

RESPONSIBILITY LENIENCE EXPENDING APTNESS SYSTEM TRENDY FOG TOTALING

¹ Ushaben Barad

¹Gokul Global University, Sidhpur, Gujarat, India ¹ Asst. professor

Abstract: With the immense growth of internet and its users, Cloud computing, with its incredible possibilities in ease, Quality of service and on-interest administrations, has turned into a guaranteeing figuring stage for both business and non-business computation customers.. The dynamic environment of cloud results in various unexpected faults and failures.as the number of request increases over the datacenter then the failure probability load is also increases .so the requested task should be balance so there is minimum failure occur over the datacenter.so to overcome this issue ,we proposed a adaptive genetic algorithm using the levenberge marquardt algorithm(LM) which overcome the difficulties that arise due to fault and existing implemented algorithm .the paper analyze the fault on different parameter in term of request count and request failed.

Index Terms – cloud computing ,fault tolerance ,LM equation, adaptive genetic algorithm.

I. INTRODUCTION

The word cloud computing is the general term for the delivery of hosted services over the internet.it is more promising for both client or consumer and provider in different field like education ,marketing, engineering, business , industries and in so many field .cloud computing make the long held dream of utility computing possible with various properties like scalability ,multi-tenancy ,loose coupling , reliable delivery , easier management ,low cost, backup and recovery ,increased storage space, automatic updating ,versatile compatibility and so on[1] . but this is the one side of the coin as every coin has another side that one in cloud as in form of failure and management of that failure is one of the major issue in cloud.

Fault tolerance enable the system to continue operation possibly at reduced level rather than failing completely when some system behave malfunction ally unexpectedly. It not only manage the subcomponent of the system but also manage the QoS by maintain high load on data center because high load datacenter has high failure probability to compute the task request which is not good for the user as well as the service provider. So there is need of new approach that have sufficient strategy to increase the reliability by completing maximum task over the datacenter and minimum task fail by allocating task over the resources with low failure probability on virtual machine to provide high QoS to users.

There are many type of fault occur over the system like transient error which is temporary error that like to disappear soon ,intermittent error that randomly occur over the sub is crash failure in which system component stop functioning or remain inactive which is one of the major reason of the fault ,all these error can reduce the reliability rate of the datacenter and task completion rate .[2]

reactive fault tolerance are used to reduce the impact of failure on a system when failure occur i.e. as checkpoint ,replication ,job migration, retry , task resubmission and rescue workflow etc.[1]

This paper focus on the fault tolerance base on method to increase the accuracy of the system by minimizing the fault probability and task completion time using fitness algorithm which will overcome the gaps of previously implemented algorithms provide reduce in fault probability by proposing the algorithm base on adaptive learning to find the appropriate solution which cannot be achieved using static algorithm .

II. LITERATURE REVIEW

Author JI WANG, WEIDONG BAO, XIAOMIN ZHU, LAURENCE T. YANG, AND YANG XIANG, proposed an approach of FESTAL[3] which focuses on the reliability and availability of cloud for real time application in safety critical nature . the technique used is resource scaling up and resource scaling down to copies the task to some idle system and eliminate the long idle system for better utilization. author also incorporate the use of primary and backup copies in case of failure and BB overlapping technique . this technique provide better performance ratio of task over host time and gain ratio but it do not consider the task dependency for the particular system.

FASTER also incorporate the backward shifting method to make full use of the idle resources and incorporates task overlapping and VM migration for high resource utilization and vertical/horizontal scaling-up technique to quickly provision resources for a burst of workflows but it has disadvantage of not take consideration of predicting the fault .

MOHIT KUMAR GOKHROO, MAHESH CHANDRA GOVIL, EMMANUEL S. PILLI.[5] Suggest new approach in fault tolerance for detection and mitigation both in cloud computing .the approach base ON the method of detecting the fault on running status of the job. The detection algorithm periodically monitors the progress of job and reports the stalled job due to failed VM to fault tolerant manager (FTM). The prediction method is a proactive measures to detect abnormal condition before it actually occur .the advantages of this approach is in low resource wastage and avoid the penalty of service level agreement .

Author present the A working model of the strategy and a mathematical relationship to represent the fault tolerance model system using the concept virtualization and checkpoint/replay scheme. the checkpoint/replay model is developed using the reward renewal process (RRP) theory a backward recovery is performed and the Vm is immediately restarted and recovered from the last successful checkpoint.

It propose a dynamic resource allocating mechanism with fault tolerance to improve resource utilization. It incorporate a backup overlapping mechanism and efficient VM migration strategy for designing novel Dynamic Fault Tolerant Scheduling Mechanism for Real Time Tasks in cloud computing.

PUNEET GUPTA AND S P GHRERA[8] suggest the fault tolerance mechanism base on the load over the datacenter as the request over the datacenter is increases which lead to task request.

There are various fault tolerance strategies which are proposed by authors that also play a crucial role in fault tolerance mechanism such as Y. ZHANG, Z. ZHENG, and M. R. LYU proposed BFTCloud [9] for voluntary resource cloud computing The architectures operate on five basic operations: primary node selection, replica selection, request execution, primary node updating, and replica updating. The primary node is selected based on QoS requirements. The request for the service is handled by the primary node. G. FAN, H. YU, L. CHEN, and D. LIU. Propose CFN[10] for service resources, cloud module, the detection and failure process ... etc. P. M. KHILAR. suggested VFT[11] a reactive fault tolerant technique; it consists of a Cloud Manager (CM) module and a Decision Maker (DM) which are used to manage the virtualization, load balancing and to handle the faults. The virtualization includes a fault hander. Fault handler finds these unrecoverable faulty nodes and It also helps to remove the temporary software . G. CHEN" H. JIN, D. ZOU, B. B. ZHOU, W. QIANG, and G. HU.

III. OUR CONTRIBUTION

In the survey I conclude that in existing approaches are either discuss about task scheduling and resource management and utilization and various proactive and reactive method to decrease the fault over the system but the existing algorithm do not consider the factor or parameter that responsible for the software fail.

The problem with these approaches are that they do not consider the load over the datacenter or Vm to cope with the request task and efficient time to complete that task, which we conclude that as number of request over the data center increases which increase the load and failure probability over the data center and decrease A fault management system should match the task to the right Vm with all resources so the chances of occurrence of fault is less.

So to overcome this issues we propose an efficient fault management system by considering the different factors such as the request failed count and request complete count by the Vm to complete the task. Also applying a priority base genetic algorithm using the Levenberg Marquardt [14] fitness equation to decrease the fault probability rate .

IV. PROPOSED METHODOLOGY

4.1 Proposed Solution

The proposed method is to incorporate the fault aware learning based resource allocation algorithm by proposing the priority base genetic algorithm .the priority base genetic algorithm helps to find the solution which cannot be achieved by any static or dynamic algorithm which help in learning the previous past probability rate to reduce the future fault rate and it also help to find the fittest solution .the priority base genetic algorithm help in choosing the priority between the task completion or failure probability which take consideration the service level agreement.

The proposed algorithm incorporate LM (Levenberg Marquardt) fitness equation to find the fittest solution and also to minimize the machine fluctuation by different task size and waiting time based on linear equation .Algorithm is uses the Poisson distribution for random request failure at virtual machine such as host and datacenter.

4.2 Proposed Model

It is frequently used to solve optimization problems.

It is an optimization method based on population and is based on Darwin's theory of evolution . An initial population is taken randomly and it is used as a starting point. A fitness function is calculated for each chromosome so that it is known whether the chromosome is suitable or not.

4.3 Phases Of Proposed Algorithm

4.3.1 Initialization

Here we initialize asset of sequences or schedules allocated randomly, each sequences act a chromosomes for genetic algorithm. The complete set of chromosomes is said to be a population, acting as a input for algorithm.

4.3.2 Selection

In this phase we evaluate the fitness value for each set of sequence or chromosome, which depends up on the computing capability, total time taken to complete the schedule and the failure probability of complete schedule. Where

Since faults over a datacenter are random in nature and follows Poisson distribution, which over a period of time t and $t + \Delta T$ can be defined as in equation (4.1) and (4.2) where FPI defined the fault probability over the Host I and λ is fault rate over a time T

$$FPI(t \leq T <= t + \Delta | T > t) = \frac{\exp(-\lambda t) - \exp(-\lambda(t + \Delta T))}{\exp(-\lambda t)} \quad (4.1)$$

$$FPI(t) = 1 - \exp(-\lambda \Delta t) \quad (4.2)$$

the levenberg-marquardt (lm)[14] algorithm is the most widely used optimization algorithm. it outperforms simple gradient descent and other conjugate gradient methods in a wide variety of problems in equation(4.3) where Δm_j =free request queue for server j and Q_L is queue load of threshold that define threshold of virtual machine using CPU, RAM and bandwidth and fitness equation(4.4) and (4.5) in f_j in which λ_j =service rate FR_j =fault rate and W_j is waiting time for a request at server j

$$\Delta m_j = Q_L_{\text{threshold}} - Q_L_{\text{current}} \quad (4.3)$$

$$F_j = \alpha_1 \Delta m_j + \alpha_2 \lambda_j + \alpha_3 \left(\frac{1}{FR_j}\right) + \alpha_4 \left(\frac{1}{W_j}\right) \quad (4.4)$$

$$\alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 = 1 \quad (4.5)$$

4.3.3 Crossover

In this step two fittest solutions based on least make span and failure probability is selected. We have used multi point crossover to generate new fittest sequences/ chromosome. Steps to generate crossover are as follows.

- The two fittest chromosomes are selected
- A new fittest chromosome is generated using multi point cross over by interchanging the set of two chromosomes.
- The new chromosome replaces the chromosome with highest fitness value.

4.3.4 Mutation

In this phase new merging the new offspring, which can be better solution with remaining which, regenerates population keeps the total population size constant after each iteration. After specific count of iteration predefined as an input to genetic algorithm, best chromosome is selected i.e. the chromosome with least fitness value is selected for schedule.

4.4 Proposed Algorithm

FGDATA(VM list VM_i , Task list T_i , population size Po , iteration itr)

1. $VM_i = VM_List()$;
2. $FI = getFault()$;
3. $I = \text{no of VM}$
4. $T_i = Task_List()$;
5. $C = Gentic_algo(VM_i, t_i, Po, itr)$;
6. $Allocate_Resource(C)$
7. End

Figure 1: Proposed Algorithm Initialization

Genetic_algo(VM_i , T_i , P_0 , ltr)

1. $P_0 = \text{Initiate_Population}(T_i);$
2. $\text{Evaluation}();$
3. $C1 = \text{getFittest1}();$
4. $C2 = \text{getFittest2}();$
5. $\text{Crossover}(C1, C2)$
6. $\text{Mutation}(P_0, C1, C2);$
7. $\text{Return}(\text{getFittest}());$
8. End

Figure 2: Proposed Algorithm Genetic Algorithm

1. $\text{Evalutin}() \{$
2. For each C_i $i=0-P_0$
3. For each T_i
4. $\text{Temp_F}_j = \alpha_1 \Delta m_j + \alpha_2 \lambda_j + \alpha_3 \left(\frac{1}{FR_j}\right) + \alpha_4 \left(\frac{1}{W_j}\right)$
5. $\text{Fitness}_i = \text{Fitness}_i + \text{temp}$
6. End
7. END
8. $\}$

Figure 3: Proposed Algorithm Evaluation Phase

1. $\text{Allocate_Resource}(c)\{$
2. $C_i = \text{Gethromosome}();$
3. for each C_i
4. $\text{Allocate}(C_i);$
5. End
6. $\}$

Figure 4: Proposed Algorithm Allocation Phase

Proposed algorithm provides benefit over existing genetic algorithm that search the best solution from the past data over the datacenter and also the proposed algorithm takes consideration the faulty behavior of the cloud by considering the parameter like fault rate, waiting time for request at server, service rate and threshold value of the virtual machine. All these various parameters help to find the high utilization and getting least failure rate by completing maximum task request.

V. IMPEMENTATION

5.1 Software specification

For the implementation of our proposed algorithm , we will using CloudSim 3.0[1] which is the event driven simulator and implementing code using JAVA language in IDE eclipse which is the most powerful object oriented programming language used in CloudSim, CloudSim module can be easily extendable with user's requirement .Proposed algorithm is being tested over the various test case with 10 server ,2 GB RAM ,CORE_QUAD,2- HOST and requesting poisson distribution model for random request and fault model in distributed environment.

5.2 Result

Figure shows the improvement in number of request completed vs total number of request based on ratio of given data. And the next figure shows the increase request size completed over the total size of request in existing algorithm and proposed algorithm.

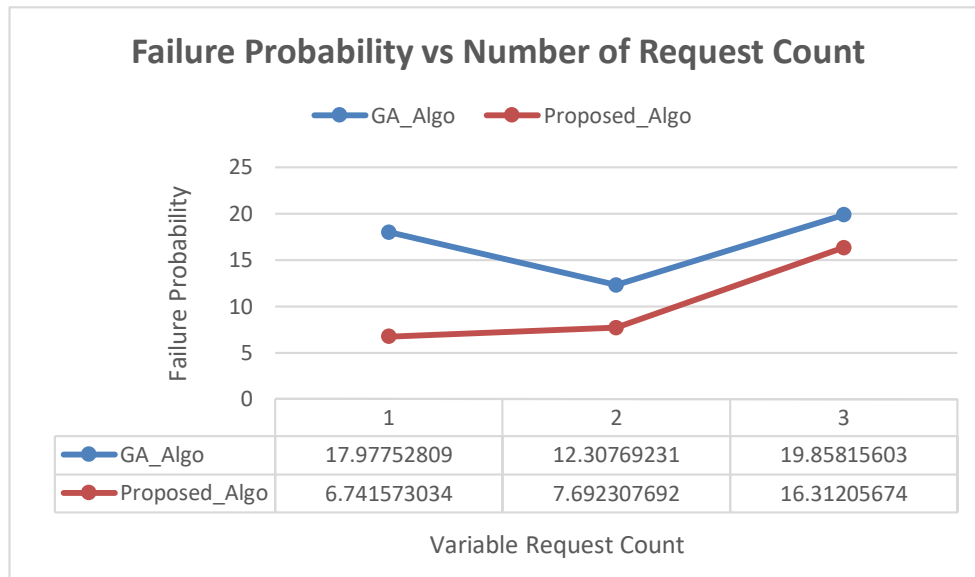


Figure 5: Comparison of decrease in failure probability

The figure shows the decrease in the failure probability over the existing and the proposed algorithm.

VI. CONCLUSION

Reliability and availability of cloud is become major concern of cloud service provider to enhance the scalability and elasticity for real time world and scientific workflow where deadline should be satisfied so fault tolerance mechanism is used which also include the fitness equation so we can not only handle the fault tolerance but decrease the fault probability rate with increase the maximum number of request count.

VII. FUTURE SCOPE

The work is done only considering the failure rate ,so in the future we can incorporate some new feature to increase the accuracy of the proposed fault management system and also the efficiency of the algorithm and implementing the work in real cloud.

REFERENCES

- [1] Zeeshan Amin,Nisha Sethi,Harshpreet Singh, "A Review Paper on Fault Tolerance in Cloud Computing", International Journal of Computer Applications, 2015
- [2] Kashif Bilal, Osman Khalid, Saif Ur Rehman Malik, Muhammad Usman Shahid Khan, Samee U. Khan and Albert Y. Zomaya "Fault Tolerance in the Cloud" ,ENCYCLOPEDIA OF CLOUD COMPUTING, 2016
- [3] Ji Wang, Weidong Bao, Xiaomin Zhu, Laurence T. Yang, and Yang Xiang, "FESTAL: Fault-Tolerant Elastic Scheduling Algorithm for Real-Time Tasks in Virtualized Clouds" ,IEEE, 2014
- [4] Xiaomin Zhu, Ji Wang, Hui Guo, Dakai Zhu, Laurence T.Yang, and Ling Liu "Fault-Tolerant Scheduling for Real-Time ScientificWorkflows with Elastic Resource Provisioning inVirtualized Clouds", IEEE, 2015

- [5] Mohit Kumar Gokhroo, Mahesh Chandra Govil, Emmanuel S. Pilli , “ *Detecting and Mitigating Faults in Cloud Computing Environment*”, IEEE, 2017
- [6] Bashir Mohammed, Mariam Kiranand Irfan and Ullah Awan, Kabiru.M. Maiyama, “*An Integrated Virtualized Strategy for Fault Tolerance in Cloud Computing Environment* ”, IEEE, 2016
- [7] J.Soniya, J.Angela Jennifa Sujana, Dr.T.Revathi, “*Dynamic Fault Tolerant Scheduling Mechanism for Real Time Tasks in Cloud Computing* ”, IEEE, 2016

