IOSR Journal of Engineering (IOSRJEN) **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
Vol. 04, Issue 03 (March. 2014), ||V6|| PP 24-27

High Adaptive Fault Tolerance in Real Time Cloud Computing

Anjandeep Kaur Rai, Parveen Kumar, Pradheep Manisekaran
[1](M.Tech Scholar, Department of Computer Science and Engg, Lovely Professional University (LPU))
[2](Assistant Professor, Department of Computer Science and Engg; Lovely Professional University (LPU))
[3](M.Tech Scholar, Department of Computer Science and Engg, Lovely Professional University (LPU))

Abstract: **-** With the advancement of technology, computing has0 changed in a very drastic way. It has travelled a very long way from parallel to distributed to grid computing. Now a-days, the most pre- dominant internet based computing is Cloud computing. This technology provides a large number of pros like on-demand access, ubiquitous network access, rapid elasticity etc. But like other technology, it suffers from some serious issues as well like workflow scheduling, security etc. Here, we are going to propose a model which is providing high fault tolerance to real time systems in cloud environment. The main feature of this proposed model is the adaptive behaviour of the reliability of each and every virtual machine along with addition and removal of nodes on the basis of reliability If a virtual machine produces correct result within time, then its reliability increases and if it fails to do so, then its reliability decreases as well. And also, priority scheduling has been introduced so as to determine the best node when the reliabilities of two nodes come out to be same.

Keywords: - *Cloud computing, Fault tolerance, Real-time systems, Reliability, Timeliness*

I. INTRODUCTION

Cloud computing has indeed changed the way the computing is performed. It has gained a lot of popularity in a variety of sectors. But like every technology, it suffers from some serious issues, out of which fault tolerance for real time systems are the major one. Real time systems are used in a variety of domains ranging from defence space systems to embedded automotive electronics like mobile phones etc. Behaviour of real time systems not only depends upon the correctness of the logical computations but also on the factor that whether the computation gets performed in an allowed period of time or not [1]. Real time systems are monitored by two properties which are timeliness and fault tolerance. Timeliness is the ability of a real time system to accomplish its intended task within a time limit. And fault tolerance is the capacity of the system to function gracefully even in the presence of fault [2].Real time systems being very critical in nature require to be highly fault tolerant [3]. Inability of such systems to work properly in any case may lead to a financial loss or causalities as well [4]. So, the need of the hour is to enable such systems able to tolerate any kind of fault with cloud. In this paper, we have purposed a model for providing high adaptive fault tolerance in real time systems in cloud.

II. REAL TIME SYSTEMS IN CLOUD

A lot of work has been already done for providing fault tolerance for real systems in case of cloud. However, there is a need for it in real time systems. Cloud sometimes introduces problems to real time systems like it is difficult to determine the latency; still if one needs to know about latency, then it used to continuously change over time [5]. In addition to that, the users of real time systems in cloud don't have full control over the nodes or the virtual machines as they don't know exactly where their applications are going to be processed [6]. But on the good side, cloud provides a dynamic nature to the virtual machines, so, at any time a node can be added or removed [7].

III. EXISTING MODELS

X.Kong et. al. [5, 8] gave a model for the performance and fault tolerance of virtual infrastructure. However, it is not applicable to real time systems. For non-cloud applications, distributed recovery block for distributed real time systems has been proposed by K.H. Kim [9]. Another model, "A formal approach for the fault tolerance of distributed real time systems" has been proposed by J. Coenen and J.Hooman [10]. S. Malik and M. J. Rehman proposed "Time stamped adaptive fault tolerance of distributed real time systems" [11] .This model introduces the idea of time stamping with the outputs. Sheheryar Malik and Fabrice Huet gave "Adaptive Fault tolerance in Real Time Cloud Computing" .This model is based upon an assessment of adaptive reliability of virtual machines in cloud and fault tolerance of real time applications on those virtual machines [7].

IV. PROPOSED MODEL

This paper introduces a model which is an enhancement to the model proposed by Sheheryar Malik and Fabrice Huet in paper entitled "Adaptive Fault Tolerance in Real Time Cloud Computing" [7]. A model is proposed here which is for the fault tolerance of real time applications running on a cloud. The model name is High Adaptive Fault Tolerance in Real Time Cloud Computing (HAFTRC). This scheme tolerates the fault on the basis of reliability of each node, i.e. virtual machine. A virtual machine is selected for computation on the basis of its reliability and can be removed, if it is not able to perform up to the mark. The model is illustrated in Fig. 1.

In this model, we have 2 types of nodes. One type is a set of virtual machines, running on cloud and other is adjudication node. Virtual machine contains some operation to be performed and a corroborating module for determining the correctness and timeliness of virtual machines. In the adjudicator, we have three nodes viz. Reliability calculation, elasticity computation and decision making module.

In this scheme, we have 'N' virtual machines which run some operations. Then we have acceptance module(AM) which is responsible for verifying whether the virtual machine produces correct output and that too within a time limit or not. On, the basis of results produced by the acceptance module, Reliability Calculation (RC) module calculates and reassigns the reliability of each virtual machine. Then the results are passed to the Elasticity Calculation (EC) module which provides some elasticity to the nodes on the basis of their reliability and the system's reliability in terms of CPU's cycles. Then all the results are forwarded to the Decision Making (DM) module which makes the final decision, by selecting the output on the basis of the best reliability. The output of the node with the highest reliability is selected as system output.

4.2 Working of Model

As stated earlier, this technique has N nodes (virtual machines). Each node is taking input data from the input buffer. This input is concurrently passed to all the virtual machines, which run different operations. Each node takes the input, executes the operation and produces a result. These results are passed to the corroborating module. Corroborating module then passes these results to adjudication module for making reliability calculation, elasticity calculation and decision making. There are separate modules in the proposed model which are given different responsibilities.

4.2.1 Acceptance Module (AM)

Acceptance module (CM) is offered to each VM. Actually this module is responsible for two things; one, it is responsible for verifying whether the output that has been produced by the virtual machine is correct and secondly, whether the correct output has been produced in the prescribed time limit or not. It only passes the correct result of those nodes which produces the result before deadline time. If all the nodes fail to produce the result, then CM performs the backward recovery. It also informs the RC module to calculate the new reliabilities of all the nodes.

4.2.2 Reliability Calculation (RC) Module

Reliability calculation (RC) analyzes the reliability for each virtual machine. It is actually the core module of this proposed model. The reliability of virtual machine is adaptive, which changes after every computing cycle. Initially, reliability of each virtual machine is 100%.If a node produces the correct result within deadline time, then its reliability increases and if it fails to do so, then its reliability decreases as well.

4.2.3 Elasticity Calculation (EC) Module

This module analyzes the reliability of the virtual machine. There is a System Reliability Level (SRL) associated with the system, which is the minimum reliability level to be achieved by a virtual machine to be considered as pass. If the reliability of virtual machine is more or equal than SRL, then it is considered pass, and if its reliability is less than SRL, then it is given a chance to be considered as pass by providing elasticity to the virtual machine by allowing the cloudlet to have 15cycles of CPU time if that is getting lacked by 15 cycles. If in case its reliability is very less than the SRL, then it is simply discarded as fail and that node is not allowed to work further and is removed.

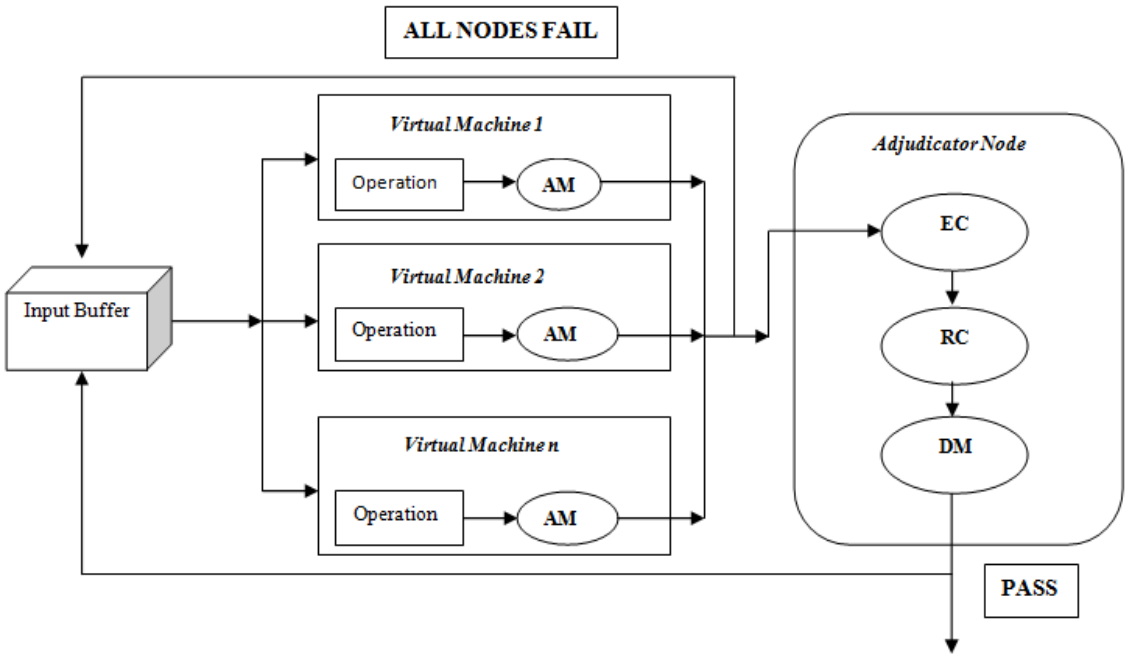

Fig.1. Proposed System Model (HAFTRC)

4.2.4 Decision Making (DM) Module

Decision Making (DM) module has the responsibility of selecting the final output for a computing cycle. It selects the output of a node which has the highest reliability among all the virtual machines. All the tasks that are running on the virtual machine are given priorities. If at this stagse, two nodes have highest and same reliability, then the node will be selected depending upon the priority i.e. whichever node has the highest priority, will be considered as the best These virtual machines are the competing nodes which have produced the correct results within time. These reliabilities are given elasticity by EC module if they are applicable to make it. This proposed model provides a kind of automated forward recovery. If a node fails to produce output or produce output after the system has run successfully, then the system will not fail. The system will continue to operate with the remaining nodes. This model will produce output until all the nodes fail.

V. CONCLUSION

In this paper, a model, HAFTRC, has been proposed which is providing high adaptive fault tolerance to real systems in cloud. This model is an enhancement to an already existing model (AFTRC) [2] and provides high reliability to the real time systems in cloud. It is emphasizing on having more passed nodes in comparison to failed ones and also it gives output even if one nodes produces output In future, more parameters can be added to it for improving the reliability and also here we are implementing priority scheduling algorithm whenever two nodes have same reliability, so some other algorithm like round robin algorithm can also be used in its place. In previous model (AFTRC), if one of the node gets failed to produce the desired result, then entire model is not producing result, but in this proposed model, even if one of the node is producing result, then even we are getting something out of it i.e. it is more fault tolerant in comparison to previous model.

VI. ACKNOWLEDGEMENT

The making of the paper needed co-operation and guidance of a number of people. I therefore consider it my prime duty to thank all those who had helped me through their venture. We express our sincere thank to all the staff members of CSE department for their kind co-operation. We are thankful to our friends whose encouragement and suggestion helped us to purpose our work.

REFERENCES

[1] https : // www . ece . cmu . edu /~koopman /des_s99/real-time/

[2] Malik Sheheryar, Huet Fabrice ,Adaptive Fault Tolerance in Real Time Cloud Computing, *2011 IEEE World Congress on Services, pp.280-287*

[3] K. H. Kim, Distributed Execution of Recovery Blocks: An Approach to Uniform Treatment of Hardware & Software faults, *Proceeding fourth International Conference on Distributed Computing Systems, 1984, pp. 526-532*

[4] L. L. Pullum, Software Fault Tolerance and Implementation, *Artech House, Boston, London, United Kingdom, 2001*

[5] X. Kong, J. Huang, C. Lin, Comprehensive Analysis of Performance, Fault-tolerance and Scalability in Grid Resource Management System, *2009 Eighth International Conference on Grid and Cooperative Computing, Lanzhou, China, August 27-29, 2009*

[6] W. T. Tsai, Q. Shao, X. Sun, J. Elston, Real Time Service Oriented Cloud Computing, *School of Computing Informatics and Decision System Engineering Arizona State University USA, http://www.public.asu.edu/~qshao1/doc/RTSO A.pdf*

[7] Malik Sheheryar and Huet Fabrice (2011) , Adaptive Fault tolerance in Real Time Cloud Computing, *2011 IEEE World Congress on services, pp. 28-287*

[8] X. Kong, J. Huang, C. Lin, P. D. Ungsunan, Performance, Fault-tolerance and Scalability Analysis of Virtual Infrastructure Management System, *2009 IEEE International Symposium on Parallel and Distributed Processing with Applications, Chengdu, China, August 9-12, 2009*

[9] K. H. Kim, *Structuring DRB Computing Stations in Highly Decentralized Systems*, Proceedings International Symposium on Autonomous Decentralized Systems, Kawasaki, 1993, pp. 305-314

[10] J .Coenen, J. Hooman, A Formal Approach to Fault Tolerance in Distributed Real-Time Systems, *Department of Mathematics and Computing Science, Eindhoven University of Technology, Netherland*

[11] S. Malik, M. J. Rehman, Time Stamped Fault Tolerance in Distributed Real Time Systems, *IEEE International Multitopic Conference, Karachi, Pakistan, 2005*