Hybrid Invasive Weed Optimization with Tabu Search Algorithm for an Energy and Deadline Aware Scheduling in Cloud Computing

Pradeep Venuthurumilli[1]
Research Scholor, Acharya Nagarjuna University
Guntur, Andhra Pradesh

Dr. Sridhar Mandapati[2]
Dept.of Computer Applications, R.V.R and J.C College of Engg., Chowdavaram, Guntur-522019, Andhra Pradesh

***Abstract*—The current existing high flexibility, profitability, and potential have made cloud computing extremely popular among the companies. This is used for improving and applying resources in an efficient manner and optimize makespan of the tasks. Scheduling is easy while there are only a few tasks to complete with few resources. Contrastingly, at the time the users forward several demands to the environment of the cloud, there may be a need for optimally selecting and allocating resources for achieving the desired quality of service that makes scheduling challenging. In this work, using intelligent metaheuristic algorithms for processing the requests and tasks of users in energy-aware scheduling made for a deadline is proposed. Genetic Algorithm (GA) the evolutionary algorithm that is inspired by the natural process of selection and the evolution theory. The Invasive Weed Optimization (IWO) was yet another novel stochastic based on the population that was a derivative-free technique of optimization inspired by the growth of the weed plants. The TABU Search (TS) was a generalization technique of local search where the TABU list was used for preventing cycling and further generating the candidates of the neighborhood. A hybrid GA with the TS (GA-TS) with a hybrid IWO with TS (IWO-TS) has been proposed for the energy and deadline aware scheduling. The framework further offers optimization of energy and performance. The primary purpose of this algorithm has been to improve deadline and scheduling in cloud computing along with local as well as global search algorithms. This framework will offer optimization of performance and energy. The reason behind presenting this algorithm was improving both scheduling and deadline in cloud computing using both local and global algorithms and results proved the algorithm to have better results.**

Keywords—Cloud computing; scheduling; Genetic Algorithm (GA); Invasive Weed Optimization (IWO) and Tabu Search (TS)

I. INTRODUCTION

Cloud computing can be termed as a new paradigm where computing has been delivered as a new service as opposed to being a product with shared resources. A major advantage of cloud computing is the reduction of capital expenditure for cloud users and their service providers. It is well-suited for a range of applications like hosting websites, scientific workflow, high-performance computing, customer services and social networking [1].

Cloud computing may be termed as a model of expanding computation that is based on the technology of virtualization which is in response to the request of users through the network of the internet with dynamic resource allocation. Such virtualization will reduce the cost of maintenance of the organizations and also improve their accessibility. For the purpose of cloud computing, at the time a user asks for services, task scheduling becomes important for selection. The primary objective of such scheduling is to identify a new and optimal mapping for a task set and scheduling is quite easy at the time the number of resources and tasks is less. On opposing, at the time users send different demands to the environment of the cloud, there may be a need to choose optimally and allocate resources to the desired quality of users making scheduling challenging [2].

The problems in scheduling can be challenging as are combinatorial optimization problems which is either a maximization or a minimization issue containing: (1) one set of instances, (2) one finite set of the candidate solutions for every instance, and (3) a function for every instance. These problems will be solved only by finding an optimal solution to it. The problems are also Non-Deterministic Polynomial (NP)-hard and finding an efficient method for this is challenging. There are other problems in scheduling like timetable scheduling, processor scheduling, job-shop scheduling and so on [3].

Both “meta” and “heuristic” originated from Greek. The former refers to a “higher level” and the latter refers to “be aware” or “investigate”. To identify a good and optimal solution that has low cost of computation there are heuristic methods used that do not have any guarantee of feasibility or optimality. For enhancing efficiency, there is a set of metaheuristics that are employed for which there are numerous methods. Furthermore, the current algorithms will have lag in performance such as being trapped into a low optimum or a slow convergence rate. They may also have complex operators. Thus, presenting newer algorithms for catering to the weaknesses is a major and problematic aspect here [4].

Speaking in simple terms, metaheuristics may be a general class of an algorithm that is employed for finding solutions to problems of optimization at the time the normal techniques fail to help. By using the metaheuristic, an objective function can be of a type that precedes into considering different objectives. At the same time, all the other metaheuristic algorithms will have a suspicious modification of the parameters of optimization that can be crucial to discover better solutions that lack time. There are several metaheuristics that are available with some novel variations recommended for the allocation of resources in different fields. Some more metaheuristics are very prominent in cloud computing and for the management of resources. These may be the Memetic Algorithm (MA), League Championship Algorithm (LSA), Ant Colony Optimization (ACO), Immune Algorithm (IA), Cuckoo Search (CS), Harmony Search, (HS), Firefly Algorithm (FA) and the Differential Evolution (DE).

This hybrid optimization approach was proposed for designing a linkage method for its dimensional synthesis to merge the advantages of the stochastic and the deterministic methods of optimization. For this work, the GA, TS, IWO, hybrid GA-TS and the IWO-TS algorithm were proposed for the deadline scheduling that is energy-aware. The rest of this examination has been composed therefore. The related work in the writing has been examined in Section 2. All strategies utilized are clarified in Section 3. Results were presented in Section 4 and the end was made in Section 5.

II. Literature Works

For the purpose of this work, the problem of task scheduling was taken to be a problem that was bi-objective that includes the minimization of consumption of energy and its makespan. Firstly, Sahoo et al., [5] had proposed another novel Learning Automata-based Scheduling (LAS) based framework for the tasks that were deadline sensitive. Later, there was a LAS algorithm that had been introduced to exploit the task heterogeneity. There was some extensive simulation that was conducted for designating the applicability and effectiveness of the LAS for the task scheduling that was deadline sensitive in a cloud environment.

For the purpose of this work, there was a new cost-aware job scheduling method that was based on the queuing theory presented by Li et al., [6]. The problem of job scheduling in a private cloud was modelled to be a queuing model. After this, the time for task execution was predicted using a backpropagation neural network. A max–min strategy was applied to the schedule tasks regarding the outcome of forecast in the half and half mists. Its goal is utilizing assets appropriately and overseeing load between assets with least execution time. High correspondence cost brings about in mists forestall task schedulers from being applied in an enormous scope conveyed condition. Booking is a noticeable movement that is executed in a distributed computing condition. To build distributed computing remaining task at hand productivity, assignments planning is performed to get greatest benefit. In cloud, high correspondence cost forestalls task schedulers from being applied in huge scope dispersed conditions. Cloud condition framework planning is NP-finished. To explain the NP complete and NP difficult issues heuristic methodologies are utilized.

The experiment showed this cost-aware job scheduling algorithm to decrease the average response waiting time for a job in a private cloud. Also, the proposed algorithm of job scheduling was able to progress the throughput of the system with an average task response time and also the total cost in the hybrid clouds.

The metaheuristic techniques will have their efficiency established in a range of algorithms of workflow scheduling but it is yet not known if the metaheuristic selected is well-suited to solve it. Shishido et al., [7] had examined the effects of the GA and the PSO in the optimization of workflow scheduling. There was an algorithm of security and cost-aware workflow scheduling chosen for the evaluation of metaheuristic performance. There were three algorithms evaluated using real-world workflows having a risk limitation ranging between 0 and 1 and a 0.1 step. The GA-based algorithms had suggestively overtook the PSO in terms of response time and efficacy of cost.

Liu et al., [8] had recently modelled task scheduling in the form of an optimization problem that considered time deadline constrained and unconstrained cases. For addressing this issue, there was a Heterogeneous Earliest Finish Time (HEFT) with a technique for a request for inclination to perfect arrangements named as the HEFT-T calculation. For an unconstrained case, there was another three-phase approach which depended on the HEFT-T calculation to pick ideal arrangements utilizing the non-overwhelmed arranging approach. On account of a cutoff time obliged choice, there was a versatile weight alteration put together methodology with respect to the HEFT-T that was proposed for adjusting weight values in the case of time. In comparison with the other algorithms that were state-of-the-art, the proposed one has performed better in terms of optimization criterion for cost and mean load.

Since there are certain inherent defects found in mobile devices like limited space for storage of battery backup, there are some challenges faced in the management of mobility, security, energy management, and QoS. This has resulted in the emergence of several paradigms like fog computing and Mobile Cloud Computing (MCC). All these paradigms permit offloading certain tasks for the execution of the cloud that makes tasks scheduling important in the mobile cloud and the mobile device. Tang et al., [9] [19] have modelled the problem to be of optimization of energy consumption and this also considers the dependency of the task, transmission of data and other constraints like cost and time deadline.

Haidri et al., [10] [18] had addressed these problems found in the scheduling of the workflow tasks on the cloud computing systems like Total Price for Execution (TPE) and the Total Execution Time (TET). This is for the workflow and for meeting the limitations of a deadline in an environment that is stochastic. As opposed to the TET and the TPE, the acquisition delay of the which is a major trait of the cloud. The work initially formulates an issue of a model of stochastic scheduling on the cloud. There is a Stochastic Cost-Effective Deadline-Aware (S-CEDA) based resource scheduler that had been established. There was the S-CEDA that had incorporated the variance and the expected value of the processing time of the task and inter-task communication workflow scheduling.

III. Methodology

Distinguishing a worldwide minimizer is a difficult errand. The creator has misused both the highlights of GA, IWO and TS be the altered variant of a coordinated calculation. Capacity of worldwide hunt and getaway from a nearby least is improved and a mixture GA-TS and IWO-TS methods are examined.

A. Genetic Algorithm (GA)

The GA had been settled centered on the principle of "survival of the fittest" by Charles Darwin. It exploits the best solution that has been accomplished by the searches conducted earlier along with an investigation of newer sections of their explanation planetary. A solution for a given problematic has been signified by a new gene that contained elements known as genes. The algorithm will begin with a population of solutions made in a random fashion and the population's quality will be identified by means of evaluating them with any objective function called the fitness function. Once the population is initialized using solutions generated randomly, there are new descendants genes that are generated using either the crossover or the mutation operations. The solutions generated are then evaluated and acclimated for being adapted to the people if it is better. This process that generates offspring will be repeated until an optimal solution with the best offspring has been created [11][17].

B. Invasive Weed Optimization (IWO) Algorithm

The IWO had been initially brought about by Mehrabian and Locus which is an algorithm based on population originated from the weed colonial behaviour. The IWO is quite simple but very efficient in identifying optimum solutions to the objective function. This has been implemented on the basis of the features of the weeks found in a colony as reproduction, the competition to survive and growth. When compared to the other algorithms, the IWO is very simple and has sufficient convergence ability and rate. There are some more major features found in the algorithm that is different from the other methods and these are exclusive competition, distribution of space and reproduction. For simulating the weed behaviour, the following algorithm is used as per [12].

Stage 1: The Initial populace creation: each populace will require N0 seeds that are disseminated haphazardly inside an n-dimensional space.

Stage 2: Reproduction: each seed will develop and turn into a develop plant after which the seed creation happens for another new age. The seeds that are created by each plant will increment directly between two distinct estimations of the base (Smin) and the greatest (Smax) that have potential measures of seeds delivered. The genuine measure of seeds that are created for the ith plant for each and every recurrent will be founded on the objective worth (Fi), it's the best (Fbest) and the most exceedingly awful (Fworst) objective qualities registered according to condition (1):

$$Numseed(i) = \left[S_{\min} + (S_{\max} - S_{\min}) \frac{f - f_{worst}}{f_{best} - f_{worst}} \right] \tag{1}$$

Step 3: The distribution will space both assimilation and randomness of this algorithm that is connected to this particular stage. The seeds that are produced will be distributed within the d dimensional space using a normal distribution that has a zero mean with a different variance which is ($N(0,\delta_t)^d$). The seeds will have to be closer to the breeder plant. Even though there is a standard deviation which will decrease from its initial amount ($\sigma_{initial}$) to the final amount (σ_{final}) for every repeat, in all simulations, where a non-linear variation of the standard deviation as in (2):

$$\sigma_t = \left(\frac{T-t}{T} \right)^n \times (\sigma_{initial} - \sigma_{final}) + \sigma_{final} \tag{2}$$

As per equations 1 & 2, T will represent the maximum repeats that are related to (t) and this will be a factor of non-linear modulation. For this status, the seed position (S_j) for that of the ith plant (W_i) will be calculated as per (3):

$$S_j = W_i + N(0,\delta_t)^d, 1 <= j \le numseed(i) \tag{3}$$

Step 4: Exclusive competition: the actual number of plants that are formed by a quick reproduction can reach its maximum value which is ($W_{\max}$). Each plant will be allowed to produce seeds according to the method of reproduction. These seeds will be authorized to spreading within the search space along with a correspondence to the distribution space method at the time the seeds reach their position; after this, they form a new colony alongside the parent plants. Those members having less propriety will be deleted for the other members to be able to reach the maximum allowable value. All parent plants will associate alongside their children and the ones having the highest propriety in the group will be preserved and replacement will be permitted. This method of the crowd control mechanism is forced on the subsequent generations till such time it reaches its final period.

Step 5: If criterion is met, end if not repeat from Step 2.

C. Tabu Search (TS) Algorithm

The TS is yet another metaheuristic algorithm that is used to solve problems in combinatorial optimization like the problem of bin packing. For the purpose of escaping from their potential local optimum, the TS will make use of a local or a neighbourhood search for moving from solution x to solution y in an iterative manner. The termination criteria used commonly for this is to make use of a fixed and maximum number of iterations, the maximum number of iterations that are done without refining the objective function, the maximum limit on the time taken for searching and so on. When the local search is executed, certain regions may not get traversed and the TS will resolve this by modifying their neighbourhood structure for the solution. These solutions will be acknowledged to $N_t(x)$, which is the new neighbourhood and will be determined by using memory structures [13]. This search will progress by an iterative move from solution x to solution x' as in $N_t(x)$.

The TS will make use of random selection by default and for speeding up the process of search it also uses a “TABU List (Tabus)” that can restrict search space and avoid cyclic behaviour. This TABU list is of short-term memory consisting of solutions recently searched. The TS eliminates solutions found in the TABU list from $N_t(x)$. At times, the TABU list can also avoid solutions from taking certain types of moves (for example an item which was added to the bin that cannot be removed within the subsequent n moves). The chosen attributes in the recently visited solutions will be considered as “TABU-active”. The solutions consisting of TABU-active elements will be called the “TABU”. Such types of short-term memory are known as a “Recency-based” memory. The TABU list will contain attributes that are more effective in certain types of domains even though they may result in the creation of a new obstacle. If a particular attribute is noticeable to be a TABU, it may result in more than one of such solutions being a TABU. Certain solutions will have to be avoided and they may also be of great quality but not visited yet. For the purpose of mitigating this issue, “aspiration criteria” had been introduced: these will override the TABU state of a solution thus with the excluded solution in the set. The aspirations criterion used most often is accepting solutions that are better than the current ones.

D. Proposed Hybrid Genetic Algorithm (GA) – Tabu Search (TS) Algorithm (GA-Tabu)

Premature convergence is probably the primary issue in the GA and the weakness of the TS was its dependent on initial solutions and single-point modes of search. The GA is able to make available initial solutions of high quality for the TS and also has a fast speed of the search that is able to reward for the problem of speed of the TS. Furthermore, the TABU list is flexible and the TS aspiration criterion is able to help the GA in escaping from the local optima. For the proposed GA-TS method, the TS has been integrated with the GA as follows: a mutation operator based on the TS will replace its original operator as soon as the warning for prematurity is triggered. For judging if the process of search was trapped inside premature convergence, the index of prematurity was defined by computing the degree of similarity between both individuals as in (4):

$$I = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \frac{(N-2)!*2!}{N!} * \frac{W}{L} \tag{4}$$

Wherein, I means the rashness list, N signifies the genuine include of people in the populace, $N!/((N-2)!*2!)$ speaks to the check of blends of the pairwise people in the populace, Wrepresents the real include of same qualities in similar areas for each two people in the whole populace and Lsignifies the real length of chromosomes.

The means in the half breed GA and TS strategy proposed are [14]:

Stage 1: Fix the boundaries like populace size (popsize), the greatest emphasis of a half breed GA and TS (iter_GATS), the most extreme cycle of the TS (iter_TS), the hybrid likelihood (pc), the transformation likelihood (pm), length of the TABU rundown (lt).

Stage 2: Create the popsize populace. Unravel each person for getting wellness esteem and relate them to achieve an underlying arrangement x.

Stage 3: Set n=1.

Stage 4: Create the new populace for resulting ages. The hereditary advancement utilizing three administrators that incorporates choice, hybrid, lastly change will be applied for the formation of posterity for the resulting populace. On the off chance that chi, I = 1, 2… the popsize will be for the new populace. A roulette wheel methodology will be utilized for playing out a determination.

Stage 5: Presently every person in the new populace chk, k=1, 2… with a popsize as the TS beginning arrangement is utilized. Rehash stages 6 to 8 for improving individual quality until the emphasis approaches popsize. Presently continue to stage 9.

Stage 6: Set j=1.

Stage 7: Distinguish all areas of the TS and its present arrangement. Pick the best non-TABU neighborhood fulfilling the targetmeasure.

Stage 8: Presently update the TABU rundown to check end condition and if this (j = iter_TS) is met of if all the basic squares have not exactly an activity, stop the TS and the ideal arrangement. Individual chk will be subbed by the ideal arrangement. If not, j = j + 1 and now return to stage 7.

Stage 9: If the end condition (n = iter_GATS) is met, stop the half and half GA and TS technique and yield ideal arrangement. If not, n = n + 1 and now move back to stage 4.

The flowchart for proposed hybrid GA and TS method [15] as shown in Fig. 1.

E. Proposed Hybrid Invasive Weed Optimization (IWO)-Tabu Search (TS) Algorithm

There have been several attempts made for improvement strategies of the IWO and the TS for practical applications. There was a hybrid strategy that was in combination with the TS and the IWO which has been introduced for the generation of test data that satisfies the criterion of decision coverage. In the hybrid IWO-TS algorithm proposed, the IWO was applied for searching the entire solution space for ensuring diversity of population and also for the TS that is applied to a local search for avoiding convergence to the local optimum. For every iteration of the IWO-TS, it will not carry out any competitive exclusion operator once a spatial dispersion operation is conducted but will check if every weed has been enhanced by the seeds. The week is not enhanced inside the g_1 (g_1> 0) and successive iterations, it may remain stagnated and placed into the taboo list. The plants in the neighborhood of the tabooed weeds will have lower levels of fitness that has to be removed from the present population [16].

Let W denote the current weed set, S_i the new set of seeds that are produced by the weed W_i∈W in the current iteration, and TL denotes the taboo list. Now the current population of the plant may be shown as (5):

$$P = W \cup S_1 \cup S_2 \cup ..... \cup S_{|W|} \quad (5)$$

The plants to be removed using a tabooed solution T_i∈TL are shown as (6)

$$E_i = \{P_j \in P \mid \Delta(P_j, T_i) \le \delta_i ; F(P_j) < F(T_i)\} \quad (6)$$

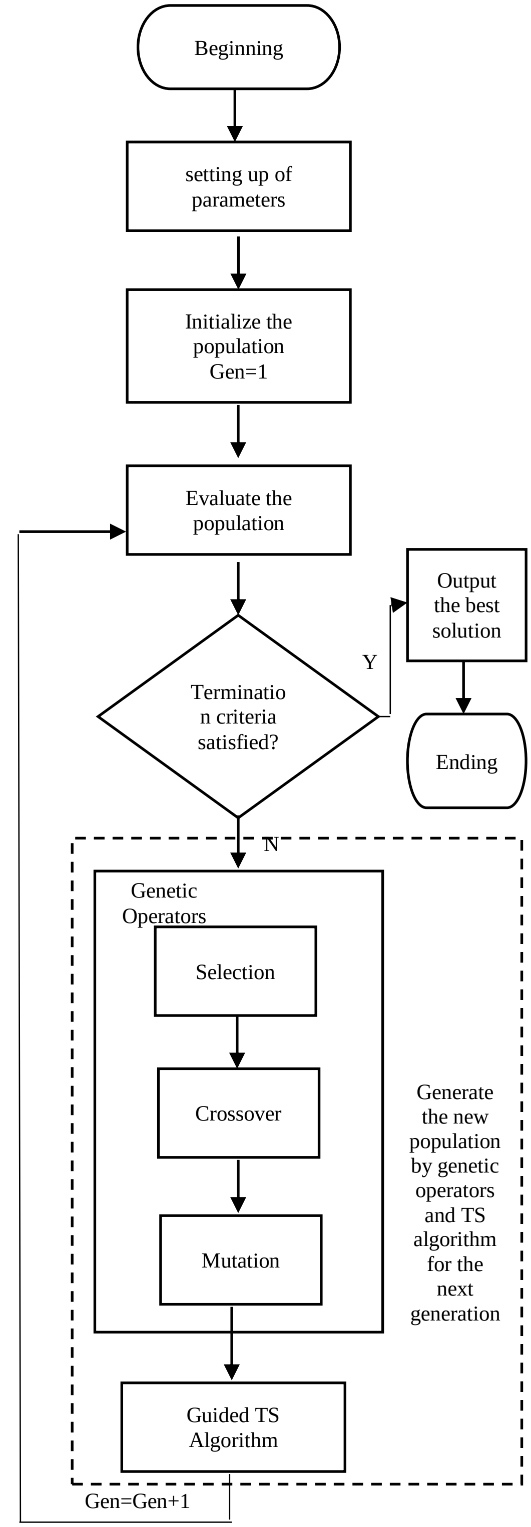

Fig. 1. Flowchart for the Proposed Hybrid GA-TS Algorithm.

Wherein, Δ (·,·) indicates the actual distance between two solutions, δ_i denotes a positive parameter to decide the size of its tabooed regions using solution T_i, and F (·) which denotes its fitness function.

As both IWO-TS focus on the continuous problems, Δ (·,·) may be definite to be the Euclidean distance. Parameter δ_i will have a significant level of influence on the IWO-TS and its performance. The higher the δ_i is, the more will be the plants that are deleted by the T_i. It can be challenging to choose a fixed value for the different δ_i for adapting to different problems. One more feasible way was to choose a value for every δ_i to the solution and its real size. This is discovered by the IWO-TS around its corresponding weed. This means, if the weed W_i∈W has been judged as tabooed (without any loss of generality it is assumed to correspond to T_i∈ TL), I δ will be set to (7):

$$\delta_i = \max\{\Delta(S_{ij}, W_i) \mid \forall S_{ij} \in S_i\} \quad (7)$$

When an elimination process of the plant P is complete, all remaining plants $P' \in P \mid (E_1 \cup E_2 \cup .... \cup E_{|TL|})$ will be sent to the operator of competitive exclusion. A TS approach will provide well-organized methods to handle this but it may not always be easy for the tabooed regions to have promising solutions. So, an aspiration criterion is introduced and the TL is set with a fixed length tl (tl> 0) to release the tabooed solutions. Once this is done, the neighborhood will further explore certain better solutions. In case no tabooed solution gets released, the IWO-TS will have to identify a better solution with the current weeds. Also, introducing the aspiration criterion can avoid an infinite increase to the TL thereby saving cost. As the IWO-TS works on local search, some seeds may be near the tabooed weeds and it may be redundant to do aremoval operation using a tabooed weed for all iteration. An easy way was to occasionally perform in each g_2 (g_2> 0) iterations to reduce computation cost.

For the purpose of enabling the exploration of a new solution by the IWO-TS a new self-production operator has to be defined. This will generate new weeds randomly and provide chances to explore undetected regions. The weeds generated can get excluded in the subsequent iteration by means of a competitive exclusion operator that can bring down its performance. So, the weeds randomly generated will be moved to the best weed found until now and the primary motive will be to move them to a region with promising solutions and is described mathematically in (8 and 9):

$$W^0_{new} = X_{\min} + rand(0,1).(X_{\max} - X_{\min}) \quad (8)$$

$$W_{new} = W^0_{new} + rand(0,1).(W_{gb} - W^0_{new}) \quad (9)$$

Wherein the X_{max} and X_{min} denote the upper bound and the lower bound on different variables that are optimized, W_{gb} denotes the best weed that is found so far, and the rand (0, 1) will return a random number that is uniformly distributed within the interval (0, 1). Since the IWO-TS have a strong ability of exploration in an early stage of the search, it will have to complete a fine local search in a later stage and it is important to keep changing the weeds that are created by this self-production operator in other iterations. The number has been altered as in (10):

$$n_{sp} = \left\lfloor (1-4(t/t_{\max}-0.5)^2)n_{\max} \times 20\% \right\rfloor \tag{10}$$

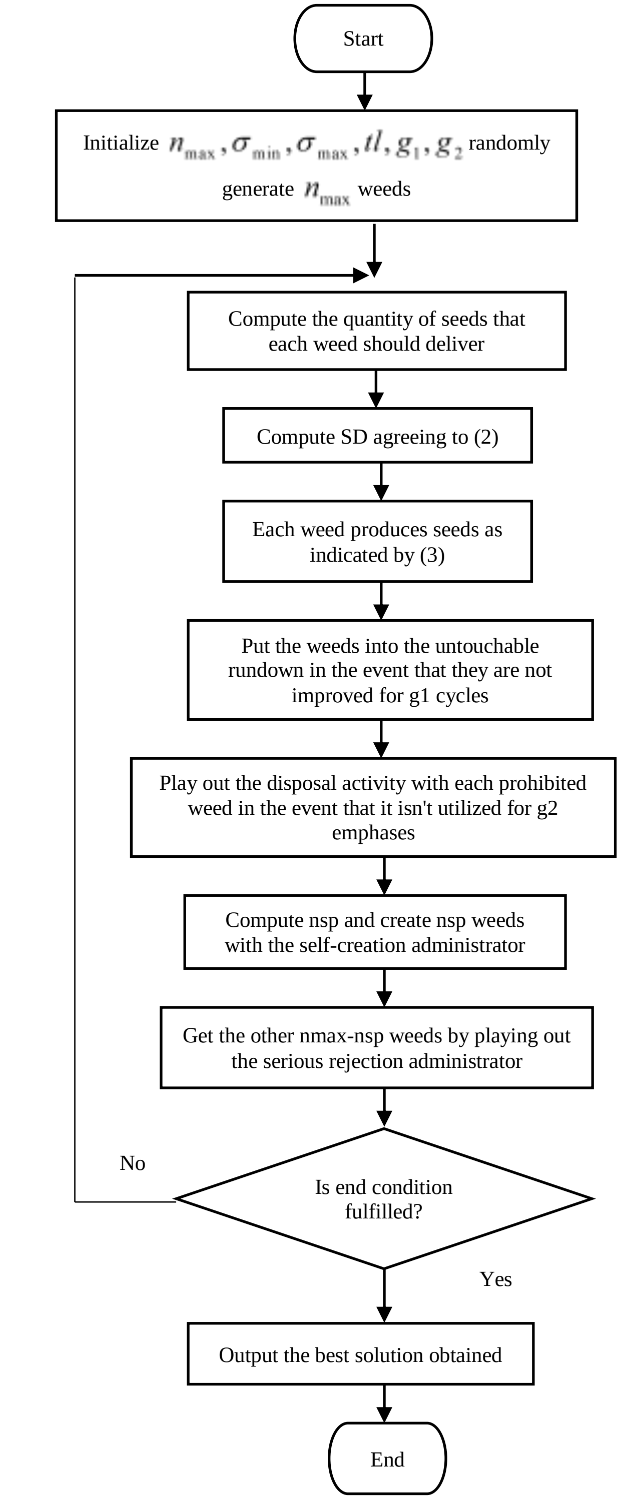
Start
Initialize $n_{\max}, \sigma_{\min}, \sigma_{\max}, tl, g_1, g_2$ randomly generate $n_{\max}$ weeds
Compute the quantity of seeds that each weed should deliver
Compute SD agreeing to (2)
Each weed produces seeds as indicated by (3)
Put the weeds into the untouchable rundown in the event that they are not improved for g1 cycles
Play out the disposal activity with each prohibited weed in the event that it isn't utilized for g2 emphases
Compute nsp and create nsp weeds with the self-creation administrator
Get the other nmax-nsp weeds by playing out the serious rejection administrator
No
Is end condition fulfilled?
Yes
Output the best solution obtained
End

Fig. 2. Flowchart for the Proposed Hybrid IWO-TS Algorithm.

Wherein, the notation $\lfloor \cdot \rfloor$ will be a round down operation. In accordance with (10), the n_{sp} will increase nonlinearly from zero till the search reaches the maximum value $\lfloor 0.2n_{\max} \rfloor$. After this, the n_{sp} will reduce gradually to zero. For every iteration, the IWO-TS will get another $n_{\max}$ – n_{sp} weeds from its plant set P' which is by carrying out on it a competitive exclusion operator.

The flowchart for hybrid IWO-TS as illustrated in Fig. 2.

IV. Results and Discussion

The GA, TS and IWO methods are used. Here, 5000 to 20000 number of tasks are considered. The guarantee ratio, energy savings and resource utilization are represented in Tables I to III and Fig. 3 to 5.

From the Fig. 3, it is seen that the IWO-Tabu has higher guarantee ratio by 3.24%, by same value, by 1.02%, by 2.06%, by same value, by same value and by same value for number of tasks 5000, 7500, 10000, 12500, 15000, 17500 and 20000 than GA-Tabu, respectively.

TABLE I. Guarantee Ratio % for IWO-Tabu

Number of Tasks	GA-Tabu	IWO-Tabu
5000	91	94
7500	98	98
10000	97	98
12500	96	98
15000	99	99
17500	99	99
20000	99	99

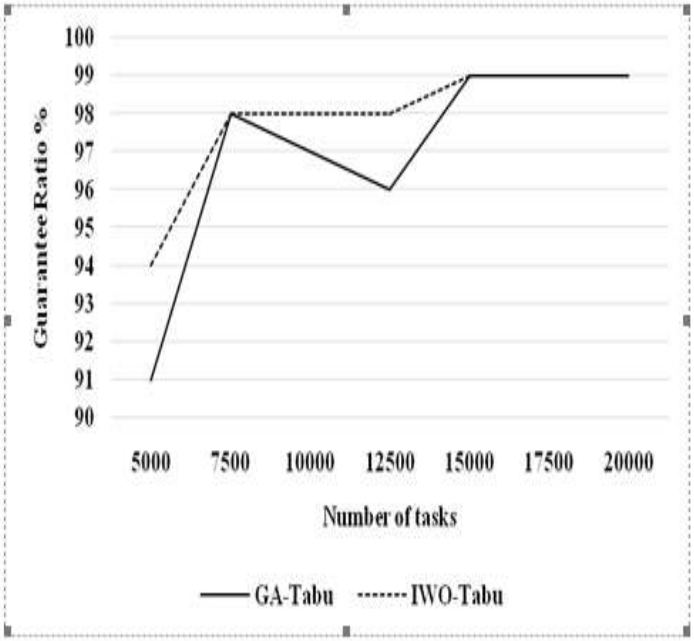

Fig. 3. Guarantee Ratio % for IWO-Tabu.

TABLE II. ENERGY SAVINGS % FOR IWO-TABU

Number of Tasks	GA-Tabu	IWO-Tabu
5000	3	3.3
7500	3.6	4.1
10000	3.3	4.4
12500	3.2	4.2
15000	3.6	4
17500	3.1	3.8
20000	2.8	3.3

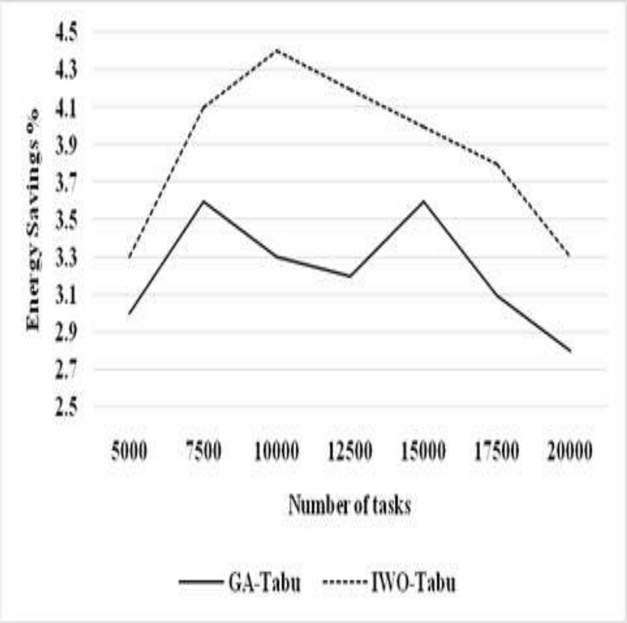

Fig. 4. Energy Savings % for IWO-Tabu.

From the Fig. 4, it is seen that the IWO-Tabu has higher energy savings by 9.52%, by 12.98%, by 28.57%, by 27.02%, by 10.52%, by 20.29%and by 16.39% for number of tasks 5000, 7500, 10000, 12500, 15000, 17500 and 20000 than GA-Tabu, respectively.

From the Fig. 5, it is seen that the IWO-Tabu has higher resource utilization by 4.58%, by 5.4%, by 3.59%, by 4.82%, by 3.51%, by 4.65%and by 6.97% for number of tasks 5000, 7500, 10000, 12500, 15000, 17500 and 20000 than GA-Tabu, respectively.

TABLE III. RESOURCE UTILIZATION % FOR IWO-TABU

Number of Tasks	**GA-Tabu**	**IWO-Tabu**
5000	64	67
7500	72	76
10000	82	85
12500	81	85
15000	84	87
17500	84	88
20000	83	89

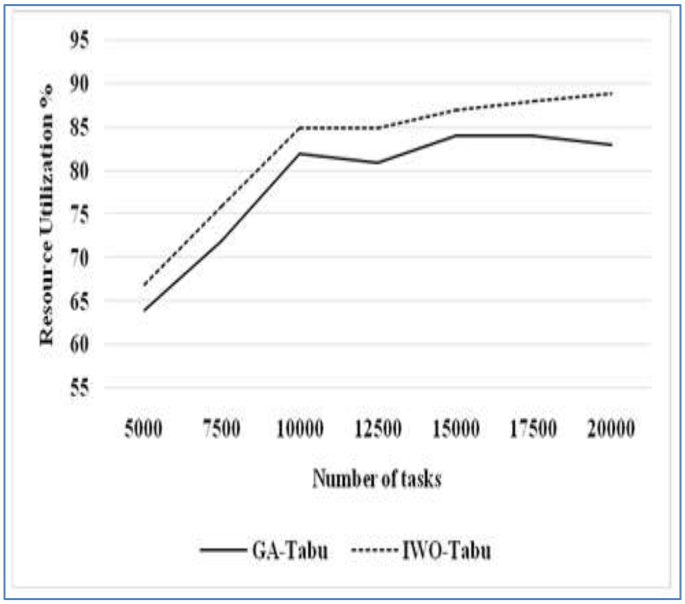

Fig. 5. Resource Utilization % for IWO-Tabu.

V. CONCLUSION

There is an ability to allocate limited resources for computing large tasks that have a goal of optimization inspiring wider solutions in the domain of cloud computing. To solve both algorithms like the GA or the TS (GA-TS) is used. Owing to the GA having the ability that handles complex search spaces that have a high chance of success to find optimal solutions. IWO is a metaheuristic that was recently developed there is a metaheuristic imitating the invasive weed behaviour. The reproduction and the spatial dispersal operators in the original IWO that make the seeds positioned in and around the best weed that can lead to premature convergence. For overcoming such drawbacks, the work proposed the IWO-TS algorithm using the core idea of the TS. If there is no better solution found, the IWO-TS will judge the weed that is stagnated and will taboo it avoiding any repeated search in the neighbourhood. Additionally, the IWO-TS will define another self-production operator to generate new weeds randomly as opposed to directly selecting it from the plant population in order to ensure new solutions are also explored.

REFERENCES

[1] Panda, S. K., & Jana, P. K. (2015). Efficient task scheduling algorithms for heterogeneous multi-cloud environment. The Journal of Supercomputing, 71(4), 1505-1533.

[2] Kashikolaei, S. M. G., Hosseinabadi, A. A. R., Saemi, B., Shareh, M. B., Sangaiah, A. K., &Bian, G. B. (2019). An enhancement of task scheduling in cloud computing based on imperialist competitive algorithm and firefly algorithm. The Journal of Supercomputing, 1-28.

[3] Sonawane, M. P. A., &Ragha, L. (2014). Hybrid genetic algorithm and TABU search algorithm to solve class time table scheduling problem. International Journal of Research Studies in Computer Science and Engineering, 1(4), 19-26.

[4] Madni, S. H. H., Latiff, M. S. A., Coulibaly, Y., &Abdulhamid, S. I. M. (2016). An appraisal of meta-heuristic resource allocation techniques for IaaS cloud. Indian Journal of Science and Technology, 9(4), 1-14.

[5] Sahoo, S., Sahoo, B., &Turuk, A. K. (2019). A Learning Automata-based Scheduling for Deadline Sensitive Task in The Cloud. IEEE Transactions on Services Computing.

[6] Li, C., Tang, J., & Luo, Y. (2018). Towards operational cost minimization for cloud bursting with deadline constraints in hybrid clouds. Cluster Computing, 21(4), 2013-2029.

[7] Shishido, H. Y., Estrella, J. C., Toledo, C. F. M., &Arantes, M. S. (2018). Genetic-based algorithms applied to a workflow scheduling algorithm with security and deadline constraints in clouds. Computers & Electrical Engineering, 69, 378-394.

[8] Liu, L., Fan, Q., & Buyya, R. (2018). A deadline-constrained multi-objective task scheduling algorithm in mobile cloud environments. IEEE Access, 6, 52982-52996.

[9] Tang, C., Xiao, S., Wei, X., Hao, M., & Chen, W. (2018, January). Energy Efficient and Deadline Satisfied Task Scheduling in Mobile Cloud Computing. In 2018 IEEE International Conference on Big Data and Smart Computing (BigComp) (pp. 198-205). IEEE.

[10] Haidri, R. A., Katti, C. P., & Saxena, P. C. (2019). Cost-effective deadline-aware stochastic scheduling strategy for workflow applications on virtual machines in cloud computing. Concurrency and Computation: Practice and Experience, 31(7), e5006.

[11] Singh, P., Dutta, M., & Aggarwal, N. (2017). A review of task scheduling based on meta-heuristics approach in cloud computing. Knowledge and Information Systems, 52(1), 1-51.

[12] Hosseini, Z., &Jafarian, A. (2016). A Hybrid Algorithm based on Invasive Weed Optimization and Particle Swarm Optimization for Global Optimization. International Journal Of Advanced Computer Science And Applications, 7(10), 295-303.

[13] Nasim, R., &Kassler, A. J. (2017, May). A robust Tabu Search heuristic for VM consolidation under demand uncertainty in virtualized datacenters. In 2017 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID) (pp. 170-180). IEEE.

[14] Zhang, L., Gao, L., & Li, X. (2013). A hybrid genetic algorithm and tabu search for a multi-objective dynamic job shop scheduling problem. International Journal of Production Research, 51(12), 3516-3531.

[15] Li, X., & Gao, L. (2016). An effective hybrid genetic algorithm and tabu search for flexible job shop scheduling problem. International Journal of Production Economics, 174, 93-110.

[16] Ren, Z., Chen, W., Zhang, A., & Zhang, C. (2013, July). Enhancing invasive weed optimization with taboo strategy. In Proceedings of the 15th annual conference companion on Genetic and evolutionary computation (pp. 1659-1662). ACM.

[17] Gopi, A., et al. "Designing an Adversarial Model Against Reactive and Proactive Routing Protocols in MANETS: A Comparative Performance Study." International Journal of Electrical & Computer Engineering (2088-8708) 5.5 (2015).

[18] Kumar, S. Ashok, et al. "An Empirical Critique of On-Demand Routing Protocols against Rushing Attack in MANET." International Journal of Electrical and Computer Engineering5.5 (2015).

[19] G. L. Sravanthi, M.Vasumathi Devi, K.Satya Sandeep, A.Naresh and A.Peda Gopi, “An Efficient Classifier using Machine Learning Technique for Individual Action Identification” International Journal of Advanced Computer Science and Applications(IJACSA), 11(6), 2020. http://dx.doi.org/10.14569/IJACSA.2020.0110664.