

# IMPROVED GRID SCHEDULING USING HYBRID IWD-ABC ALGORITHM

**Dr. D. Thilagavathi**

Associate Professor and Head, Department of Computer Technology,  
Nallamuthu Gounder Mahalingam College, Pollachi

Received: March 01, 2019

Accepted: April 05, 2019

**ABSTRACT:** *Grid computing is a high performance computing technique that solves a large problem faster by using many systems that are tightly coupled by software and networks working together. Scheduling of jobs which is a process of finding out optimal resource for the incoming jobs and to execute those jobs in the corresponding resource in a Grid environment is a great challenge. IWD (Intelligent Water Drop) and ABC (Artificial Bee Colony), a nature-inspired approaches is employed in the proposed work to attain an optimal solution for the Job Scheduling Problem in Computational Grid. The primary objective is to decrease the makespan of the submitted jobs and to ensure higher resource utilization rate of the Grid environment. In this paper, the proposed scheduling algorithm IWDEABC using GridSim simulator tool with appropriate Grid Workloads archive is implemented and analyzed.*

**Key Words:** *Grid Computing, Scheduling, Intelligent Water Drop, Artificial Bee Colony, Makespan*

## I. Introduction

Geographically dispersed computers connected by web in a Grid-like manner has been used to create virtual environment with supercomputer having high computing capacity to solve complex problems from e-Science in no matter of time [Foster, 2002]. Job Scheduling process in Grid [Jennifer Schopf, 2004] involves three main phases: Resource Discovery that generates a list of potential resources, Information Gathering about those resources and selection of a best set and Job Execution that includes file staging and cleanup. The goal of Grid scheduling is to make an optimal mapping of tasks to computational resources and utilize them efficiently with the objective(s) of minimizing makespan, flowtime, cost and maximizing load balancing across resources, resource utilization, throughput etc [Thilagavathi et al., 2012].

SI, a nature inspired algorithms like Particle Swarm Optimization, Ant Colony Optimization, ABC, Bacterial Foraging Optimization Algorithm, Firefly Algorithm, Artificial Immune System, Fish Swarm Algorithm, IWD, Shuffled Frog Leaping Algorithm, Group Search Optimizer, etc., are metaheuristics that mimics the nature for solving optimization problems opening a new era in computation. ABC is an approach based on honey bee swarm for solving multidimensional and multimodal optimization problems. Due to its encouraging results in solving TSP and classification problems it is obvious that ABC is being used by many researchers in Job Scheduling in Grid. Shah-Hosseini, 2007 introduced the IWD algorithm for the first time. It was an inspiration from the movement of natural water drops which flow in rivers, lakes and seas. It is a population-based metaheuristic where the IWDs construct a better solution through cooperation with each other. The capability of IWD algorithm is to deal with optimization problems in finding solutions with good or optimal qualities.

The proposed IWDEABC improves the optimal schedule solution as EABC integrated in it accounts for faster emergence of path to their present location from past trajectory continuously instead of backtracking their original route that soar up the results of IWD algorithm. Makespan and average resource utilization rate are the performance metrics used in this paper to evaluate the efficiency of the proposed algorithm. The proposed algorithm has been tested using GWA-T-1 DAS2 workload trace in GridSim Simulation toolkit and the simulation results show that the proposed IWDEABC algorithm produces minimal makespan time with average resource utilization rate gets increased.

## II. Related Works

Researchers have provided solutions to scheduling of jobs in Grid from diverse perspective. SI techniques are increasingly being used for solving optimization problems that have proven themselves as a good candidate for scheduling where the aim is to maximize the resource utilization of the Grid. The pros of these techniques stems from their capability in searching large search spaces, that arises due to many combinatorial optimization problems, very efficiently [Grosan et al., 2007]. This can be proven by recent research in this area.

Job-shop scheduling is a typical NP-hard problem which has drawn continuous attention from researchers. Niu et al., 2012 customized the IWD algorithm, for solving job-shop scheduling problems. Five schemes are used to improve the original IWD algorithm and the improved algorithm is named the EIWD algorithm. The optimization objective is the makespan of the schedule. Experimental results show that the EIWD algorithm is able to find better solutions for the standard benchmark instances than the existing algorithms. The results tried to increase the diversity of the solution space as well as the solution quality.

The Niu et al., 2013 proposed a modified version of the original IWD algorithm to solve the Multi-objective job shop scheduling (MOJSS) problem. The optimization objective is to find the best compromising solutions considering multiple criteria, namely makespan, tardiness and mean flow time of the schedules. A scoring function which gives each schedule a score based on its multiple criterion values that is embedded into the MOJSS-IWD's local search process. Experimental evaluation shows that the customized IWD algorithm can identify the Pareto non-dominance schedules efficiently.

Arsuaga-Rios et al., 2011 presented a multi-objective swarm optimization algorithm for scheduling experiments across the Grid. The obtained result shows not only the Multi-Objective Artificial Bee Colony (MOABC) algorithm is reliable taking into account the standard deviation, but also that its results dominate the results obtained by other Grid schedulers. Vivekanandan et al., 2011 implements the ABC algorithm for Job Scheduling in Grid. With the implementation of this approach, the ABC algorithm reaches optimal solution as well as obtains the better QoS than ACO. Kim et al., 2013 proposed a Binary Artificial Bee Colony (BABC) algorithm for binary integer Job Scheduling Problems in Grid computing. Further the author proposed an efficient binary artificial bee colony (EBABC) extension of BABC that incorporates a Flexible Ranking Strategy (FRS) to improve the balance between exploration and exploitation. Simulation results for benchmark Job Scheduling Problems show that the performance of proposed methods is better than those alternatives such as GA, SA and PSO.

Thilagavathi et al., 2015 proposed algorithm named EIWD-TS, a meta-heuristic algorithm based on swarm intelligence to find the near optimal solution considering multiple objectives namely makespan, slowdown ratio, failure rate and resource utilization of grid scheduling. The result of the proposed model is tested with PSA (Parameter Sweep Application) dataset and the results are compared with Risky-MinMin (RMM), Preemptive-MinMin (PMM), PSO and IWD. Experimental evaluation shows that the EIWD-TS algorithm has good convergence property and better in quality of solution than other algorithms reported in recent literature. Thilagavathi et al., 2015 proposed a hybrid approach of IWD algorithm with a PSO algorithm for the job scheduling problem in Grid Computing. The proposed algorithm was evaluated and tested on NAS (Numerical Aerodynamic Simulation) data set showing promising results.

### III. Problem statement

Computational Grid is a large scale distributed system consisting of a huge number of heterogeneous resources that belong to different administrative domains. Scheduling jobs in such high performance computing environment represents a great challenge. Let there be  $n$  jobs submitted to a Grid scheduler and let  $m$  resources be the set of resources available at the time of job arrival then the problem considered is scheduling of jobs. It is a process of finding out optimal resource for the incoming jobs and to execute those jobs in the corresponding resource.

### IV. Proposed iwdeabc method for job scheduling in grid

The enhancement in ABC and its hybridization with IWD improves the exploration and exploitation capability of the proposed algorithm and also enhance the convergence ability of the algorithm. Pseudo code of the proposed algorithm IWDEABC is explained as in Algorithm-1. The flowchart of the proposed IWDEABC algorithm has been presented in Figure-1.

**Input:** Grid workload traces, IWD and ABC parameters

**Output:** Makespan and ARU rate of near optimal schedule solution

**Method:**

**Step 1:** Initialization of food sources, onlooker bees and scout bees, initialization of *IWD* parameters, initialize  $itr=1$ .

**Step 2:** Put all *IWDs* on the first node

**Step 3:** For each *IWD* construct a schedule

**Step 3.1:** Update the velocity of the *IWD*

**Step 3.2:** Select an edge to reach to the next node

**Step 3.3:** Compute the amount of soil ( $\Delta soil$ ) which is gathered by the *IWD*

**Step 3.4:** Update the edge soil and the *IWD* soil  
**Step 3.5:** If all *IWDs* completed their solution goto step 4 else goto step 3.2  
**Step 4:** Each *IWD* produce schedule  
**Step 5:** Find the elitist *IWDs*  
**Step 6:** Perform the local search on *IWDs* using employed bee phase  
**Step 7:** Update the global best solution  
**Step 8:** Calculate Probabilities for onlooker bees  
**Step 9:** Onlooker bees Phase  
**Step 10:** Scout bees Phase  
**Step 10:** Memorize the best solution  
**Step 11:** If ( $itr < max\_itr$ ) then  
     Return the optimal schedule with minimum makespan  
 else  
    $itr = itr + 1$   
   goto the step 2.

---

#### Algorithm-1. Proposed IWDEABC ALGORITHM

In the proposed IWDEABC algorithm, the initialization of *IWD* parameters and initialization of EABC parameters are done. After each *IWD* constructs a schedule, the local search on selection of best *IWDs* are done by employed bee phase of EABC algorithm. The global best solution is updated. Onlooker bee phase depending upon the calculated probability chooses the solution which improves the exploration and exploitation capability of the schedule solution. Scout bee phase of EABC algorithm is then invoked to further improve the solution. Finally the best solution is memorized. If iterations reach the desired maximum number of iterations mentioned, the algorithm stops by returning to the near optimal schedule with a minimum makespan.

## V. results and discussions

GridSim which is a Java-based Grid simulation toolkit is used for simulation in the Grid environment. It is designed to evaluate resource management and scheduling strategies for large scale distributed computing [Rajkumar Buyya and Manzur Murshed, 2002]. It is used to evaluate the performance of the proposed algorithm IWDEABC. The recital of proposed algorithm is tested using GWA-T-1 DAS2 workload.

### 5.1 GWA-T-1 DAS2

The GWA-T-1 DAS2 trace file [[gwa.ewi.tudelft.nl/datasets/gwa-t-1-das2](http://gwa.ewi.tudelft.nl/datasets/gwa-t-1-das2)] has a total of 1124772 jobs. Among that 384787 are sequential jobs and 739985 are parallel jobs. The traces collected from the DAS include applications from the areas of physics, robotics, graphics, CAS, AI, math, chemistry, climate, etc. In addition, the DAS traces include experimental applications for parallel and distributed systems research.

DAS2 is a Grid environment composed of five clusters. These clusters are located at five different universities in the Netherlands. The cluster located at Vrije University, Amsterdam has 144 processors and rest of four have 64 processors. Each cluster has 20GP data saving capacity per processor. There is no failure trace available and the workload trace contains no specific job requirements. The 400 nodes of the DAS2 log are mapped to 5 Grid sites, 4 of the sites each contain 64 nodes and the one site located at Vrije University Amsterdam contain 144 nodes.

### 5.2 Performance Metrics

Performance metrics helps the researchers to prove the competence of their algorithm and to measure their significance of work thereby unmeasured work can be minimized or eliminated. The following section discuss about the metrics used to measure the performance and to compare with TS, *IWD*, PSO, ABC and EABC [Thilagavathi et al., 2014].

### 5.3 Makespan

In the proposed work, the scheduling optimization parameters used is makespan. It is a measure of efficiency and throughput of a Grid computing system [Song et al., 2005; Xhafa and Carretero, 2012; Xie et al., 2007].

Let  $R = \{r_1, r_2, \dots, r_m\}$  be  $m$  Grid resources and  $J = \{j_1, j_2, \dots, j_n\}$  be  $n$  independent client jobs.

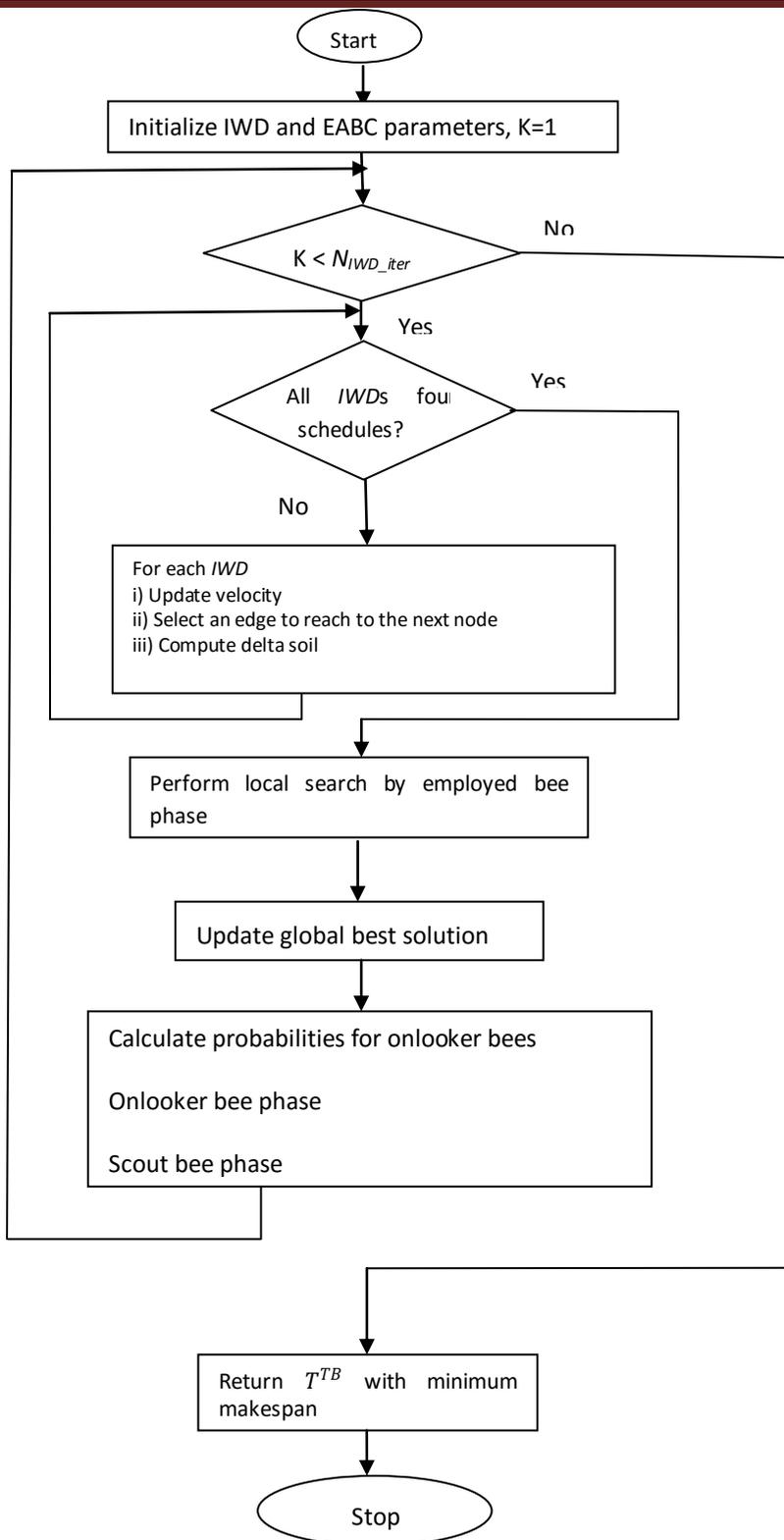


Fig. 1. Flowchart of the proposed IWDEABC ALGORITHM

Average Resource Utilization Rate

Define  $C_{ij}$  as the time that resource  $r_i$  needs to finish job  $j_i$ ;  $\sum C_i$  is the total time that resource  $r_i$  completes all the jobs submitted to it. Then the function described in Eq.5.1 is makespan time.

$$f_{max}(C) := \max\{\sum C_i\} \tag{5.1}$$

Then the makespan optimization problem is to minimize the  $Eq(1)$ .

The main objective of Grid computing systems is to maximize resource utilization. This metric improves the utilization of resources by minimizing the idle time of resources. It is defined as the percentage of time that resource  $R_j$  is busy during the scheduling process [Rafsanjani and Bardsiri, 2012]. ARU is calculated using the formula in Eq 5.2

$$ARU = \frac{\sum_{j=1}^m ru_j}{Number\ of\ resources} \times 100 \tag{5.2}$$

Where  $ru_j$  is calculated by using Eq.5.3

$$ru_j = \frac{\sum_{i=1}^n (t_{ei} - t_{si})}{TQRU} \tag{5.3}$$

Where  $TQRU$  = Total Quantity Resource Used

$$TQRU = \sum_{i=1}^n CT$$

$CT$  = Completion Time of Task

$ru_j$  = Resource Utilization of resource  $j$

$t_{ei}$  = End Time of Task  $i$

$t_{si}$  = Start Time of Task  $i$

### 5.4 Simulation Parameters

The simulation parameters measure the performance of the proposed metaheuristic scheduling algorithm and are shown in Table-1 for GWA-T-1 DAS2.

Table 5.1: Simulation Parameters of GWA-T-1 DAS2 Trace

PARAMETER	VALUE SETTING
Number of jobs $N$	36000
Number of sites $M$	5
Job arrival rate	Given by trace
Job workloads	Given by trace
Site processing speed	1x144 nodes and 4x64 nodes

In order to compare the performance of the proposed IWDEABC algorithm, TS, IWD, PSO, ABC and EABC algorithms are chosen, which are stochastic in nature. Figure-2 shows the makespan of the job sizes 250 to 2000 for GWA-T-1 DAS2. EABC works better than PSO and ABC algorithms but poor comparing to TS, IWD and IWDEABC algorithm. The results prove that the IWDEABC algorithm provides better makespan results than all the existing algorithms taken in the experiment.

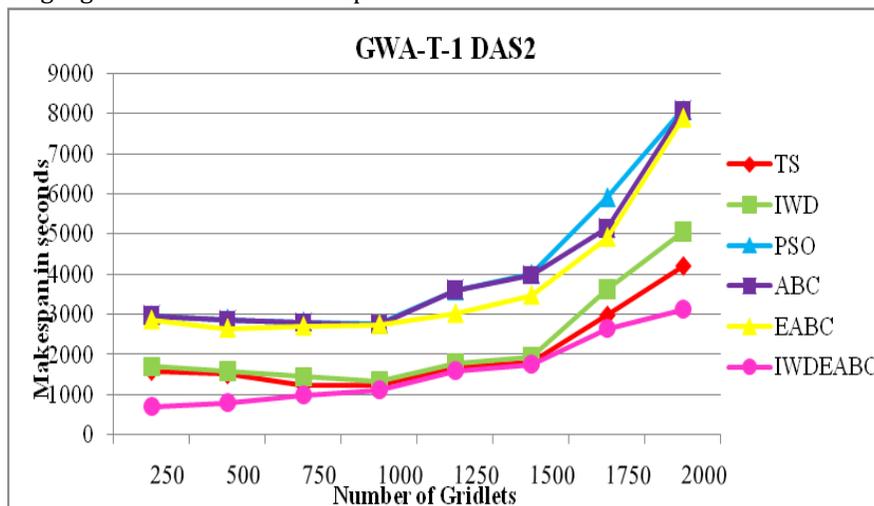


Fig-2. Makespan Analysis of IWDEABC for lightweight job sizes in GWA-T-1 DAS2

Figure-3 illustrates the makespan of 4000 to 36000 different heavy job sizes in GWA-T-1 DAS2. It is observed that the IWDEABC performs better for heavy job sizes too. TS perform better than PSO, ABC, EABC and IWD for smaller job sizes in terms of makespan values but as the job sizes gets increased it shows worst results.

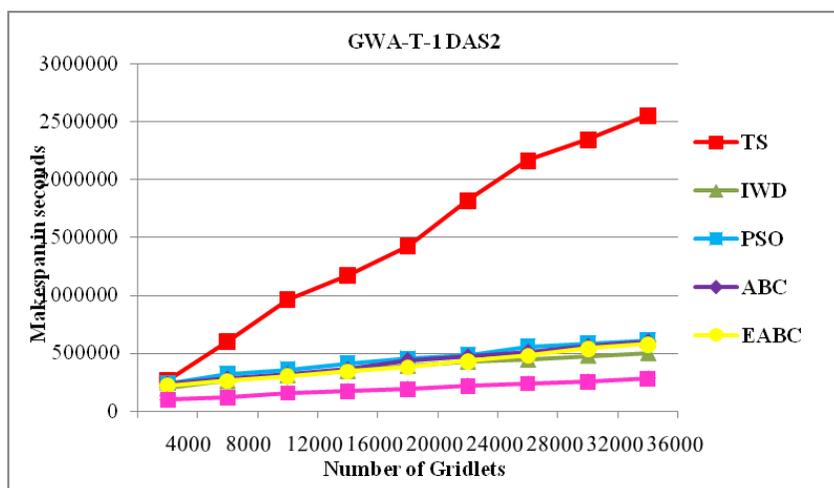


Fig-3. Makespan analysis of IWDEABC for heavy job sizes in GWA-T-1 DAS2

Figure-4 provides ARU rate in % of the proposed IWDEABC for GWA-T-1 DAS2 traces used. It can easily found out from the experimental results that the proposed IWDEABC algorithm can achieve good makespan in any situation by taking less time to execute jobs and also achieves 10.7% improvement in ARU rate using GWA-T-1 DAS2 in contrast to IWD algorithm. IWDEABC recorded the minimum makespan and the maximum ARU rate for the total simulated experiments in the Grid. It means that IWDEABC can handle different size jobs in Grid and thereby provides effective throughput by reducing machine idle time.

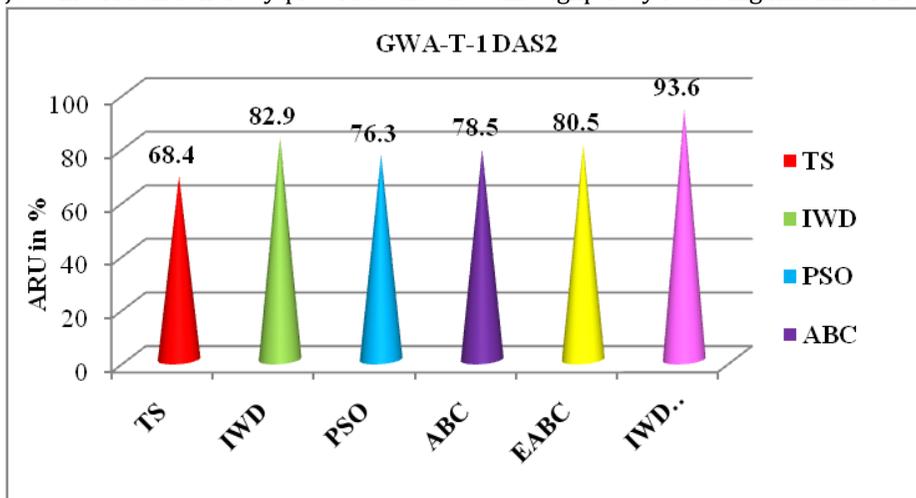


Fig-4. ARU rate analysis for IWDEABC in GWA-T-1 DAS2

**vi. conclusion**

The proposed IWDEABC scheduling algorithm reports minimum makespan with the maximum ARU rate showing prospective results. This research work proves that there is a tremendous potential for extending the original IWD in future. The next step would be to apply and incorporate the algorithms in a real world Grid. The research is carried out for single objective Job Scheduling that is the primary criteria makespan is idealized. But as an enrichment of this work multi-objective Job Scheduling metaheuristic algorithms can be developed by considering optimization criteria like tardiness, flowtime, etc.

**References**

1. Foster, I. (2002), What Is The Grid? A Three Point Checklist, Grid Today, Vol. 1, No. 6.
2. Jennifer Schopf (2004), Ten actions when Grid Scheduling: the user as a Grid Scheduler, Grid resource management, Kluwer Academic Publishers Norwell, MA, USA, pp.15-23

3. Thilagavathi, D., and Antony Selvadoss Thanamani (2012), A Survey on Dynamic Job Scheduling in Grid Environment Based on Heuristic Algorithms, *International Journal of Computer Trends and Technology*, ISSN: 2231 – 2803, Vol. 3, No. 4, pp. 531-536.
4. Shah-Hosseini (2007), Problem Solving by Intelligent Water Drops, in: *IEEE Congress on Evolutionary Computation*, Swisshotel The Stamford, Singapore, pp. 3226–3231.
5. Grosan, C., Abraham, A., and Helvik, B. (2007), Multiobjective Evolutionary Algorithms for Scheduling Jobs on Computational Grids, *ADIS International Conference, Applied Computing*, Salamanca, Spain, Nuno Guimare and Pedro Isaias (Eds)..
6. Niu, S. H., Ong, S. K., and Andrew YC Nee (2012), An Improved Intelligent Water Drops Algorithm for Achieving Optimal Job-Shop Scheduling Solutions, *International Journal of Production Research*, Vol. 50, No. 15, pp. 4192
7. Niu, S. H., Ong, S. K., and Andrew YC Nee (2013), An Improved Intelligent Water Drops Algorithm for Solving Multi-Objective Job Shop Scheduling, *Engineering Applications of Artificial Intelligence*, Vol. 26, No. 10, pp. 2431-2442.
8. Arsuaga-Rios, M., Vega-Rodriguez, M., A., Prieto-Castrillo, F. (2011), Evaluation of Multiobjective Swarm Algorithms for Grid Scheduling, *11th International Conference on Intelligent Systems Design and Applications*, Vol. 22, No. 24, pp. 1104-1109.
9. Vivekanandan, K, Ramyachitra, D., and Anbu, B. (2011), Artificial Bee Colony Algorithm for Grid Scheduling, *Journal of Convergence Information Technology*, Vol. 6, No. 7, pp. 328-339.
10. Kim, Sung-Soo, Ji-Hwan Byeon, Hongbo Liu, Ajith Abraham and Sean McLoone (2013), Optimal Job Scheduling in Grid Computing Using Efficient Binary Artificial Bee Colony Optimization, *Journal of Soft Computing*, Vol. 17, No. 5, pp. 867-882.
11. Thilagavathi, D., and Antony Selvadoss Thanamani (2015), Intelligent Water Drop Algorithm Powered by Tabu Search to Achieve Near Optimal Solution for Grid Scheduling, *ARNP Journal of Engineering and Applied Sciences*, ISSN 1819-6608, Vol. 10, No. 8, pp. 3673-3678 [ISI Science Citation Index, Scopus indexed Journal].
12. Thilagavathi, D., and Antony Selvadoss Thanamani (2015), Intelligent Water Drop Algorithm based Particle Swarm Optimization (IWDPSO) towards Multi-objective Job Scheduling for Grid Computing, *Research Journal of Applied Sciences, Engineering and Technology*, © Maxwell Scientific Organization, ISSN: 2040-7459; e-ISSN: 2040-7467, Vol. 9, No. 11, pp. 982-989.
13. Thilagavathi, D., and Antony Selvadoss Thanamani (2015), An Improved Intelligent Water Drop Algorithm for Job Scheduling in Grid Computing, *International Journal of Applied Engineering Research*, ISSN 0973-4562, Vol. 10, No. 2, pp. 19979-19986
14. Rajkumar Buyya and Manzur Murshed (2002), Gridsim: A Toolkit for the Modeling and Simulation of Distributed Resource Management and Scheduling for Grid Computing, *The Journal of Concurrency and Computation: Practice and Experience*, Vol. 14, 1175-1220.
15. [gwa.ewi.tudelft.nl/datasets/gwa-t-1-das2](http://gwa.ewi.tudelft.nl/datasets/gwa-t-1-das2)
16. Thilagavathi, D., and Antony Selvadoss Thanamani (2014), An Impression on Performance Metrics for Scheduling Problem in Grid Computing Environment, *International Journal of Research in Computer Applications and Robotics*, ISSN: 2320-7345, Vol. 2, No. 8, pp. 40-45.
17. Song, S., Yu-Kwong Kwok and Kai Hwang (2005), Security-Driven Heuristics and A Fast Genetic Algorithm For Trusted Grid Job Scheduling, in: *Proceedings 19th International IEEE Symposium Parallel and Distributed Processing*, pp. 65a.
18. Xhafa and Carretero (2012), A Tabu Search Algorithm for Scheduling Independent Jobs in Computational Grids, *Computing and informatics*, Vol. 28, pp. 1001-1014.
19. Xie, Tao and Xiao Qin (2007), Performance Evaluation of a New Scheduling Algorithm for Distributed Systems with Security Heterogeneity, *Journal of Parallel and Distributed Computing*, Vol. 67, No.10, pp. 1067-1081.
20. Rafsanjani, M. K., and Bardsiri (2012), A New Heuristic Approach Based On Load Balancing for Grid Scheduling Problem, *Journal of Convergence Information Technology*, Vol. 7, No. 1, pp. 329-336.

## Bibliography

Dr. D. Thilagavathi presently holds a PhD in Computer Science by the Bharathiar University, Coimbatore, India and is Associate Professor and Head for the Department of Computer Technology, Nallamuthu Gounder Mahalingam College, Pollachi, Coimbatore. She has to her credit 18 years of teaching experience. Her main area of interest is Object oriented analysis and design, Grid Computing, Big Data. She has presented and published more than 20 papers in National/International conferences and journals.