TIME AND RELIABILITY OPTIMIZATION BAT ALGORITHM FOR SCHEDULING WORKFLOW IN CLOUD

Prachi Chaturvedi[1], Abhishek Satyarthi[2], Sanjiv Sharma[3]

[123] *CSE/IT Dept., Madhav Institute of Technology & Science, RGPV, Gwalior, India*

Abstract - Effective scheduling is one of the key concerns while executing workflows in the cloud environment. Workflow scheduling in clouds refers to the mapping of workflow tasks to the cloud resources to optimize some objective function. In this paper, we apply a recently developed meta-heuristic method called the BAT algorithm to solve the multi-objective problem of workflow scheduling in clouds that minimizes the execution time and maximizes the reliability by keeping the budget within user specified limit. Comparison of the results is made with basic, randomized, evolutionary algorithm (BREA) that uses greedy approach to allocate resources to the workflow tasks on the basis of low cost, high reliability and improved execution time machines. It is clear from the experimental results that the BAT algorithm performs better than the basic randomized evolutionary algorithm.

Keywords— *Cloud Computing, BAT Algorithm, Workflow Scheduling Algorithm.*

I. INTRODUCTION

Cloud computing is emerging as a popular and widely used platform in almost every field. Ambrust et al.[1] define cloud computing as applications which are delivered over the Internet in the form of services, as well as the hardware and software systems in the data center that actually provide these services. Buyya et al. [2] expresses cloud computing on the basis of its utility-oriented aspect. Cloud computing also offers several features like pay-per-use, broad network access, virtualization, rapid elasticity and on-demand access. These features make cloud environment highly suitable for handling the scientific workflow applications.

Workflow is a series of related tasks connected by control-flow and data-flow dependencies. To run workflows on cloud, high-level approach is to make a virtual cluster, formed of cloud resources, and then schedule the tasks of the workflow on that cluster. Scheduling in clouds is an NP-Hard problem. Many scheduling methods which include heuristic as well as meta-heuristic techniques have been applied for workflow scheduling in clouds and more are constantly being experimented and researched, still a lot of work needs to be done for making scheduling in clouds more effective. The quality of the output also depends on whether the tasks are scheduled on reliable virtual machines or not because virtual machines having less failure rate are expected to successfully execute tasks allocated to them. However, scheduling a task on more reliable machine incurs more cost. Similarly, time and cost are conflicting parameters that also needs to be balanced and optimized. So, in order to get optimal solution, the tradeoff among these objectives must be effectively managed.

Unlike most of the scheduling algorithms that assign cloud resources to the input tasks based on single objective function, usually cost or execution time, in this paper, we deal with multi-objective scheduling problem where we link three objectives namely cost, execution time and reliability into a single objective function. The aim of our paper is to experiment a recently developed meta-heuristic approach called the BAT algorithm for the purpose of workflow scheduling in clouds with the objective of maximizing reliability and minimizing execution time ensuring that the cost does not exceed the user specified budget.

II. RELATED WORK

There are several algorithms designed for scheduling workflow in cloud. The algorithms are discussed below. A market oriented hierarchical scheduling strategy in cloud workflow system proposed by Wu, Zhangjun, et al. [3] is a new approach which takes into account both the aspects of workflow scheduling namely service level scheduling and task level scheduling. Here a hierarchical methodology is followed in which at the first level i.e. service level scheduling is done by using a package based random scheduling algorithm and the task level scheduling which maps the tasks to VMs is done using three prominent meta heuristic approaches namely Ant colony optimization (ACO), Genetic algorithm (GA) and Particle Swarm Optimization (PSO). All the three approaches are compared and Ant Colony optimization is found to be best according to the results. The comparison was done considering three basic parameters at the task level which eventually affects the Quality of Service (QoS) namely Number of CPU cycles, Makespan and Cost. The algorithm has been implemented using SwinDev-C [4] which is specifically designed for cloud.

Suraj Pandey et al [5] have proposed a particle swarm optimization based algorithm for scheduling workflow applications. The particle swarm optimization [6] technique was proposed by Eberhart and Kennedy in 1995 and is considered to be one of the best swarm intelligence techniques. It gives a better convergence with a few runs. This is because it adjusts individual value based on local best and the global best which is the key for optimization. The proposed algorithm mainly compares the costs required for mapping tasks in workflow by the proposed algorithm (PSO) and to total cost occurred when using Best resource Selection. According to their results the PSO based algorithm is three times more cost effective than Best Resource Selection algorithm. The implementation is done using Amazon EC2.

Similar to the above, Wu, Zhangjun, et al have used revised discrete particle swarm optimization for scheduling workflow in cloud [7]. This algorithm is similar to Particle Swarm Optimization but with a set based concept embedded in it. Here authors use GRASP (Greedy randomized adaptive search procedure) to initialize the swarm. The proposed algorithms results are compared with Particle Swarm Optimization and Best Resource Select algorithms results based on makespan and cost optimization ratio. These results and statistics are obtained by varying computational cost and communication cost. According to the results RDPSO proves to be better. These algorithms have been implemented using Amazon elastic cloud compute.

The algorithms discussed till now consider only single workflow execution but there is another algorithm which considers scientific workflow ensemble [8]. Workflow ensemble is a collection of several interrelated workflows grouped together. Malawski, Maciej, et al. have proposed a method for cost and deadline constrained provisioning for scientific workflow ensemble in IaaS. Based on static and dynamic strategies the authors have considered three algorithms. The algorithms are namely Dynamic Provisioning and Dynamic Scheduling (DPDS), Workflow aware DPDS and Static Provisioning Static Scheduling. The first two algorithms are dynamic in nature i.e. these algorithms do resource provisioning and scheduling at run time but SPSS creates the resource provisioning and schedule plan before runtime. The implementation is done by using CloudSim [9].

Similarly, Abrishami, Saeid, Mahmoud Naghibzadeh, and Dick HJ Epema [10] have used same Partial critical path for scheduling workflow in IaaS. The properties of SaaS and IaaS are different and same approach cannot be followed. So based on the properties of IaaS the authors have proposed two approaches for solving the problem of workflow scheduling in IaaS. Maintaining QoS is still the main goal. This first approach named IC-PCP is one phase algorithm which uses two methods namely parent Schedule and path Schedule repetitively used to schedule all the nodes. Here parent Schedule schedules all the unscheduled predecessors given the scheduled node as input and path Schedule schedules all the tasks into a single instance of computational service with minimum price and considering the latest finish time. Similarly another approach called IaaS Cloud Partial Critical Paths with Deadline Distribution (IC-PCPD2).It is two phase approach with two phases namely Deadline Distribution and planning phase. Both of these phases uses parent Schedule and Path Schedule algorithm with some minor changes in it. These two approaches are compared for their efficiency and it is found that the IC-PCP performs better in several cases [11].

There is another algorithm that is proposed by Lin et al [12] which proposes solution for scheduling workflow in elastic cloud. There are two phases involved in workflow scheduling these are: task prioritizing phase and resource allocation phase. In the task prioritizing phase tasks are selected by using the proposed priority based algorithm. In the second phase for resource allocation a new algorithm by name Scalable Heterogeneous Earliest Finish Time is proposed. This algorithm uses several parameters such as Earliest Start Time, Earliest Finish Time, Earliest Ready time and Task Finish time. This algorithm is extension of Heterogeneous Earliest Finish time (HEFT) and is specifically proposed for elastic cloud.

There is another algorithm proposed by Genez, Thiago AL, Luiz F. Bittencourt, and Edmundo RM Madeira [13] for Scheduling Workflow cloud specifically SaaS workflows into Components of IaaS i.e. mapping the SaaS workflows which are used by Customer and the IaaS components of multiple IaaS which are present in the datacenter. This method involves two levels of SLA's. Integer Linear Program is formulated for solving the scheduling problem and they have also proposed two heuristics to obtain integer solution from the relaxed version of ILP. Both the algorithms are implemented and it is found that ILP gives a low cost solution in very less time and thereby achieving the goal of proposing an optimized scheduling process.

Above all there is another paper by which checks the efficiency with which scientific workflows runs in cloud. This paper in short checks the suitability of cloud to run scientific workflows. All the experiments are done based on the astronomical tool named montage [14]. The montage workflows are tested in three cloud environments namely local environment, virtual environment and wide area environment. According to their results it is found that the local environment is better for small workflows. But there is a scalability issue which is solved by using virtual environment. But on the whole, the wide area network (public) is found to be the best option for scheduling scientific workflow in cloud.

In this paper the authors have proposed a new algorithm by name iterative ordinal optimization. This method was designed to reduce the scheduling overhead and to reduce the time taken by the algorithm to schedule a complex task. The authors have compared this methodology with Monte Carlo method and blind pick methods. The algorithm proves to be better than them. The authors have done a detailed study and have demonstrated the working of scientific workflows in different clouds. According to the results the different types of clouds doesn't have any specific significant differences. They also test scientific workflow manager pegasus [15] over different cloud infrastructure and come to the conclusion that sky computing supports scientific workflow scheduling.

A lot of work has already been done in the field of workflow scheduling in clouds using different meta-heuristic methods. In our work, we chose the BAT algorithm for scheduling tasks to cloud resources because the BAT algorithm has shown good results when applied to different optimization problems14-17 and exhibits features like frequency tuning, parameter control and automatic zooming into the region where the optimal solution lie. So we are motivated to use this algorithm in the field of workflow scheduling in clouds. Not only this, most of the work done in this area generally take cost and time as the objectives to be optimized. Reliability is also an important factor along with cost and time since scheduling tasks on low reliability machines can result in high failure rate. The algorithm presented in the paper works to optimize the execution time considering the user defined budget constraint and also ensures that high reliability virtual machines are assigned to the tasks.

III. PROPOSED WORK

3.1 The BAT Algorithm

In this paper we develop a workflow scheduling set of rules based totally on a heuristic method ,For scheduling workflow responsibilities on cloud assets, a new meta-heuristic method called the BAT algorithm is carried out. It is referred to as so due to the fact its working is primarily based on the echolocation conduct of the digital bats17-19.

that it associates with the objective function which is to be optimized. Bat algorithm is designed by Xin-She Yang [16]. This algorithm is a metaheuristic algorithm which is designed for optimization purposes. This is one of the recently developed technologies which was primarily aimed at solving engineering problems. Metaheuristic algorithms are known for optimization. There are several other metaheuristic algorithms that are designed for optimization. The most basic problem of optimization is maximization or minimization of a function. Many metaheuristic algorithms concentrate on this aspect of optimization. Usually a good algorithm is characterized by the capability of maximizing or minimizing more accurately. Basically optimization is used for the problems which do not have exhaustive solution. Thus the best optimality involves finding out the best solution (Optimal solution), usually by means of randomization.

Similarly bat algorithm uses randomization for calculating the optimal value. To prove the superiority of an algorithm, an algorithm is tested with several standard functions available such as Rastrigin's function etc. If an algorithm is able to minimize these standard functions with utmost accuracy, then these algorithms are considered to be good algorithms. In this context the Bat algorithm, according to the authors is the best algorithm as it is one of the few algorithms which gives accurate results. This is one of the prime reasons for choosing this algorithm for workflow scheduling. Mapping of tasks and process units in a workflow is an important step. The mapping process is an NP-hard problem. These NP-hard problems cannot be solved in exhaustive manner i.e. it is not possible by any algorithm to solve it exhaustively within polynomial time. Hence a non-exhaustive approach is suitable for these kinds of problem and Bat algorithm is the obvious choice for this as it is the best among all the metaheuristic approaches.

This behavior of echolocation can be formulated in a way bat algorithms a kind of Swarm intelligence set of rules. Bat set of rules turned into designed to contain residences of several other metaheuristic algorithms so that it can perform better than other algorithms with the aid of thinking about the advantages and downsides of all of the other algorithms. These algorithms include the simulated annealing, Harmony seek and Particle swarm optimization. The bat algorithm achieves this and is considered to be a great set of rules which has the positives of different metaheuristic algorithms. The algorithm become advanced by means of Xin-She Zang within the year 2010 and is based totally on the subsequent regulations:

In order to sense the distance, bats use their echolocation behavior. Bats fly with certain velocity vi at position xi. They emit pulse whose frequency is adjusted automatically and the rate of the emitted pulse r ℵ [0, 1] is adjusted depending on the distance of the bat from its target. Loudness of bats varies from a large value L0 to a minimum value Lmin.

The Bat Algorithm is iterative in nature, hence for each iteration the parameters needs to be updated. After the position and velocity are updated the loudness and pulse emission rate needs to be updated. As the distance between the bat and the prey changes the bat has to automatically change the loudness and rate of pulse emission. The algorithm is described in detail by Xin-She Yang in his paper [20].

The algorithm above is described for continuous optimization but discrete set of inputs are given here, so as to suit the application, changes are made with reference to the Binary Bat Algorithm (BBA) [22].

Binary Bat algorithm is based on bat algorithm with a slight difference, bat algorithm is for continuous values but Binary Bat Algorithm can be used to select from Binary values. BBA algorithm is proposed for the feature selection. For feature extraction continuous values are not used for optimization.

Here, few modifications in binary bat algorithm are made and with bat algorithm it is used here for selecting the optimal cost.

Bat Algorithm

Objective function $f(x), X = (x_1, \dots. x_d)^T$
Initialize the bat population $x_i (i = 1,2, \dots, n)$*and* v_i
Define pulse frequency f_i *at* x_i
Initialize pulse rates r_i *and the loudness* A_i
while *(t < Max number of iterations)*
Generate new solutions by adjusting frequency, and updating velocities and locations/solutions
  if *(rand> r_i)*
  Select a solution among the best solutions
  Generate a local solution around the selected best solution
  end if
  Generate a new solution by flying randomly
  if *(rand < A_i & $f(x_i) < f(x_*)$)*
  Accept the new solutions
  Increase r_i *and reduce* A_i
  end if
Rank the bats and find the current best x_*
end while
Results / Visualizations

Figure 1 Bat Algorithm adapted from [21]

The pseudo-code of the BAT algorithm is given in Fig. 1. The BAT algorithm is a population based evolutionary algorithm where each bat represents a solution. It is designed according to the echolocation behavior of the virtual bats. This property enables them to detect the position of their prey. Frequency modulated signals are used for echolocation. Bats listen to the sound pulse emitted by them when it bounces back from the prey or the surrounding objects. As the bat approaches near its prey, it reduces the loudness of their echo and increases the rate of the sound pulse. These characteristics are formulated in a way described in the pseudo code given in Fig. 1 such that they associate with the objective function to be optimized. For the problem of scheduling workflows in cloud environment, a bat individual represents the task to virtual machine mapping for each task in the workflow.

Although the BAT algorithm has not yet been explored for scheduling purposes in clouds but due to certain amazing features that it exhibits like frequency tuning, capability of automatically zooming into the regions where optimal solution lies and parameter control, it can prove to be a promising algorithm for scheduling cloud resources. Not only this, algorithms like the Harmony search and PSO can be considered as the special cases of the BAT algorithm.

Replacing variations of frequency in the BAT algorithm by random parameters and setting loudness and pulse rate to 0 and 1 respectively, it behaves like the standard PSO. If we fix loudness and pulse rate and do not use velocities in the BAT algorithm, it simply reduces to the Harmony search. So we can say that the BAT algorithm is very powerful and have high chances to produce the good quality results.

Objective Function

The core of any heuristic or meta-heuristic algorithm (including the BAT algorithm) is the objective function. In this paper, the goal is to schedule workflow tasks on cloud resources in such a way that the execution time is minimized and reliability is maximized while ensuring that the cost is within the user specified budget. If S denotes schedule, then the execution time, cost and reliability of the schedule S are calculated as follows:
Execution time: The execution time of the schedule S is denoted by ET(S) and is defined as the sum of the execution time of all the tasks of the schedule S. If n is the total number of tasks in a schedule, the execution time can be calculated as:

$$ET(S) = \sum_{i=1}^{n} T(i) \quad (1)$$

where T(i) is the execution time of the task i in the schedule and is calculated by the following formula:

$$T(i) = \frac{\text{Instruction length of the task } i}{\text{MIPS rate of the virtual machine on which the task is scheduled}} \quad (2)$$

Cost: The cost of the schedule S is denoted by C(S) and is defined as the sum of execution cost of all the tasks of the schedule S. If n is the total number of tasks, cost can be calculated as:

$$C(S) = \sum_{i=1}^{n} C(i) \quad (3)$$

where C(i) = (Instruction length of the task i) * (cost scaling factor) and cost scaling factor is high for virtual machines of higher reliability and low for less reliable virtual machines.

Reliability: Reliability of a virtual machine depends on certain characteristics of the virtual machine like the processor speed (in MIPS), RAM size, transfer rate and failure rate. Reliability of the schedule S is denoted by R(S) and is decided by the following formula:

$$R(S) = \min\left(r(1), r(2) \ldots\ldots r(n)\right) \qquad (4)$$

where n is the total number of tasks and r(1) denotes the reliability of the VM on which first task is scheduled,
r(2) denotes the reliability of the VM on which second task is scheduled and so on.

We denote the fitness function by F. If MET(S) is the maximum execution time of the schedule among all the Bat individuals and B is the user specified budget then the fitness function is formulated as follows:

$$F = \begin{cases} \frac{ET(S)}{MET(S)} * \frac{1}{R(S)}, & if \ \frac{C(S)}{B} \leq 1 \\ \frac{C(S)}{B}, & if \ \frac{C(S)}{B} > 1 \end{cases}$$

The aim is to minimize the value of the fitness function so that the execution cost is minimized and reliability is maximized so that the schedules whose cost exceeds the budget constraint are ruled out.

The value of the fitness function will be less than 1 for the schedules whose cost will be within the user specified budget. For the schedules whose cost exceeds the user specified budget, the value of the fitness function will always be greater than 1.

It is to be noted that the fitness function is a minimization function and it will assume lower values for fit schedules and higher values for unfit schedules.

IV. RESULTS AND COMPARISONS

The WorkflowSim simulator developed by Chen et al.22 is used for simulations. 200 virtual machines with MIPS rate varying from 1000 to 4000 are considered for experiments.

Population size is taken as 30 and the initial population is generated randomly. The results of the BAT algorithm are compared with the Basic Randomized Evolutionary Algorithm (BREA).

BREA randomly generates a schedule and then randomly selects a task and replaces the virtual machine assigned to it with a machine having lower cost, then selects another task and replaces the virtual machine assigned to it with a machine having lower time and similarly randomly selects third task and replaces the virtual machine assigned to it with a machine having higher reliability.

Fig. 2 shows the comparison of the execution time of the BAT and the BREA technique and Fig. 3 shows the comparison of the cost value of the BAT and the BREA technique when applied to five different workflows namely Montage workflow with 25 nodes.

In spiral workflow with 50 nodes, Montage workflow with 100 nodes, Cyber Shake workflow with 1000 nodes and Southeastern Coastal Ocean Observing and Prediction Program (SCOOP) workflow.

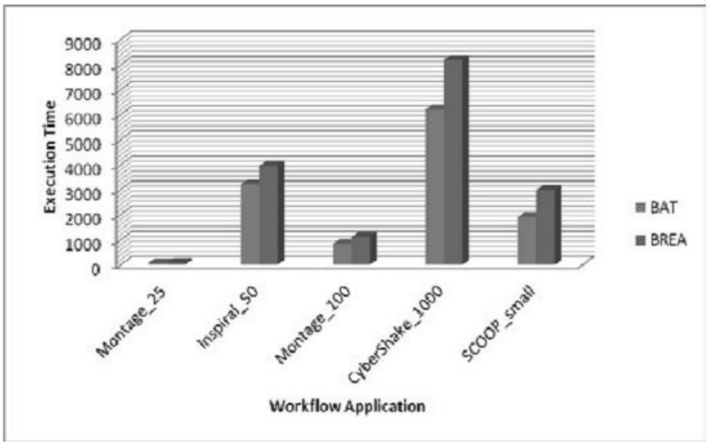

Fig. 2. Execution time comparison of the BAT and the BREA technique on 5 workflow applications

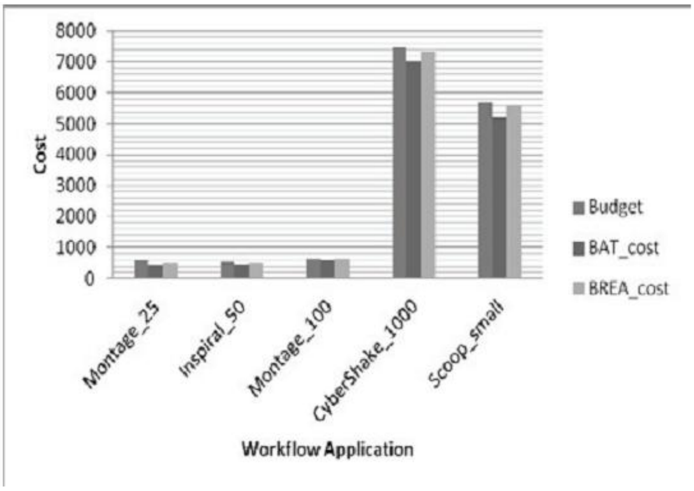

Fig. 3. Cost comparison of the BAT and the BREA technique on 5 workflow applications

The horizontal axis in both the cases represents the workflow application considered for the experiments.

The vertical axis of the graph in Fig. 2 gives the actual execution time taken by the BAT and the BREA algorithm. For all the five workflows, the BAT takes less time to execute as compared to the BREA. The vertical axis in Fig. 3 represents the actual cost values. Comparison of the cost value of the BAT algorithm and the BREA are made against the user defined budget. The results show that for both the algorithms, cost value does not exceed the user defined limit and the BAT algorithm produces low cost schedule as compared to the BREA for all the five workflows. The results clearly show that the BAT algorithm outperforms the BREA in all the scenarios.

The BAT algorithm schedules the cloud resources to the tasks of the workflow in less time and higher reliability within the user defined budget constraint as compared to the BREA technique.

V. CONCLUSIONS

In this paper, a recently developed meta-heuristic technique called BAT algorithm is used for workflow scheduling in clouds. The algorithm not only optimizes the execution time but also ensures that the tasks are assigned to high reliability virtual machines while meeting the user defined budget. Five different scientific workflows are considered to experiment and evaluate the effectiveness of the technique. The results obtained clearly show that the BAT algorithm performs better than the Basic Randomized Evolutionary Algorithm (BREA) in all the five scenarios.

References:

[1] Lin, Cui, and Shiyong Lu. "Scheduling scientific workflows elastically for cloud computing." Cloud Computing (CLOUD), 2011 IEEE International Conference on. IEEE, 2011.

[2] Pandey, Suraj, et al. "A particle swarm optimization-based totally heuristic for scheduling workflow applications in cloud computing environments." Advanced Information Networking and Applications (AINA), 2010 twenty fourth IEEE International Conference on. IEEE, 2010.

[3] Wu, Zhangjun, et al. "A revised discrete particle swarm optimization for cloud workflow scheduling." Computational Intelligence and Security (CIS), 2010 International Conference on. IEEE, 2010.

[4] Malawski, Maciej, et al. "Cost-and closing date-restricted provisioning for medical workflow ensembles in IaaS clouds." Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis. IEEE Computer Society Press, 2012.

[5] Abrishami, Saeid, Mahmoud Naghibzadeh, and Dick HJ Epema. "Deadline-limited workflow scheduling algorithms for Infrastructure as a Service Clouds." Future Generation Computer Systems 29.1 (2013)

[7] Wu, Zhangjun, et al. "A market-orientated hierarchical scheduling method in cloud workflow structures." The Journal of Supercomputing 63.1 (2013).

[8] Liu, Xiao, et al. "SwinDeW-C: a peer-to-peer based totally cloud workflow machine." Handbook of Cloud Computing. Springer US, 2010.

[9] Altintas, Ilkay, et al. "Kepler: an extensible machine for layout and execution of medical workflows." Scientific and Statistical Database Management, 2004. Proceedings. 16th International Conference on. IEEE, 2004.

[10] Calheiros, Rodrigo N., et al. "CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms." Software: Practice and Experience 41.1 (2011).

[11] Genez, Thiago AL, Luiz F. Bittencourt, and Edmundo RM Madeira. "Workflow scheduling for SaaS/PaaS cloud vendors considering two SLA ranges." Network Operations and Management Symposium (NOMS), 2012 IEEE. IEEE,

[12] Kennedy, James, and Russell Eberhart. "Particle swarm optimization." Proceedings of IEEE global convention on neural networks. Vol. Four. No. 2. 1995.

[13] Montage : An Astronomical Image Mosaic Engine [Online] Available: http://montage.Ipac.Caltech.Edu/docs/gridtools.Html

[14] Tsai PW, Pan JS, Liao BY, Tsai MJ, Istanda V. Bat algorithm stimulated set of rules for fixing numerical optimization issues. In: Applied Mechanics and Materials 2012. Trans Tech Publications; 2012.

[15] Yang XS. Bat set of rules for multi-goal optimization. International Journal of Bio-Inspired Computation.

[16] Ramesh B, Mohan VC, Reddy VV. Application of bat set of rules for mixed financial load and emission dispatch. Int. J. Of Electrical Engineering and Telecommunications. 2013.

[17] Musikapun P, Pongcharoen P. Solving multi-level multi-system multi-product scheduling trouble using bat algorithm. In: 2d worldwide conference on control and artificial intelligence 2012. IACSIT Press Singapore.2012; 35:ninety eight-102.

[18] Altringham JD. Bats: biology and behaviour. Oxford, UK: Oxford University Press; 1996.

[19] Colin T. The varienty of lifestyles. Oxford, UK: Oxford University; 2000.

[20] Richardson P. Bats. London: Natural History Museum; 2008.

[21] Yang XS. A new metaheuristic bat-stimulated set of rules. In: Nature stimulated cooperative strategies for optimization (NICSO 2010). Springer Berlin Heidelberg; 2010.

[22] Chen W, Deelman E. Workflowsim: A toolkit for simulating clinical workflows in allotted environments. In: E-Science (e-Science), 2012 IEEE 8th International Conference; 2012.

BIOGRAPHIES

Prachi Chaturvedi,pursued Bachlors of engineering Institute of Technology and Management(ITM), India in 2014. She is currentlyPursuing her Master of Engineering from Madhav Institute of Technology and Science,Gwalior (MP), India.

Abhishek Satyarthi,pursued Bachlors of Engineering From LNCT, Bhopal (MP), India in 2013. He is Currently pursuing his Master of Engineering from Madhav Institute of Technology and Science,Gwalior (MP), India

Sanjiv Sharma PhD, M.Tech(IT), B.E.(IT) is an Assistant Professor in the Department of Computer Science Engineering and Information Technology at Madhav Institute of Technology & Science Gwalior (MP), India. He received his PhD degree (Computer Science & Engineering) from Banasthali University, Jaipur (Raj.), India in 2014 and M.Tech (Information Technology) with honors from School Of Information Technology, Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal(MP), India in 2007. His current research interests include Social Network Analysis, Data Mining, Network Security, Adhoc Network and Mobile Computing and their interdependency.