ISSN: 0975-8585

Research Journal of Pharmaceutical, Biological and Chemical Sciences

# Fault Tolerance Approach To Improve Performance Computation Of Biological Jobs Using Cloud Computing.

**P Padmakumari *, and A Umamakeswari.**

CSE, School of Computing, SASTRA University, Thanjavur, Tamilnadu, India.

**ABSTRACT**

Various biological jobs are executed in cloud environment to utilize the resources and improve the performance of the computation. Free from unnecessary yoke related to supervising those biological computing resources and get rid of the expenditure of maintaining the resources. Reliability is a complex issue in a cloud environment, a key to improve it by using Fault Tolerance Approach (FTM). Proactive and Reactive Fault Tolerance strategies are there to provide services continuously in spite of failures or faults occur to offer highly reliable cloud service. The paper proposes a framework which integrates Fault tolerance and Anti-fragility to survey on both fault prediction and recovery method. This framework has three stages (i) Fault Detection Watcher (FDW) (ii) Fault Overseer (FO) (iii) Fault Resilience. FDW is used to identify faults either known or unknown. Fault overseer is used as a monitoring mechanism for proactive policy predict and recover known faults and Fault Resilience is for fault tolerance as reactive policy detect and recover unknown faults. Network and application faults are concerned with fault overseer; Fault induction and event log are the phases. Database and VM failures look after by Fault resilience stage. Fragments and replication are core objective of this phase. Proposed approach validates using reliability metrics in cloudsim simulator. The experiment result revealed the probability of proposed approach under the conditions of availability, reliability and performance. This paper shows a novel framework in an integrated manner, of proactive and reactive policies in terms of fault overseer and anti-fragility mechanism to execute biological job.

**Keywords:** cloud computing, fault tolerance, monitoring mechanism, biological jobs.

***Corresponding author***

ISSN: 0975-8585

## INTRODUCTION

Large scale biological jobs can be executed with resources available in cloud environment. Computation of biological jobs should be uninterrupted to acquire high performance in the result. Failures in computation direct to take away Reliability of the jobs. Possible fault can be occurring in cloud can be as follows (i) virtual failure (ii) network failure (iii) application failure. To improve performance, failure has to tolerate with the mechanism. Fault tolerance is the technique to permit a system to continue functioning even failure occurs in case of some component fails. Reliability and Availability can be enhancing with fault tolerance mechanism. Proactive and Reactive are the forms of fault tolerance. In reactive fault tolerance, recovery of failures acquire after it happens. In proactive fault tolerance, fault can be predicted and proactively recover it. In adaptive fault tolerance can be automatically fault can be recovered.

## METHOD

### Fault Tolerance Approach

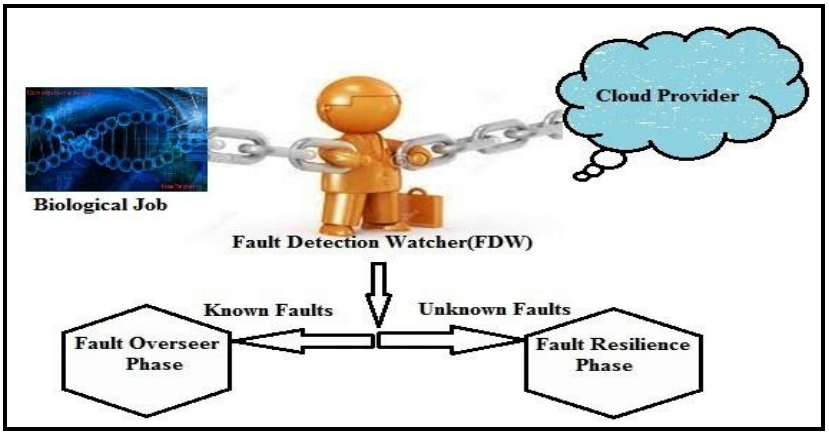
This section illustrates about Fault t Tolerance Approach functionalities. And also elaborate different stages of approach combined proactive and reactive fault policies. Following are the steps how it works, Fault Detection Watcher (FDW) stage acts as middleware between cloud provider and customer. It monitors and detects network, database and virtual machine failures occurs both in cloud provider and customer side.

1) Detection of the faults from FDW is either directed to fault overseer stage if it is known fault else to Fault Resilience stage in case of unknown fault
2) In fault overseer stage faults are injected as a preventive method. Using risk analysis method, analysis the performance of the fault and maintain the solution for predicting fault in the event log. Detected fault direct from FDW is checked with a replacement solution from the event log.
3) Unknown faults directs to Fault Resilience stage where fragmentation and replication of data stored in cloud happens to make available the data whenever needed by customer.
4) Proposed approach makes retrieval or storing of data in cloud storage as reliable by using combined proactive and reactive fault tolerance policies.

FDW is a middleware between Fault overseer stage and Fault Resilience stage. This approach is designed to provide proactive and reactive fault tolerance, which can afford customers reliable storage capacity. Fault Tolerance Approach is a combined framework which gives monitoring as a preventive method for fault occurrence and resilience as a recovery mechanism after detecting the fault.

Fig.1 illustrates about the overall structure of Fault Tolerance Approach. FDW monitors both cloud provider and customer. It makes fault prediction and recovery as trouble-free. To improve the reliability of cloud service, responsibilities are shared among both cloud provider and cloud customer. In this connection, FDW be in touch and monitors the action on both sides. In this paper, introduce the unified monitoring and resilience mechanism and its framework in the following sections.

**Figure 1: Fault Tolerance Approach**



**Fault Detection Watcher (FDW) Mechanism**

This section brings out the purpose and benefit of the Fault Detection Watcher (FDW) mechanism. FDW is monitoring mechanism which comes in proposed approach. It acts as middleware between cloud provider and customer and monitors the occurrence of faults. FDW focus on network, application, and virtual machine and database failures. Interact with cloud customer, often to monitor network or application failure occurrence. Virtual machine failure or database failure can be tolerated by communicating with a cloud provider. So FDW makes cloud provider and customer to share the responsibility to retrieve and storing of data to and from cloud storage. FDW mechanism is having Watcher Head (WH) and multiple Watcher Workers (WW). WH maintains a management plan, log, which holds the information about the faults which collects from various WW. According to this log FW identifies fault may be known or unknown faults and redirects to either Fault overseer stage or Fault Resilience stage. In this paper, application and network failure while retrieving or storing is considering as known faults. Virtual machine and database failures are considered as unknown faults.

**Detection of Application faults**

Application failure causes service disruption of cloud. Application failure also include with hardware and software failures. If customer requests for a data from cloud storage, sometimes it cause of incorrect items displayed in response to a customer request, inability to complete request, data loss or corruption and performance slowdown. In order to monitor those failures.WW follows the following techniques (i) watch customer response time (II) keep an eye on request and response data from storage (III) collects mean time to request and mean time to respond continuously monitoring the infrastructure and details are maintained in log handle by WH. Now WH decides either faults can recovery immediately by Fault overseer stage or Resilience stage.

**Detection of Network faults**

An important part of managing the cloud servers is monitoring network connectivity. To supervise the network connection, WW is provided with the tool called trace route to diagnose where a network issue may be happening. Following information get back as a result of this tool (i) specified host (ii) IP address (iii) maximum number of hops to check (iv) size of the packet used. Periodically this information is maintained in a log with WH. If WW doesn't receive the reply properly, it intimate to WH.WH takes necessary steps to direct to Fault overseer stage to recover the problem instantly.

**Detection of Virtual machine (VM) or database failure**

WH is having details about the current status of VM's and databases. Through WW, it updates the details within a period of time. If any of the following circumstances happens, it is considered as VM failure. (i) Host stops its working task (ii) isolate from host network (iii) not a response to the request within the specified time. WH redirects to Fault Resilience stage to recovery the VM failure. WW monitors the customer request for retrieving or storing of data in cloud storage. If customer does not receive or store within a specific period of time, alert given to WH to redirect the control to Fault Resilience stage for recovery of data in the database.

The above are the purpose of FDW. Classification of error detection and move for recovery to related stages makes fault recovery very accurate and make available of data at any time. Availability obviously increases the Reliability. In subsequent sections Fault Overseer and Fault Resilience stages can be elaborately discussed.

**Faults Overseer stage**

Fault overseer stage follows the proactive fault tolerance policy. Proactive fault tolerance policy avoids faults by predicting in earlier stage. Some techniques used are self healing, rejuvenation etc.. In this framework, fault injection and event log techniques are used to predict the faults. There are two phases (i) Fault stimulation and Analysis phases (ii) event log. Fig.2. shows the overall function of the Fault overseer stage. Following sub sections explained clearly about each phase. Fault stimulation phase concentrates on network and application failure. Monitoring eye come forward and induce the error using fault injection

ISSN: 0975-8585

method. Possible nodes can fail within a given time is noted carefully. Fault injection times, time takes to recover the fault and delay period are sending to risk analysis phase. In order to detect probable failures, stimulate the failures and create statistical models using the time analysis method. A Probabilistic model *p* is taken for the faults with reduced dimensionality. It takes a fault occurrence in a series of time *t* as input and outputs a probability for those faults. The parameters of *p are* learned from stimulating faith in an unsupervised manner. If t (p) <T then it is predicted as failure where T is threshold can be determined learning experience. The result of predicted information got logged in Event log phase.

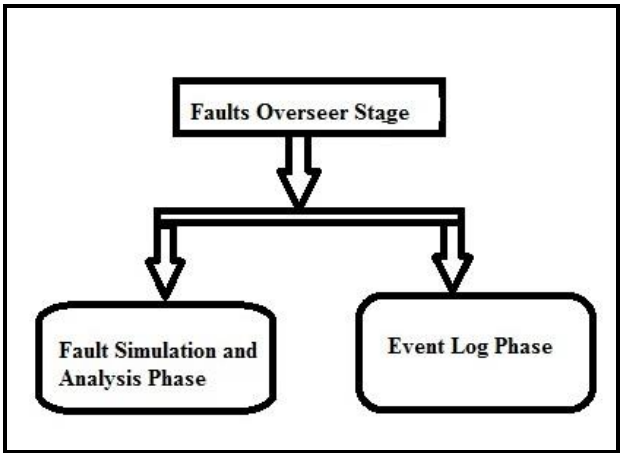



**Figure 2: Fault Overseer Stage**

**Fault Resilience Phase**

Reactive fault tolerance comes under Fault Resilience Phase, where fault recovery can be occurred after fault occurs. Resources can be run on different virtual machines. In order to failure occur in a virtual machine while resources are in process. Fault Resilience phase maintains logs about the virtual machine. From this failed process can be move to its replicated machine. Fig 3 illustrates the overall method of Fault Resilience Phase.

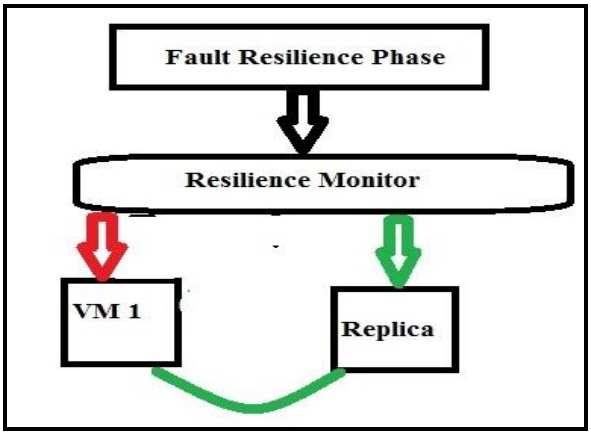



**Figure 3: Fault Resilience Phase**

**RESULTS AND DISCUSSION**

This section deals with efficiency and effectiveness of Federated FOR method. Assessment can be evaluated using CloudSim. CloudSim is framework for simulating cloud computing infrastructure and services. Implementation handled on CloudSim 3.0.3 integrated with eclipse IDE, intel i5 machine 4GB RAM running a window7.To find out the efficiency metrics(E) of proposed method, it is necessary to calculate Total fault recovery time (RFT).Result can be find out by adding time taken to use normally(Nt),Faulty(Ft) and Recovery(Rt).

ISSN: 0975-8585

E=Nt+Ft+Rt

To illustrate the performance of Federated FOR three situations are noted

(i) Finest situation, where failure is not occur
(ii) Intermediate situation, where atleast one failure in executed task
(iii) Worst situation, where failure in all situation

Suppose 10 tasks computed in cloud environment at the same time. Assume that no heartbeat message receives from the provider side. Failure detection and recovery can be compared based the above three situation by using and without using Federated FOR. Herewith consider Existing as M1 and Proposed Method as M2.Using M2 to diagnosis and recover both proactive and reactive faults take less time when compared with M1.

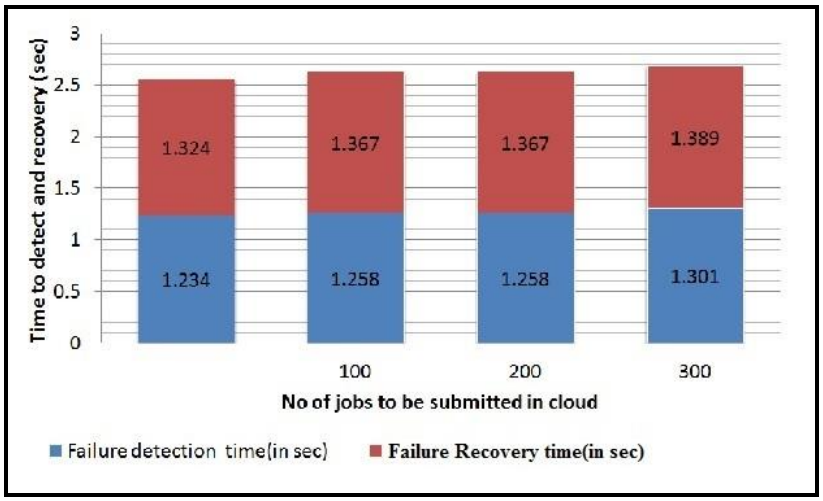



**Figure 4: Time to detect and recover the failure using Proposed Fault tolerance approach**

In the Figure 4 explains that number of biological jobs submitted in cloud environment for computation. This graph give details about the time taken to detect the failure occurs and time taken to recover that failure to stable with the performance of submitted job

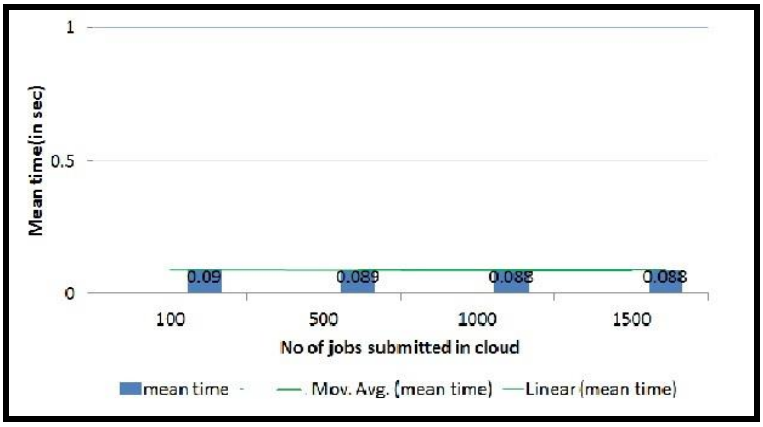



**Figure 5: Performance of Proposed Fault Tolerance Approach**

In figure 5 makes clear that number of biological jobs increase may result to fault detection and recovery mean time to be moderate. Number of jobs increases may result to nearly similar mean time. Performance of proposed approach is enhanced even number of jobs increase. Figure 6 compares existing fault tolerance method with proposed Fault tolerance approach

ISSN: 0975-8585

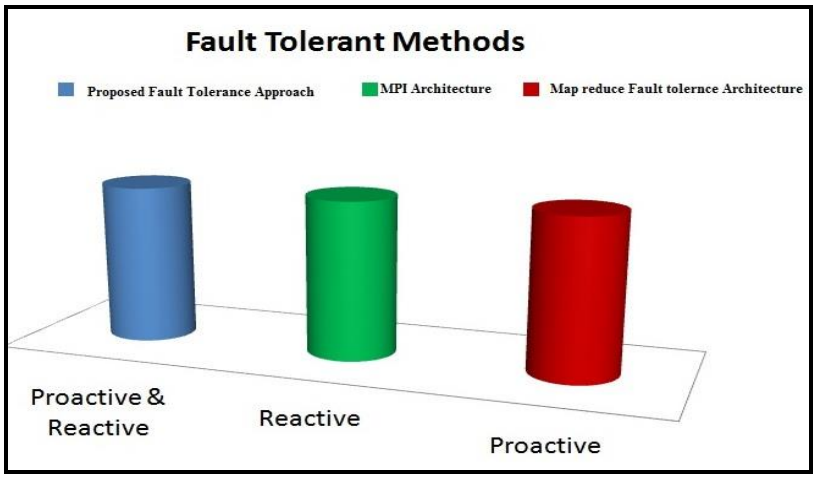



**Figure 6: Comparison of proposed approach with existing architecture**

## CONCLUSION

This paper proposed with Fault tolerance approach to compute biological jobs in cloud environment without any failure and improves the reliability and performance. Two stages are there in this approach (i) fault overseer stage which concern application and network failure in proactive manner (ii) fault resilience stage which concern virtual failure in reactive manner. Result shows that faults can be tolerate with proposed approach even though number of jobs increased.

## REFERENCES

[1] Wang L, Kalbarczyk Z, Iyer RK, Iyengar A. Checkpointing Virtual machines Against Transient Errors: Design, Modeling, and Assessment. 2010

[2] Haikun Liu, Hai Jin, Xiaofei Liao, Optimize Performance of Virtual Machine Checkpointing via Memory Exclusion, 2012.

[3] Jun Nakano, Pablo Montesinos, Kourosh Gharachorloo†, and Josep Torrellas, ReViveI/O: Efficient Handling of I/O in Highly-Available Rollback-Recovery Servers, 2012.

[4] Bala, A., & Chana, I. Fault tolerance-challenges, techniques and implementation in cloud computing. IJCSI International Journal of Computer Science, 2012 Issues, 9(1), 1694-0814.

[5] Varghese, B., McKee, G., & Alexandrov, V. Automating fault tolerance in high-performance computational biological jobs using multi-agent approaches. Computers in biology and medicine, 2014, 48, 28-41.

[6] Koushik, C. S., Reddy, K. R., Reddy, Y. R., Padmakumari, P, Umamakeswari, A. Location as attribute and re-encryption-based secure and scalable mechanism for mobile based applications in cloud. Indian Journal of Science and Technology, 2015, 8(12).

[7] Akshaya, G., Subha, K., Baggiya, R. T., Padmakumari, P., & Umamakeswari, A.Coalition of Cloud Monitoring Systems Postulating Anti-Fragility. Indian Journal of Science and Technology, 2015, 8(S9), 181-187.

[8] Giriesh, S., Sindhuja, V., Padmakumari, P., & Umamakeswari, A. (2015). Dynamic Data Fault Tolerance Mechanism to Enhance Reliability and Availability in Cloud. Indian Journal of Science and Technology, 2015, 8(S9), 300-305.

[9] Cheraghlou, M. N., Khadem-Zadeh, A., & Haghparast, M, A survey of fault tolerance architecture in cloud computing. Journal of Network and Computer Applications, 2015

[10] Li, Z., Cai, W., Turner, S. J., Qin, Z., & Goh, R. S. M. Transparent three-phase Byzantine fault tolerance for parallel and distributed simulations. Simulation Modelling Practice and Theory, 2016,60, 90-107.