Genetic-Based Task Scheduling Algorithm in Cloud Computing Environment

Safwat A. Hamad
Department of Computer Science,
Faculty of Computers & Information, Cairo University,
Cairo, Egypt

Fatma A. Omara
Department of Computer Science,
Faculty of Computers & Information, Cairo University,
Cairo, Egypt

***Abstract*—Nowadays, Cloud computing is widely used in companies and enterprises. However, there are some challenges in using Cloud computing. The main challenge is resource management, where Cloud computing provides IT resources (e.g., CPU, Memory, Network, Storage, etc.) based on virtualization concept and pay-as-you-go principle. The management of these resources has been a topic of much research. In this paper, a task scheduling algorithm based on Genetic Algorithm (GA) has been introduced for allocating and executing an application's tasks. The aim of this proposed algorithm is to minimize the completion time and cost of tasks, and maximize resource utilization. The performance of this proposed algorithm has been evaluated using CloudSim toolkit.**

Keywords—Cloud computing; Task Scheduling; Genetic Algorithm; Optimization Algorithm

I. Introduction

Due to the development of virtualization and Internet technologies, Cloud computing has emerged as a new computing platform [1]. Cloud computing can be defined as a type of distributed system consisting of a collection of interconnected and virtualized computers that are dynamically provisioned. It provides one or more consolidated computing resources based on service-level agreements (SLA) between the service providers and service consumers [2].

Cloud computing has some challenges (e.g., security, performance, resource management, reliability, etc.) [3]. One of the resource management issues is related to task scheduling. Task scheduling on Cloud computing refers to allocating the users' tasks on the available resources to improve execution of tasks, and increase resource utilization [4].

As the allocation of Cloud resource is based on SLA, the task execution cost is considered one of the main performance parameters of the task scheduling algorithm [5]. On the other hand, the task scheduling algorithm is considered a complex process because it must schedule a large number of tasks into the available resources. In the other side, there are many parameters that should be taken into consideration to develop a task scheduling algorithm. Some of these parameters are important from the Cloud user perspective (i.e., tasks compilation time, cost, and response time). Other parameters are important from the Cloud provider perspective (i.e., resource utilization, fault tolerant, and power consumption) [6].

The task scheduling problem is considered NP-Complete problem. Therefore, optimization approaches could be used to solve it by considering performance parameters (i.e., completion time, cost, resource utilization, etc.) [6]. The aim of this paper is to develop a task scheduling algorithm in the Cloud computing environment based on Genetic Algorithm for allocating and executing independent tasks to improve task completion time, decrease the execution cost, as well as, maximize resource utilization.

The rest of the paper is as follows: Section 2 discusses the related work. In Section 3, the principles of the modified GA-based task scheduling are described. The configuration of the CloudSim simulator, implementation of the proposed Genetic Algorithm, as well as, performance evaluation are discussed Section 4. Finally, conclusion and future work are given in Section 5.

II. Related Work

In recent years, the problem of task scheduling on a distributed environment has caught the attention of researchers. The main issue is the execution time which should be minimized. On the other hand, scheduling of tasks is considered a critical issue in the Cloud computing environment by considering different factors like completion time, the total cost for executing all users' tasks, utilization of the resource, power consumption, and fault tolerance.

GE Junwei [6] has presented a static genetic algorithm by considering total task completion time, average task completion time, and cost constraint.

One of the scheduling issues is to allocate the correct resource to the arriving tasks. The dynamic scheduling process is considered complex if several tasks arrive at the same time, Therefore, S. Ravichandran and D. E. Naganathan [7] have introduced a system to avoid this problem by allowing the arrived tasks to wait in a queue and the scheduling will recompute and sort these tasks. Therefore, the scheduling is done by taking the first task from the queue and allocated to the resource that will be the best fit using GA. The objective of this system is to maximize utilization of resources as also reduce execution time.

R. Kaur and S. Kinger [5] have proposed task scheduling algorithm-based enhancement GA. They use a new fitness function based on mean and grand mean values. They claim that this algorithm could be implemented on both task and resource scheduling.

A comparative study of three task scheduling algorithms on the Cloud computing environment - round-robin, pre-emptive priority and shortest remaining time first algorithms - has been done in [8].

V. V. Kumar and S. Palaniswami [9] have introduced a study focusing on increasing the efficiency of the task scheduling algorithm for real-time Cloud computing services. Additionally, they have introduced an algorithm to utilize the turnaround time by assigning high priority for the task of early completion time and less priority for abortion issues of real-time task.

Z. Zheng, *et al.* [10] have proposed an algorithm based on GA to deal with scheduling problem in the Cloud computing environment called Parallel Genetic Algorithm (PGA) to achieve the optimization or sub-optimization for Cloud scheduling problems mathematically.

Furthermore, one of the main goals of task scheduling from the perspective of a Cloud provider is to maximize the profit by utilizing resource efficiently. Therefore, K. Thyagaarajan, *et al.* [11] have introduced a model for task scheduling in the Cloud computing environment for an effective gain of profits on the Cloud computing service provider.

In [12], S. Singh has provided an elaborate idea about GA by introducing several variants for task scheduling in the Cloud computing environment. He has introduced an algorithm to solve task scheduling problem by modifying GA in which initial population is generated by Max-Min approach to get more optimum results in term of “makespan”.

III. THE PROPOSAL GENETIC BASED TASK SCHEDULING ALGORITHM

The Cloud provider should guarantee optimal scheduling of user's tasks in the Cloud computing environment according to SLA. At the same time, he should guarantee the best throughput and good utilization of the Cloud resources.

Generally, by increasing the users’ tasks, the complexity of scheduling these tasks in the Cloud computing environment will be increased proportionally. Therefore, the Cloud provider needs a good algorithm to schedule the users' tasks on the Cloud to satisfy QoS, minimize makespan, and guarantee good utilization of the Cloud resources [5]. Therefore, task scheduling is classified as an optimization problem.

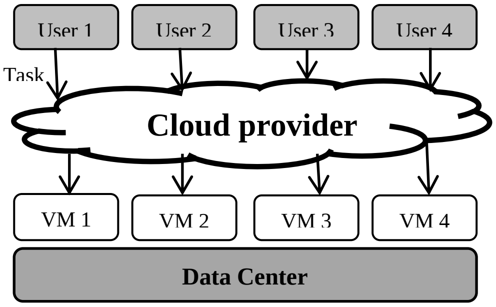

Fig. 1. Task scheduling principles

Fig. 1 illustrates the task scheduling process where each user introduces his application’s tasks, and the Cloud provider uses the appropriate approaches to schedule these tasks by considering some optimization parameters, such as minimum makespan, resources utilization, and minimum cost.

Therefore, the optimization problem can be solved using heuristic algorithm such as Genetic Algorithm (GA), Particle Swarm Optimization (PSO), and Ant Colony Optimization (ACO).

In this work, the proposed task scheduling algorithm in the Cloud environment is based on the default GA with some modifications. According to these modifications, the parents will be considered in each population beside the produced child after the crossover process. Also, the Tournament Selection is used to select the best chromosomes to overcome the limitation of the population size. Therefore, the proposed algorithm is called **T**ournament **S**election **G**enetic **A**lgorithm (TS-GA).

A. Genetic Algorithm

Genetic Algorithm (GA) is based on the biological concept of generating the population. GA is considered a rapidly growing area of Artificial Iintelligence [1] [2]. By Darwin's theory of evolution was inspired the Genetic Algorithms (GAs). According to Darwin's theory, term “Survival of the fittest” is used as the method of scheduling in which the tasks are assigned to resources according to the value of fitness function for each parameter of the task scheduling process [13]. The main principles of the GA are described as follows [1] [2]:

1) Initial Population

The initial population is the set of all individuals that are used in the GA to find out the optimal solution. Every solution in the population is called as an individual. Every individual is represented as a chromosome for making it suitable for the genetic operations. From the initial population, the individuals are selected, and some operations are applied on them to form the next generation. The mating chromosomes are selected based on some specific criteria.

2) Fitness Function

The productivity of any individual depends on the fitness value. It is the measure of the superiority of an individual in the population. The fitness value shows the performance of an individual in the population. Therefore, the individuals survive or die out according to the fitness or function value. Hence, the fitness function is the motivating factor in the GA.

3) Selection

The selection mechanism is used to select an intermediate solution for the next generation based on the Darwin's law of survival. This operation is the guiding channel for the GA based on the performance. There are various selection strategies to select the best chromosomes such as roulette wheel, Boltzmann strategy, tournament selection, and selection based on rank.

4) Crossover

Crossover operation can be achieved by selecting two parent individuals and then creating a new individual tree by alternating and reforming the parts of those parents. Hybridization operation is a guiding process in the GA and it boosts the searching mechanism.

5) Mutation

After crossover, mutation takes place. It is the operator that introduces genetic diversity in the population. The mutation takes place whenever the population tends to become homogeneous due to repeated use of reproduction and crossover operators. It occurs during evolution according to a user-defined mutation probability, usually set to fairly low. Mutation alters one or more gene values in the chromosome from its initial state. This can produce the entirely new gene values being added to the gene pool. With this new gene values, the genetic algorithm may be able to produce a better solution than was previously (see Fig. 2) [1].

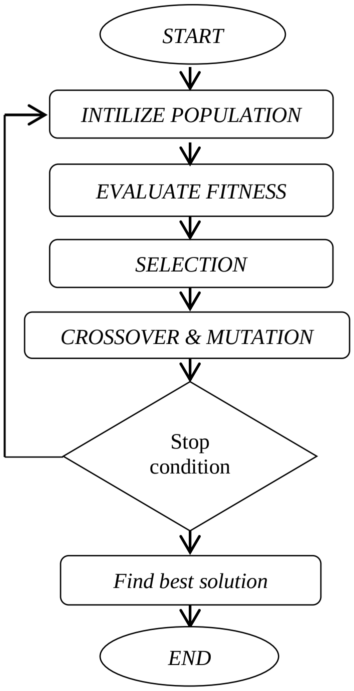

Fig. 2. Genetic Algorithm [14]

B. The Proposed Tournament Selection Genetic Algorithm (TS-GA)

In this work, a modified GA is proposed to solve task scheduling problem in Cloud computing environment to enhance the completion time for executing all tasks on the VMs, in the same time, minimize the total cost of usage the resource and maximize utilization of the resource. The main idea of this proposed algorithm (i.e., TS-GA) is that after each selection in the population, there is a solution that might satisfy good fitness function, but it is not selected to crossover process. By the proposed algorithm, this solution is not removed from the population, but it is chosen and added to the population when next iteration is started. This step is considered as a good step as some of the iterations can generate the best solution.

1) Initialize Population

According to the proposed TS-GA algorithm, the population is randomly generated using encoded binary (0, 1).

Therefore, the representation of solutions in task scheduling for each gene or (chromosome) consists of VM ID and ID for each task to be executed on these VM (see Fig. 3).

Each VM and the executed tasks on it are encoded into the binary bit (e.g., VM3: - TS4-TS8-TS9 → [0110 – 0100-1000-1001]).

2) The Fitness Function Representation

The main objective of task scheduling in the Cloud computing is to reduce completion time for execution all tasks on the available resources. Therefore, the completion time of task Ti on VMj as CTij is defined using equation "(1)" [15]:

$$\text{Completion Time} = \mathrm{CTmax}[\mathrm{i},\mathrm{j}]$$

$$\mathrm{i} \in \mathrm{T}\,, \quad \mathrm{i} = 1, 2, 3, \ldots \mathrm{n}$$

$$\mathrm{j} \in \mathrm{VM} \quad , \quad \mathrm{j} = 1, 2, 3, \ldots \mathrm{m} \qquad (1)$$

Where CTmax denotes maximum time for complete Task i on VMj. *n* and *m* denote the number of tasks and virtual machines respectively.

Therefore, to reduce the completion time which can be denoted as CTmax, the execution time of each task for each virtual machine must be calculated for the scheduling purpose. If the processing speed of virtual machine VMj is PSj, then the processing time for task Pi can be calculated by equation "(2)" [15]:

$$\mathrm{Pij} = \frac{\mathrm{Ci}}{\mathrm{Psj}} \qquad (2)$$

Where, Pij the processing time for task Pi on VMj and Ci computational complexity of task Pi

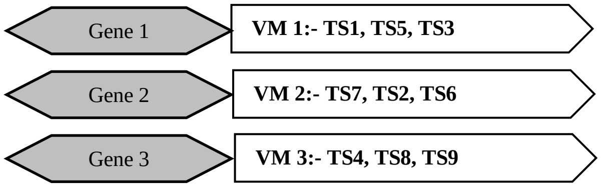

Fig. 3. Representation tasks and VMs

The processing time of each task in the virtual machine can be calculated by equation "(3)" [15]:

$$\mathrm{Pj} = \sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{Pij} \qquad (3)$$

3) Selection Process

Tournament selection is computationally more efficient and more amenable to parallel implementation [16]. Therefore, the developed TS-GA algorithm, Tournament Selection is used to overcome the limitation of the population size. Two individuals are chosen at random from the population. A random number r is then chosen between 0 and 1. If r < k (where k is a parameter, for example, 0.75), the fitter of the two individuals is selected to be a parent; otherwise the less fit individual is selected. The non-chosen individuals are then returned to the original population and could be selected again.

4) *Crossover*

In the proposed TS-GA algorithm, the new crossover has been used differently from the used crossover in the original GA. Therefore, two chromosomes which are selected to crossover process to generate two offspring will be considered as offspring also. So, the proposed crossover produces four children (see Figure 4). After that, the two best children are chosen from these.

5) *Initialize Subpopulation*

After each iteration, subpopulations (i.e., new populations after crossover) are added into old populations (i.e., parents). This step can enhance the diversity of population. This idea is introduced by the modified TS-GA algorithm.

6) *Keep Best Solution*

There is a solution that might satisfy good fitness function, but it is not selected during the crossover process. By the proposed TS-GA algorithm, this solution is not removed from the population, but it is chosen and added to the population when next iteration is started. This step is considered as good step as some of the iterations can generate the best solution.

Generally, according to the modified TS-GA algorithm, a set of modifications have been introduced. These modifications are as follows.

- The tournament is used instead of the roulette wheel in the selection process to select the best solution.
- The solutions not chosen in the selection process are considered and added to the new population. This might help in generating the best solution in the next generations.
- The new crossover is introduced by considering parents individuals as new child (see Fig. 4)

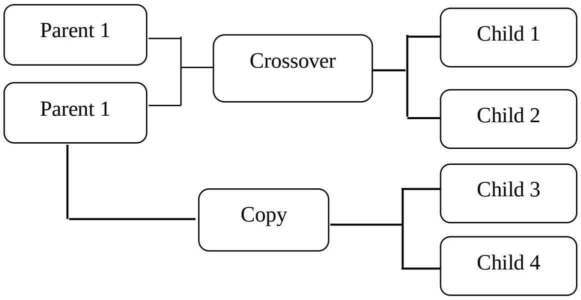

Fig. 4. Crossover process

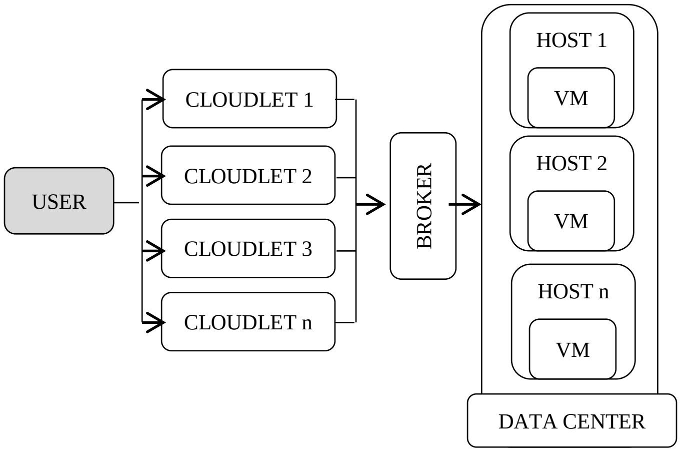

Fig. 5. CloudSim Behavior [17]

- After each iteration, subpopulations (i.e., new populations after crossover) are added into old populations (i.e., parents).

IV. P PERFORMANCE EVALUATION

In this section, the experimental evaluation of the proposed TS-GA algorithm on the original GA, and Round-Robin algorithms is presented, starting by describing the experimental environment.

A. *The Experimental Environment*

The CloudSim toolkit helps the researchers to simulate cloud computing environment, where it is released by the Cloud Computing and Distributed Systems Laboratory, University of Melbourne [17]. In other words, it provides the features of modeling and simulation of Cloud computing environment. According to CloudSim, the user tries to submit his requests in the form of cloudlets. Each cloudlet has the properties of file size, the number of instructions to be executed, etc. These cloudlets will be submitted to the broker to schedule onto VMs according to scheduling. CloudSim has an advantage of the building of broker driven policies. The defined class in CloudSim, VM, represents the virtual machine which can be created on the hosts. Creation of hosts depends on the broker where it allocates each VM to the different host. Datacenter has the capability to hold a maximum number of hosts and the broker can dynamically change the setup of hosts and VMs (see Fig. 5) [17].

B. *Experimental Results*

By using CloudSim toolkit, the proposed TS-GA is implemented, and a comparative study has been made among three algorithms; Round-Robin (RR), the default GA, and the improved TS-GA algorithms. Five parameters are considered to evaluate the performance. These parameters are the completion time, cost, resource utilization, speedup, and efficiency.

1) Completion Time:

Table 1, and Fig. (6) represent the completion time of RR, default GA and the proposed TS-GA algorithms using 8 VMs.

TABLE I. THE COMPLETION TIME OF RR, GA, AND TS-GA ALGORITHMS USING EIGHT VMS

No. of Task	**RR**	**GA**	**TS-GA**	**Improve TS-GA vs. GA**	**No. VM**
25	174.78	292.51	122.19	58.22 %	8
50	578.77	420.21	286.12	31.91 %	
75	582.16	680.84	405.97	40.37 %	
100	954.37	812.89	513.7	36.8 %	

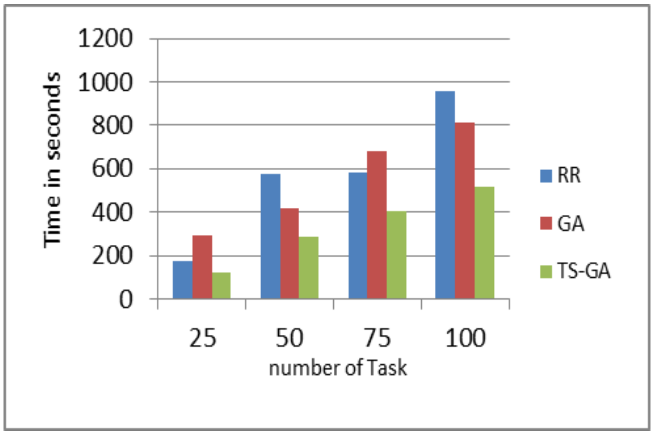

Fig. 6. the comparison completion time of three algorithms RR, GA and TS-GA

According to the results in Fig. (6), it is found that the completion time of the proposed TS-GA algorithm is reduced by (41.83%) and (39.26%) about the default GA, and RR algorithms respectively.

2) Execution Cost:

In addition, the total cost of execution of all tasks on the available VMs is calculated as "(4)" [18]:

$$Total\ Cost = \frac{Task\ length\ *\ Cost\ per\ seconds}{VM\ mips} + Processing\ Cos \quad (4)$$

Table 2. and Fig. (7) represent the execution cost of RR, default GA and the proposed TS-GA algorithms using 8 VMs.

TABLE II. THE EXECUTION COST OF RR, GA AND TS-GA ALGORITHMS

No. of Task	**RR**	**GA**	**TS-GA**	**IMPROVE TS-GA vs. GA**	**No. VM**
25	3017.96	3012.98	2915.21	3.24 %	8
50	7543.22	6813.46	6603.88	1.3 %	
75	10517.14	10549.22	9910.48	6.05 %	
100	13515.05	13118.87	12618.22	3.81 %	

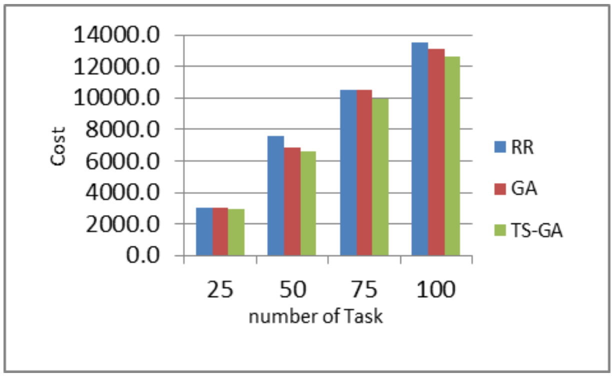

Fig. 7. The comparison cost of three algorithms RR, GA and TS-GA

According to the results in Fig. (7), the cost of the proposed TS-GA algorithm is reduced by (3.6%) and (6.07%) relative to the default GA and RR algorithms respectively.

3) Resource Utilization:

On the other side, the utilization of resources represents the ratio between the total busy time of Virtual Machine and the total finish execution time of the parallel application. It is defined as "(5)" [19]:

$$\text{utilization} = \frac{\text{final VMs available time}}{\text{\#VMs} \ * \ \text{schedule time}} * 100 \qquad (5)$$

Table 3, and Fig. (8) represent the resource utilization of RR, default GA and the proposed TS-GA algorithms using 8 VMs.

TABLE III. THE RESOURCE UTILIZATION OF RR, GA AND TS-GA ALGORITHMS

No. of Task	**RR**	**GA**	**TS-GA**	**IMPROVE TS-GA vs. GA**	**No. VM**
25	55.13	31.51	75.89	58.47	8
50	44.36	47.88	84.43	43.29	
75	69.4	39.96	80.75	50.51	
100	52.04	48.7	75.78	35.73	

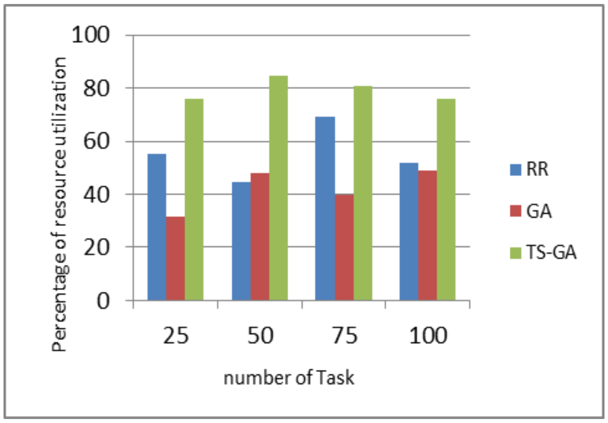

Fig. 8. Resource Utilization of RR, GA and TS-GA algorithms

According to the results in Fig. (8), it is found that the resource utilization of the proposed TS-GA algorithm is improved by (47%) and (30.04%) relative to the default GA and RR algorithms respectively.

4) Speedup and Efficiency:

The speedup and efficiency for each VM are calculated as “(6)”, “(7)” follows [19]:

$$\text{Speedup} = \frac{final\ sum\ of\ excute\ time\ of\ tasks\ to\ one\ VM/\#VMs}{schedule\ time} \qquad (6)$$

$$\text{Efficiency} = \frac{speedup}{\#\ VMs} \qquad (7)$$

Table 4, Fig. (9) and Fig. (10) represent the speedup and efficiency of RR, default GA and the proposed TS-GA algorithms using 8 VMs.

TABLE IV. SPEEDUP AND EFFICIENCY FOR RR, GA, AND TS-GA

No. TASK	Algorithm	Speedup	Efficiency
25	RR	0.193	0.241
	GA	0.143	0.179
	TS-GA	0.248	0.311
50	RR	0.314	0.393
	GA	0.419	0.524
	TS-GA	0.558	0.697
75	RR	0.598	0.748
	GA	0.543	0.679
	TS-GA	0.829	1.037
100	RR	0.622	0.778
	GA	0.691	0.864
	TS-GA	1.105	1.319

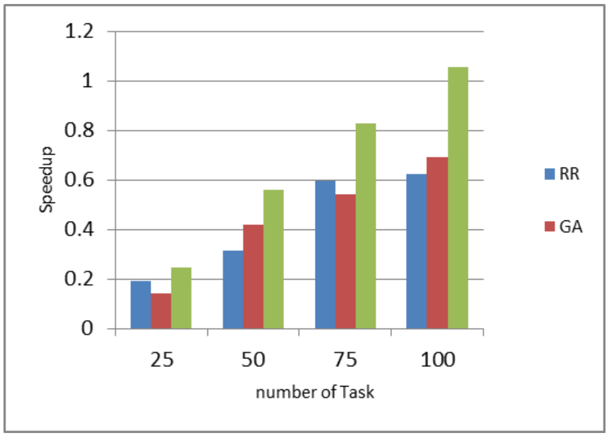

Fig. 9. The comparison speedup of three algorithms RR, GA and TS-GA

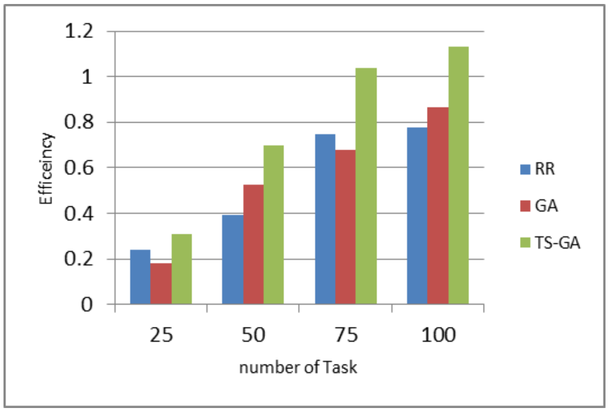

Fig. 10. The comparison efficiency of three algorithms RR, GA and TS-GA

According to the results in Fig. (9), it is found that the speedup of the proposed TS-GA algorithm is improved by (34.03%) and (33.65%) about the default GA and RR algorithms respectively. Also, the efficiency of the proposed TS-GA algorithm is improved by (34.06%) and (33.66%) about the default GA and RR algorithms respectively (see Fig. 10).

The average improved of speedup and efficiency of the proposed TS-GA algorithm about the default GA and RR algorithms are presented in Table 5 and Table 6 respectively.

TABLE V. AVERAGE IMPROVEMENT OF SPEEDUP AND EFFICIENCY FOR TS-GA ALGORITHM RELATIVE TO DEFAULT GA ALGORITHM

No. TASK	25	50	75	100	Average improvement
Speedup	42.35	24.82	34.51	34.45	34.03 %
Efficiency	42.44	24.82	34.52	34.49	34.06 %

TABLE VI. AVERAGE IMPROVEMENT OF SPEEDUP AND EFFICIENCY FOR TS-GA ALGORITHM RELATIVE TO ROUND-ROBIN ALGORITHM

No. TASK	25	50	75	100	Average improvement
Speedup	22.22	43.57	27.86	40.98	33.65 %
Efficiency	22.23	43.58	27.88	40.9	33.69 %

V. CONCLUSION AND FUTURE WORK

This paper proposes an improved Genetic Algorithm for task scheduling problem in the Cloud computing environment. The proposed algorithm targets to minimize completion time and cost, and maximize resource utilization. The completion time for the proposed TS-GA algorithm is reduced by

(41.83%) and (39.26%) about the default GA, and RR algorithms, respectively. The cost of the proposed TS-GA algorithm is reduced by (3.6%) and (6.07%) about the default GA and RR algorithms respectively. The resource utilization of the proposed TS-GA algorithm is improved by (47%) and (30.04%) about the default GA and RR algorithms, respectively. The speedup of the proposed TS-GA algorithm is improved by (34.03%) and (33.65%) about the default GA and RR algorithms, respectively. The efficiency of the proposed TS-GA algorithm is improved by (34.06%) and (33.66%) about the default GA and RR algorithms respectively.

For future work, the proposed algorithm can be extended to add possibility dynamic characteristic of VMs through run GA. Moreover, more parameters can be added based on the users' requirements.

REFERENCES

[1] S. H. Jang, T. Y. Kim, J. K. Kim, and J. S. Lee, "The study of genetic algorithm-based task scheduling for cloud computing," International Journal of Control and Automation, vol. 5, pp. 157-162, 2012.

[2] T. Goyal and A. Agrawal, "Host Scheduling Algorithm Using Genetic Algorithm In Cloud Computing Environment," International Journal of Research in Engineering & Technology (IJRET) Vol, vol. 1, 2013.

[3] B. Furht, "Armando Escalante Handbook of Cloud Computing," ISBN 978-1-4419-6523-3, Springer2010.

[4] F. Etro, "Introducing Cloud Computing," in London Conference on Cloud Computing For the Public Sector, 2010, pp. 01-20.

[5] R. Kaur and S. Kinger, "Enhanced Genetic Algorithm based Task Scheduling in Cloud Computing," International Journal of Computer Applications, vol. 101, 2014.

[6] J. W. Ge and Y. S. Yuan, "Research of cloud computing task scheduling algorithm based on improved genetic algorithm," in Applied Mechanics and Materials, 2013, pp. 2426-2429.

[7] S. Ravichandran and D. E. Naganathan, "Dynamic Scheduling of Data Using Genetic Algorithm in Cloud Computing," International Journal of Computing Algorithm, vol. 2, pp. 127-133, 2013.

[8] V. Vignesh, K. Sendhil Kumar, and N. Jaisankar, "Resource management and scheduling in cloud environment," International Journal of Scientific and Research Publications, vol. 3, p. 1, 2013.

[9] V. V. Kumar and S. Palaniswami, "A Dynamic Resource Allocation Method for Parallel DataProcessing in Cloud Computing," Journal of computer science, vol. 8, p. 780, 2012.

[10] Z. Zheng, R. Wang, H. Zhong, and X. Zhang, "An approach for cloud resource scheduling based on Parallel Genetic Algorithm," in Computer Research and Development (ICCRD), 2011 3rd International Conference on, 2011, pp. 444-447.

[11] K. Thyagarajan, S. Vasu, and S. S. Harsha, "A Model for an Optimal Approach for Job Scheduling in Cloud Computing," in International Journal of Engineering Research and Technology, 2013.

[12] S. Singh and M. Kalra, "Scheduling of Independent Tasks in Cloud Computing Using Modified Genetic Algorithm," in Computational Intelligence and Communication Networks (CICN), 2014 International Conference on, 2014, pp. 565-569.

[13] R. Buyya, R. Ranjan, and R. N. Calheiros, "Modeling and simulation of scalable Cloud computing environments and the CloudSim toolkit: Challenges and opportunities," in High Performance Computing & Simulation, 2009. HPCS'09. International Conference on, 2009, pp. 1-11.

[14] J. S. Raj and R. M. Thomas, "Genetic based scheduling in grid systems: A survey," in Computer Communication and Informatics (ICCCI), 2013 International Conference on, 2013, pp. 1-4.

[15] B. Kruekaew and W. Kimpan, "Virtual Machine Scheduling Management on Cloud Computing Using Artificial Bee Colony," in Proceedings of the International MultiConference of Engineers and Computer Scientists, 2014.

[16] M. Mitchell, An introduction to genetic algorithms: MIT press, 1998.

[17] R. N. Calheiros, R. Ranjan, C. A. De Rose, and R. Buyya, "Cloudsim: A novel framework for modeling and simulation of cloud computing infrastructures and services," arXiv preprint arXiv:0903.2525, 2009.

[18] R. Sahal and F. A. Omara, "Effective virtual machine configuration for cloud environment," in Informatics and Systems (INFOS), 2014 9th International Conference on, 2014, pp. PDC-15-PDC-20.

[19] D. M.Abdelkader, F.Omara," Dynamic task scheduling algorithm with load balancing for heterogeneous computing," system Egyptian Informatics Journal, Vol.13, PP.135–145, 2012.