ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 5, Issue 4, April 2018

A Greedy Search Aware Fuzzy Scheduling in Cloud

[1] Sirisati Ranga Swamy, [2] Dr. Sridhar Mandapati
[1] Research Scholar, Dept of Computer Science Engineering, Acharya Nagarjuna University, Guntur, Andhra Pradesh, India
[2] Associate Professor, Dept of Computer Applications, RVR&JC College of Engineering, Guntur, Andhra Pradesh, India

Abstract: - **Cloud computing can provide on-demand storage as well as computing services that have a high performance level along with high scalability. The increasing consumption of energy in the cloud data centers is a very prominent problem today. The cloud performance has been affected because of the issues in security. Thereby, the service providers have been held responsible for taking proper care of these systems here and the scheduling will be held responsible for the choice of best and also most suitable resources in the task execution by means of taking certain types of static as well as dynamic parameters along with the restrictions of such tasks into consideration. The scheduling in that of cloud computing will belong to one category of the problems that are called the Non-deterministic Polynomial (NP)-hard based problems owing to the solution space and so it takes a long time in identifying an optimal solution. Here in this work, the fuzzy logic and the methods are greedy in terms of the methods that were proposed. For calculating the value of the fitness in the fuzzy inference system the membership function is used to determine the degree until which these parameters belonging to a fuzzy set which is relevant. The work also introduced another new idea for integrating the approaches in solving the hard problems that are combinatorial. This proposed methodology which shall evaluate all the objects in a manner that can combine the fuzzy reasoning along with a greedy mechanism. It also means that the fuzzy solution space is exploited with some greedy methods. The results of the experiment proved that this method proposed can achieve better performance than that of the fuzzy logic.**

Keywords: **Cloud Computing, Scheduling, Security, Fuzzy Logic and Greedy Search.**

I. INTRODUCTION

In the current scenario, Cloud computing has grown to become one new paradigm in computing that has a flexibility that is unparalleled and also has access to that of the shared and the storage in the digital world of today resulting in a highly significant growth in data centres in which the size grows from about 1000's to about a hundred thousand servers. These types of data centres in Cloud computing can offer Information Technology (IT) based resources as services and such types of various hardware systems like the servers or the data network systems with the storage and also representing all of their resources in a data centre that can provide the Infrastructure as a Service (the IaaS) along with that of the Platform as a Service (the PaaS). Such applications like the web search, the computation along with that of social networking are all being offered by the data centres of cloud computing which have been hosted to be the Software as a Service (SaaS). These applications will be run further using the virtualized IT resources which may also be Virtual Machines (VMs), that have been duly provided by the PaaS and the IaaS. Based on that of the requests that have been made by the services providers for all of such resources there are several other types of services which may be provided by the VMs to all their requests [1].

However, the various changes that are in the minimization of the approaches will not take into consideration the cloud services and the charges depend on the minutes or the instance hours. An integral instance hour will increase the difficulty in solving the problem of cost minimization. An auto scaling and scheduling algorithm is the one that will bring down the cost by means of considering the integral instance hours. In this algorithm, tasks of local deadlines are assigned and these decide number as well as the types of the VMs that are needed for executing such an application [2].

The consumption of energy of such data centres will constitute a major part of the operation cost. This energy consumed by the large data centres [3] result in the increased energy demand being a hurdle to the scalability of the data centre. There is an Emerson report that has made an estimate of the servers in such data centres that can account for about 52% of the entire energy consumed, and the cooling systems of about 38%, with the other miscellaneous systems, like power distribution, that tend to account for what is remaining which is around 10%. These are the three different sub-systems in that of a data centre that can be optimized for the efficiency of its energy [3]. There are however certain concerns to the security that may also relate to all of the areas of risk and of the external storage along with the dependency based on which there is a public based internet, its multi-tenancy and also the lack of integration or control with that of the security. While being compared to all of these traditional technologies, the cloud has many more specific features like resources of large scale that belongs to the providers which have been completely distributed as they are heterogeneous along with being fully virtualized. Such security mechanisms used traditionally like identity or authentication along with the authorization will not be sufficient for the clouds that are seen in their current form. However, 0wing to the very fact that the models of cloud service that have been employed, such operational models along with that of the technologies, are all being used for the purpose of enabling the cloud. However, such types of integration of security using such solutions are being perceived as making them much more rigid than earlier [4].

Scheduling normally permits some optimal allocation of the resources among various given tasks within a finite time for achieving the desired Quality of Service (QoS). Formally, such types of scheduling problems will involve tasks that are scheduled to certain constraints for being optimized and in general conditions the actual problem of the mapping of the tasks on the unlimited resources of computing among cloud computing will particularly belong to another different category known as the NP-hard problems. There have been no algorithms that can produce any optimal solution within that of a polynomial time for these types of problems. The solutions that are based on such exhaustive search may not be feasible as their operating cost in the generating of schedules is found to be very high. The Meta-heuristic based techniques tend to deal with such problems by means of providing near optimal solutions inside a reasonable time frame. The Meta-heuristics have also gained huge popularity in the last few years because of their efficiency and also their effectiveness in solving large as well as complex problems [5]. Here in this work, this greedy search aware fuzzy has been proposed for the cloud scheduling. The rest of the investigation has been organized into all the following sections. The Section 2 discusses all the related work in literature. The Section 3 explains several other methods that are used in this work. Section 4 has discussed the experimental results and the conclusion is made in Section 5.

II. RELATED WORKS

Ma et al., [6] further proposed yet another Dynamic Greedy Strategy (DGS) being feasible and also flexible as its dynamic task scheduling scheme, that was able to dynamically allocate all of the virtual resources for executing their computing tasks and also promptly completing all types of scheduling as well as the process of execution by means of using an improved greedy strategy. This simulation platform Cloud Sim has been expanded to realize that the proposed scheme as well as their simulation results that show that a DGS can speed up tasks' and their completion time and further improve the cloud resource utilization for achieving load balance

Wang et al., [7] had further proposed one more new Adaptive Genetic Algorithm (AGA), that had been found to be yet another algorithm employed in scheduling which was based on the double-fitness adaptive algorithm-Job spanning time and also a Load balancing Genetic Algorithm (the JLGA). This type of a strategy works on the basis of the tasks included in scheduling and the ones that have shorter jobs wherein the average job make span will only be able to satisfy its balancing inter-node. Navimipour and Milani [8] presented another new and evolutionary algorithm which was known as the Cuckoo Search Algorithm (the CSA) that was used for scheduling of the tasks which were present in the cloud computing. Such a CSA based algorithm will be based upon the behaviour of the obligate brood parasites that belong to a particular type of cuckoos which are in combination of that of the Lévy flight behaviour that is seen in certain birds and also in some fruit flies. The simulation results have been demonstrated while the value of that of a Pa will be recorded to be extremely low, the coverage and the actual speed of this algorithm will be quite high. Such type of a system structure can possibly identify all such flow of fuzzy logic inference that will be derived for that of their input variables which will be for the output variables. For this, a fuzzification that is in the input interface can duly translate all of these analog inputs to be inside that of their fuzzy rules so that all such fuzzy inferences will be completed only within the rule blocks which may contain the linguistic control based rules. The final output of all of such rule blocks have also be the linguistic variables and the defuzzification which is in the output interfaces that will be able to translate them into analog variables.

Bianchi and Presti [9] further presented yet another two-stage Virtual Network Embedding (VNE) algorithm, that was capable of mapping the first virtual nodes for the substrate nodes that has been based on that of a suitable ranking algorithm and then also map link along with that of the shortest path that can exist among such nodes. The main ingredient in this approach will be the novel algorithm of ranking which is known as the Markov Chains with Rewards Ranking (MCRR), further based on the Markov reward processes, that are all associated with that of the actual local resources captured within the vicinity fo a certain node. All these experiments have shown that this algorithm will be able to outperform the other earlier approaches as regards the lower rate of VNE rejection, better utilization of resources and much higher revenues.

Keeping in view this min-min algorithm that has shown preference to the scheduling of all the small tasks and the max-min algorithm to their scheduling of big tasks that will lead to the problem of any type of load imbalance in that of cloud computing, with another new algorithm called min-max that has been proposed by Zhou and Zhigang [10]. This Min-max has made good use of the actual time of greedy strategy, the small tasks as well as the big tasks that have been put together for the purpose of scheduling and also for solving problems in load imbalance.

III. METHODOLOGHY

Here in this section there is fuzzy logic as well as greedy search based algorithm of fuzzy scheduling that have been discussed.

A. Fuzzy Logic

This Fuzzy logic can provide a very simple way of arriving at a proper conclusion which has been based on a vague that is based ambiguous noisy and imprecise input information. This Fuzzy logic had been proposed by Zadeh in the year 1965 and has been used widely today as it has been dealing with various imprecise as well as incorrect information. It has been found to be extremely close to the human mind and for fuzzy logic a human expertise will be embedded within this system. The Fuzzy logic further applies to all the various fields like the air conditioners, the microcontrollers, the washing machine, the microprocessor, and image processing in other applications. These Fuzzy inferences are widely used for solving the problems of reasoning in all the environments that are not certain. One such typical fuzzy inference module contains three major components that are [11]: the Inference Engine: which will defines all of the fuzzy logic operators with that of the defuzzifier that is used in a process of inference. The Membership Functions: this is the one that defines the degree to which all the fuzzy elements will be part of the fuzzy set. It further maps all the crisp values to a degree of membership among 0 to 1. In case of these fuzzy inference system, every such input as well as the output variable will have its own function set of membership. The Rule base: this will denote a set of the “If-Then” rules that can define all the ones within the inference model and a rule structure will be: “If antecedent Then consequent”, in which the antecedent and consequent one will denote the fuzzy propositions that have been connected by the “AND” or the “OR” operators.

This particular system structure will identify all the fuzzy logic inference flow that is from the input variables to that of the output variables. Fuzzification in an input interface will translate the analog inputs within their fuzzy values and such fuzzy inference will take place only in the rule blocks that consist of the linguistic control based rules. The output of all such rule blocks will be the linguistic variables and a defuzzification in that of output interfaces will translate them into the analog variables.

B. Greedy Search Algorithm

This type of a greedy search algorithm is the one that uses the heuristics for making the right choice and when there is backtracking here the choices are re-evaluated. These Greedy algorithms are often the first ones that are considered for the problems in optimization and also for that of the maximization of problems in which there is a general form as shown below: there is also another attempt made for building one more ideal solution which will be by the execution of all these steps until such time no other item is left which can be considered for the same purpose: (1) the standard of selection: this means in case of a greedy way it will choose and also further consider this item that has been found as an optimal one based on a certain criterion that is in its current stage; (2) the condition of feasibility: this is taken into consideration only for one item and will be accepted only on satisfying one more set of conditions with yet another feasible solution with the tasks being accepted inside the problem or its constraints. Such a criterion for the selection is related to all the many different constraints as well as their objective functions and are normally found to be the ratio of an ’advantage’ to ’cost’, which will be able to measure the item and its efficiency. For such a problem the prime constraint will be from the capacity of this problem to hold on to this approach and such a constraint will only be the actual value of this task and its demand [12].

C.Proposed Greedy Search Aware Fuzzy Scheduling

The actual concept of thus fuzzy greedy evaluation which is for problems involving combinatorial optimization that has been presented in the year 2005 by Sheibani and for such greedy methods that has been a significant question and also an ideal choice in a particular stage of this algorithm. The greedy algorithm will make some choice that will look with no such consideration to that of the earlier choices and also their future consequences. Therefore, it will be aware of the choices that will be the best and this may also be the worst ones as well in terms of being a solution. Such a fuzzy set theory allows us to duly represent all the vague concepts that may be expressed in one such natural knowledge which can be a choice that is made for all these greedy methods [13].

IFERP connecting engineers... developing research

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 5, Issue 4, April 2018

Consider a set of combinatorial optimisation problems where each is associated with on discrete solution space being X, and a feasible space S

That has a property $S \subseteq X$ that has been defined by a problem constraint and also an objective function $f : X \mapsto \Re$.

For situations of minimization the main aim will be to find a solution which is feasible $x^* \in S$,

So that $f(x^*) \leq f(x), \ \forall x \in S,$

In which, $x = (x_1, \ldots, x_n)$ denotes a vector of the decision variables (solution).

Another cost function c (x), $\forall x \in \{x_1, \ldots, x_n\}$ defined for every problem referring to the function c (x) as its greedy evaluation function denoting the priority for the incorporating of the element x into a solution being constructed without resulting in infeasibility.

It will now describe another alternative approach and treat a set X as its fuzzy set having a well-defined membership function $\mu(x)$, and its form as shown in equation (1).

$$\mu(x) = \frac{1}{1 + \lambda^2 \rho \left(\left(\frac{1-\lambda}{\lambda} \right) x - \theta \right)^2} \quad (1)$$

Here the variable $x \in X$ will correspond to one of such variables within the definition of a problem of combinatorial optimisation; in this it consider $x \in \Re$.

This parameter θ will be a basic measure for the evaluation of the priority that has to be assigned to that of a variable x; and it will require $\theta \in \Re$.

This parameter λ will be a tuning parameter selected by the experimentation so that the $0 \leq \lambda < 1$. The parameter will play a critical role in the developed algorithm to be considered later and a parameter ρ > 0 will be effectively in the shape parameter, so that values of ρ will increase, and the graph of the $\mu(x)$ will get narrower.

This proposed function of evaluation

$\mu : X \mapsto (0.1]$,

will have the properties below:

$\mu(\lambda\theta / (1-\lambda)) = 1$ and $\mu(x) < 1$ for all $x \neq \lambda\theta / (1-\lambda)$.

Equation (1) will be a modification of the general formulas of the families of the other fuzzy membership functions referring to the greedy evaluation replacing the role of the function of greedy evaluation to determine the priority that is assigned to the element x. This Fuzzy Greedy Search Algorithm (FGSA) [14] is actually an extension of that of the concept of fuzzy greedy evaluation which is in the form of one metaheuristic. The process begins with the initial population that is generated for the phase of construction of the Greedy Randomized Adaptive Search Procedure (GRASP). This algorithm will work based on that set of individuals (called the population) that has been divided into two sub-sets that are generated where one is through a recombination operator with a standard approach and the other by employing a procedure for construction similar to the GRASP construction phase except where the process adopts another version of the evaluation as opposed to the greedy evaluation function.

IV. RESULTS AND DISCUSSION

In this section, the fuzzy and greedy fuzzy methods are used. The makespan, degree of imbalance and resource utilization as shown in tables 1 to 3 and figures 1 to 3.

TABLE I.
MAKESPAN FOR GREEDY FUZZY

Number of Jobs Fuzzy Greedy Fuzzy

Number of Jobs	Fuzzy	Greedy Fuzzy
200	44	42
400	92	88
600	147	141
800	193	187
1000	247	236

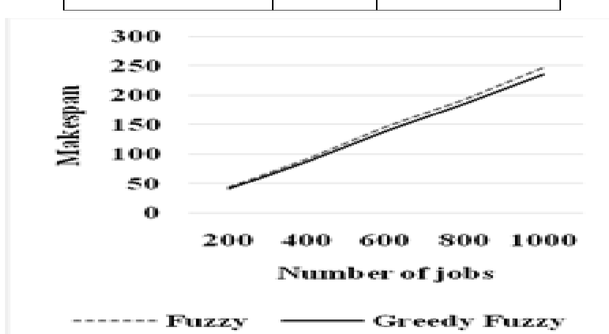

Fig.1. Makespan for Greedy Fuzzy

From the figure 1, it can be observed that the greedy fuzzy has lower makespan by 4.65% for 200 number of jobs, by 4.44% for 400 number of jobs, by 4.16% for 600 number of jobs, by 3.15% for 800 number of jobs and by 4.55% for 1000 number of jobs when compared with fuzzy.

TABLE II.
DEGREE OF IMBALANCE FOR GREEDY FUZZY

Number of Jobs	Fuzzy	Greedy Fuzzy
200	2.5	2.4
400	2.4	2.3
600	2.1	2
800	2	1.9
1000	1.9	1.8

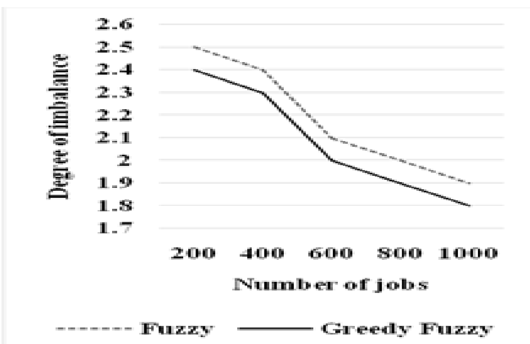

Fig.2. Degree of Imbalance for Greedy Fuzzy

From the figure 2, it can be observed that the greedy fuzzy has lower degree of imbalance by 4.08% for 200 number of jobs, by 4.25% for 400 number of jobs, by 4.87% for 600 number of jobs, by 5.12% for 800 number of jobs and by 5.4% for 1000 number of jobs when compared with fuzzy.

TABLE III.
RESOURCE UTILIZATION FOR GREEDY FUZY

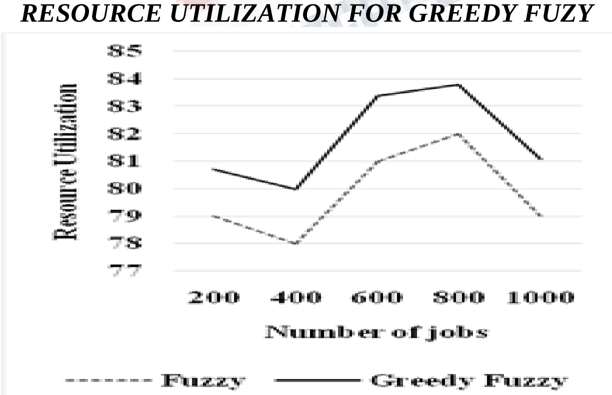

Fig.3. Resource Utilization for Greedy Fuzzy

From the figure 3, it can be observed that the greedy fuzzy has higher resource utilization by 2.12% for 200 number of jobs, by 2.53% for 400 number of jobs, by 2.91% for 600 number of jobs, by 2.17% for 800 number of jobs and by 2.62% for 1000 number of jobs when compared with fuzzy.

V. CONCLUSION

This has large prospects but also has some threats that are embedded. The work uses the method of greedy search aware fuzzy scheduling and it makes the best choice for this. This concept is integrated within the approaches for the problems that are hard and combinatorial. The method further evaluated the manner in which the fuzzy reasoning is combined with the greedy mechanism and given that these mechanisms are inexpensive compared to the other sophisticated methods for the optimization there is some practical significance to this. The efficiency has been demonstrated based on that of the hard combinatorial optimization problems that are in operational research as well as the management science. The results have proved that this greedy fuzzy has a much higher resource utilization by about 2.12% for the 200 number of the jobs, by about 2.53% for the 400 number of the jobs, by about 2.91% for the 600 number of the jobs, by about 2.17% for the 800 number of jobs and finally by about t2.62% for the 1000 number of the jobs on being compared to fuzzy.

REFERENCES

[1] Dong, Z., Liu, N., & Rojas-Cessa, R. (2015). Greedy scheduling f tasks with time constraints for energy-efficient cloud-computing data centers. Journal of Cloud Computing, 4(1), 5.

[2] Mao, M., & Humphrey, M. (2011, November). Auto-scaling to minimize cost and meet application deadlines in cloud workflows. In Proceedings of 2011 International Conference for High Performance Computing, Networking, Storage and Analysis (p. 49). ACM.

[3] Judge, J., Pouchet, J., Ekbote, A., & Dixit, S. (2008). Reducing data center energy consumption. ASHRAE Journal, 50(11), 14.

[4] Hashizume, K., Rosado, D. G., Fernández-Medina, E., & Fernandez, E. B. (2013). An analysis of security issues for cloud computing. Journal of internet services and applications, 4(1), 5.

[5] Kalra, M., & Singh, S. (2015). A review of metaheuristic scheduling techniques in cloud computing. Egyptian informatics journal, 16(3), 275-295.

[6] Ma, L., Lu, Y., Zhang, F., & Sun, S. (2012, May). Dynamic task scheduling in cloud computing based on greedy strategy. In International Conference on Trustworthy Computing and Services (pp. 156-162). Springer, Berlin, Heidelberg.

[7] Wang, T., Liu, Z., Chen, Y., Xu, Y., & Dai, X. (2014, August). Load balancing task scheduling based on genetic algorithm in cloud computing. In Dependable, Autonomic and Secure Computing (DASC), 2014 IEEE 12th International Conference on (pp. 146-152). IEEE.

[8] Navimipour, N. J., & Milani, F. S. (2015). Task scheduling in the cloud computing based on the cuckoo search algorithm. International Journal of Modeling and optimization, 5(1), 44.

[9] Bianchi, F., & Presti, F. L. (2016, September). A markov reward model based greedy heuristic for the virtual network embedding problem. In Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS), 2016 IEEE 24th International Symposium on (pp. 373-378). IEEE.

[10] Zhou, Z., & Zhigang, H. (2014). Task scheduling algorithm based on greedy strategy in cloud computing. The open Cybernetics & Systemics Journal, 8(1), 111-114.

[11] Qu, C., & Buyya, R. (2014, May). A cloud trust evaluation system using hierarchical fuzzy inference system for service selection. In Advanced information networking and applications (aina), 2014 ieee 28th international conference on (pp. 850-857). IEEE.

[12] Loiseau, P., & Wu, X. (2015). Greedy and Dynamic Programming Algorithms for Scheduling Deadline-Sensitive Parallel Tasks. Technical report, available as.

[13] Sheibani, K. (2011). A hybrid metaheuristic using fuzzy greedy search operator for combinatorial optimization with specific reference to the travelling salesman problem. Iranian Journal of operations Research, 2 (2), 63-71.

[14] Sheibani, K. (2006). Fuzzy Greedy Search and Job-Shop Problem. In 25th Workshop of the UK Planning and Scheduling Special Interest Group (PlanSIG). Nottingham, UK (pp. 147-150).