Professor at K.L. University, Guntur. His research interest includes Artificial Intelligence and Data Mining.



**Dr. Sagar U. Sapkal** is working as a Associate Professor at Walchand college of Engineering, Sangali. His research interst includes optimization and manufacturing.