

Available online at <http://www.mecspress.net/ijem>

An Improved Pre-copy Approach for Transferring the VM Data during the Virtual Machine Migration for the Cloud Environment

Praveen Jain ^a, Ratish Agrawal ^b

^a Student at School Of Information Technology, RGPV, Bhopal, 462023, India

^b Assistant Professor at University Institute of Technology, RGPV, Bhopal, 462023, India

Abstract

Virtualization technology play a vital role in cloud computing. In virtualization environment multiple applications can run at a same time. VM migration is one of the important features of the virtualization, which allows application to be transparently migrated along with their execution environments across physical machines. VM migration consists of four steps. These steps are source PM selection, VM selection from the selected PM in first step, target PM selection for placing the selected VM and selecting method for transfer VM data. In our proposed approach we are focusing on the last step i.e., choosing method for transferring VM data.

In this paper proposed approach is divided into two phases. First phase filter all pages which is modified in the last iteration and unmodified in the current iteration. In the second phase, page is divided into two types i.e., high dirty page and normal based on the modification in the last few iterations. For all filtered pages in the first phase now we check the number of times when the page is modified from the history record. If the page is modified more times the page will not be send in the current iteration otherwise the page will be send to the destination VM.

To evaluate the performance of the proposed approach it is implemented in CloudSim simulator and compare with the existing time series based pre-copy approach in term of total migration time and down time. Experiment result shows that proposed approach gives better result as compare to the base approach.

Index Terms: Pre-copy, post copy, stop and copy, migration, virtual machine, physical machine.

© 2016 Published by MECS Publisher. Selection and/or peer review under responsibility of the Research Association of Modern Education and Computer Science.

1. Introduction

Cloud computing is one of the fastest growing technology in the field of computer science [1],[13]. It received much attention in the last few years due to the attractive features like on-demand services, easy to use, measured services etc [2]. It can be deploy in different ways i.e. private, public and hybrid and support for the

* Corresponding author.

E-mail address: praveenjain45@gmail.com^a, ratish@rgtu.net^b

different types of services like software as a service (SaaS)[14], platform as a service (PaaS) and infrastructure as a service (IaaS) [3].

Virtualization technology [4, 5] play a vital role in cloud computing. In virtualization environment multiple applications can run at a same time. VM migration is one of the important features of the virtualization, which allows application to be transparently migrated along with their execution environments across physical machines. Virtual machine migration [6, 7] is needed for load balancing, server consolidation (power saving) and resource scheduling. VM migration consists of four steps. These steps are source PM selection, VM selection from the selected PM in first step, target PM selection for placing the selected VM and selecting method for transfer VM data.

In our proposed approach we are focusing on the last step i.e., choosing method for transferring VM data. Three methods are available named stop and copy, pre-copy and post-copy, for transferring the VM data from one PM to another PM [8]. Previous study shows that pre-copy approach gives better result as compare to the other two approaches. Due to this reason we are also using pre-copy approach to transfer the VM data. Pre-copy approach is the iterative method where VM pages are send continuously to the destination VM [15]. In the first phase all VM pages are transferred and in the second iteration pages modified in the first iteration are send to the destination VM, this process is continue until number of pages modified in the last iteration or maximum number of iteration is reach to the pre define threshold.

Downtime and total migration time play an important role in the VM migration. For effective migration, downtime and migration time should be minimum. Down time refers to the period between the stop-and-copy operation of VM on the source host and the activation operation of VM on the target host. That means downtime is the time where the VM is not available to the user. Whereas total migration time refers to the total time from the migration operation of VM is activated on the source host to the time VM is activated on the target host.

The proposed approach modifies the existing “Time series based pre-copy approach for live migration of virtual machine”. In this approach page is divided into two phases. First phase filter all pages which is modified in the last iteration and unmodified in the current iteration. In the second phase, page is divided into two types i.e., high dirty page and normal based on the modification in the last few iterations. For all filtered pages in the first phase now we check the number of times when the page is modified from the history record. If the page is modified more times the page is not send in the current iteration otherwise send the page to the destination VM.

This paper is organised as follows; section 1 is introduction about cloud, virtualization, Virtual machine migration, Pre-copy approach and performance parameters like downtime, total pages transferred, and total migration time, section 2 describes about the related work carried out previously in virtual machine migration approaches, section 3 gives the proposed work along with flow chart and the algorithm, section 4 discuss about the result and the last section gives the conclusion.

2. Related Work

In this chapter we discuss the work that has been carried out in the past in the field of Virtual Machine Migration along with differences and similarities between the previous work and ours. We also discuss various performance metrics which is important for VM migration, below we are presenting a review of some of the noteworthy contributions

R. Michael et al. [8] proposed live migration approach based on the post copy. Post-copy migration defers the transfer of a VM’s memory contents until after its processor state has been sent to the target host. This deferral is in contrast to the traditional pre-copy approach, which first copies the memory state over multiple iterations followed by a final transfer of the processor state. They use Dynamic Self-Ballooning (DSB) to reduce the number of free pages without significantly impacting the normal execution of the VM, so that the VM can be migrated quickly with a minimal memory footprint. Main limitation of this approach is once the VM resumes at the target, demand paging begins for missing pages that can increase the total migration time.

Wentian cui and Meina song [9], proposed a VM migration approach by using Matrix Bitmap Algorithm. This approach is based on the “Program Locality Principle”, and use “matrix bitmap algorithm” that collects the dirty page information for many times before deciding whether to transfer the page or not.

F. Ma et al. [10], proposed Improved Pre-copy approach for migrating the VM in which some bitmaps are introduced which marks frequently updated pages. Although pre-copy approach can balance the downtime and total migration time of contradictions, it will cap the number of copying iterations to the maximum number of iterations.

H. Bolin et al. [11] proposed a pre copy based live migration technique for the cloud. In which, they evaluate traditional precopy approach, and proposed an improved time-series based precopy approach for virtual machine migration. With the time-series prediction technique, they identify frequently updated dirty pages (high dirty pages) in the past and future period more precisely, and transmit them in the last round of iteration, in order to reduce unnecessary, repeated transmission of dirty pages. Doing so can significantly reduce the total migration time.

3. Proposed Work

VM migration is the key feature of the virtualization because it allows to move running VM from one PM to another PM. Performance of any VM migration is depends on the down time and total migration time. Previous study says that pre-copy approach gives better result as compare to the other approaches due to this reason proposed approach use the pre-copy approach for the migration. Main problem with the pre-copy approach is that it send the same page several times if the page is modified regularly, which increase the total migration time and downtime. To mitigate this problem a time series based live migration approach were proposed [11]. This approach divides the pages into two groups normal and high dirty page. High dirty page ware transferred in the last iteration. To define when the page is in high dirty group a threshold is use. Performance of this approach is mainly depends on this threshold value. Hence, finding the appropriate value of this threshold is a challenging task. So to mitigate this issue we proposed a live migration approach that uses the history of the page to decide whether the page is dirty or not. Hence we eliminate the requirement of the threshold.

In the proposed pre-copy approach page is divided into two group named high dirty page and normal page. A page which is modified more than it's predefine threshold then the page belong to the high dirty page group otherwise page belongs to the normal group. To minimize the number of transferred page in the iteration, pages belonging to the high dirty page group are send to the last iteration. Proposed pre-copy approach is divided into two phases

Phase – 1

In this phase we check whether the page is modified in the last iteration or not. Four bitmaps designated to_send, to_skip, to_fix and to_send_last is used to record the status of page.

to-send - bitmap hold pages modified in the last iteration.

to-skip - bitmap hold pages modified in the current iteration.

to-fix - hold pages which are fixed to be sent in last iteration.

to_send_last - bitmap hold the pages which are frequently modified. These pages are send in the last iteration.

Table 1. Different Value of to_send and to_skip Bitmap

to_send	0	0	1	1
to_skip	0	1	0	1
Send or not	No	No	Yes	No

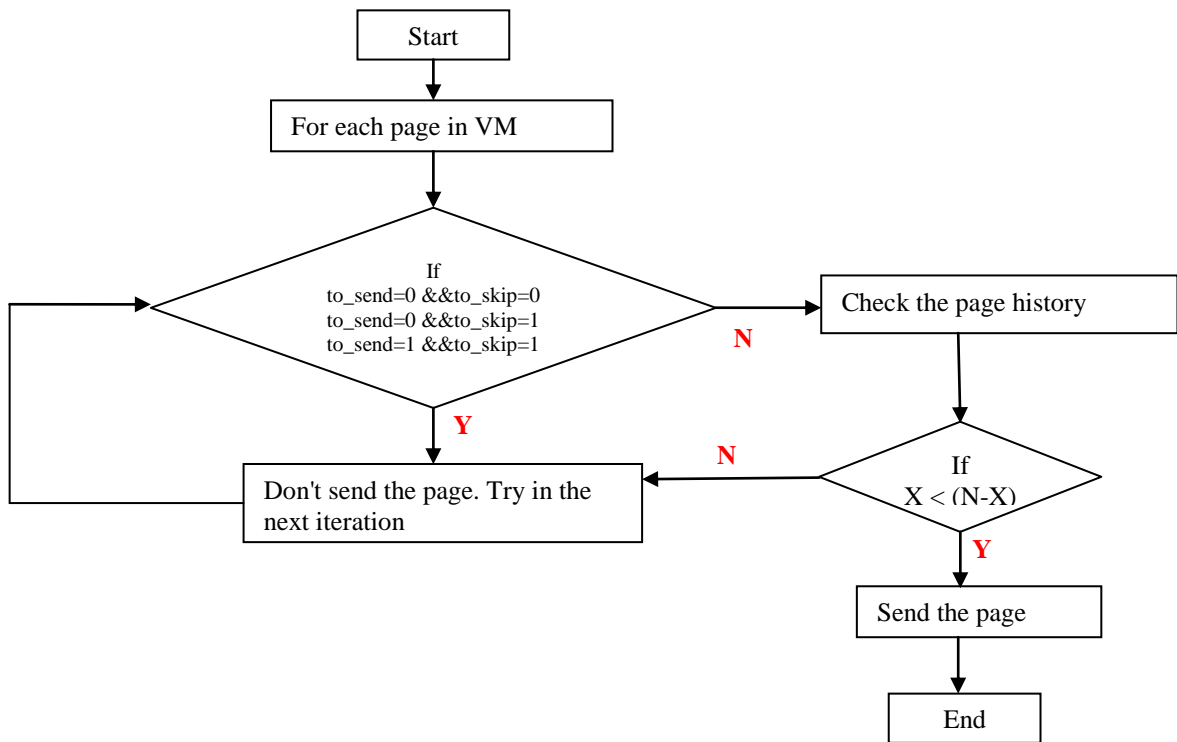


Fig.1. Flow Diagram for the Proposed Pre-Copy Approach

Algorithm for Transferring the VM Data

Number of Pages= n ,
 iteration_threshold= m
 Number of pattern= 20
 $n=1$
 Number of history page for each VM= k ,
 $to_skip=0, to_send=0, to_send_h[i]$.

```

[1] for (n=1; n<=20; n++) do
[2] for (n=1; n<=m; n++) do
[3] for each pages of VM do
[4] if (to_send=0 &&to_skip=1 || to_send=1 &&to_skip=1 || to_send=0 &&to_skip=0) then
[5] Don't send the pages into the current iteration
[6] else
[7] if (to_send=1 &&to_skip=0) then
[8] Check the history of the pages
[9] Calculate  $X = \sum_{i=1}^k to\_send\_h[i]$ 
[10]  $Y = k-X$ 
[11] if  $x < y$  then
  
```

```

[12]             Transfer the page into the current iteration
[13]         else
[14]             Don't send the page.
[15]         end if
[16]     end if
[17] n=n+1
[18] end for
[19] Stop the VM at the source PM and transferred all remaining VM pages. Now start the VM on the
    target PM
[20] end for
[21] end for

```

Here X shows the number of 1 and Y shows the number of 0 in the history record of each page. Here we avoid the use of threshold to identify the high dirty page and this deification is done with the help of history record.

4. Result Discussion

To evaluate the performance of the proposed approach CloudSim simulator [12] is use. For this purpose two VM is created and each VM is configure with 250 MIPS, 128 MB RAM, and 10000 Kbps bandwidth. An application is running on each VM which consist of number of pages. When the VM is migrated these pages need to transfer from one VM to another VM. For evaluate the performances of the proposed approach it is compare with the existing pre-copy approach name " A Time-Series Based Precopy Approach for Live Migration of Virtual Machines [11]". To check the effect of number of pages on the total migration time and down time, number of pages carry by each VM is change during the experiment. To generate the page sequence a random function is use which generates 20 random samples for each page in the VM. Time taken to transferred each page is assume to 6.78 ms. Following experimental setup is use to build the environment for evaluate the performance of proposed work.

For the proposed method

Number of creating VM = 2
 Pages per VM= 10, 15, 20
 History Record =5
 Iterations threshold =5

For the previous method

Number of creating VM = 2
 Pages per each VM= 10
 History Record for =5
 Iterations threshold =5
 Dirty_Threshold=5

To check the performance of the proposed work it compare with well know live migration approach [9] in term of total page transferred, Down time and Total Migration time.

Analysis of total number of page transmitted before applying stop and copy

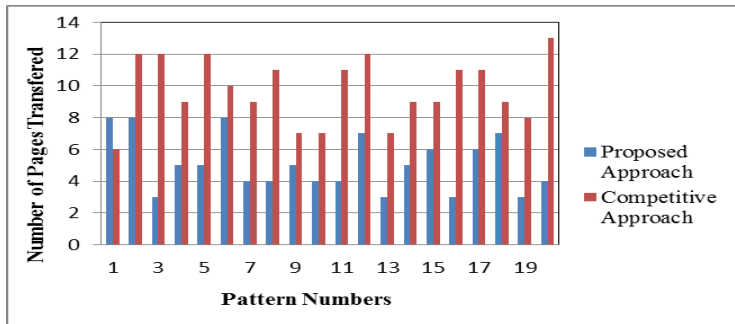


Fig.2. Number of Page Transferred for 10 Pages in Each VM before Applying the Stop and Copy in proposed and Competitive Approach

Analysis of down time:

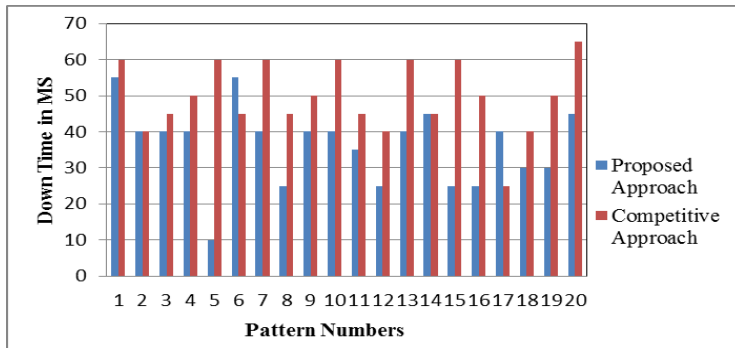


Fig.3. Down Time for 10 Pages in Each VM in Proposed and Competitive Approach

Analysis of total Transmission time:

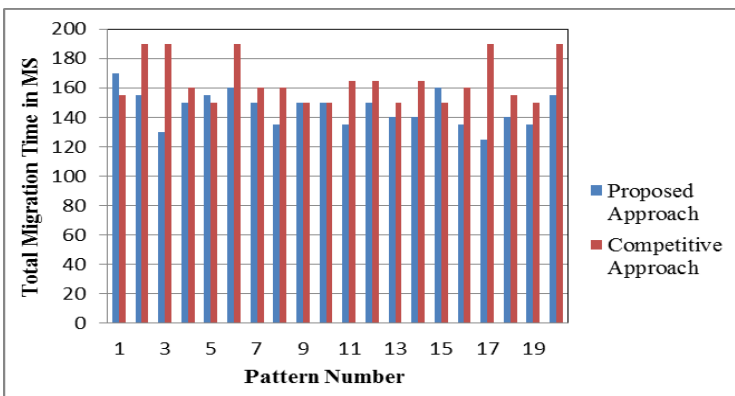


Fig.4. Total Transmission Time for 10 pages in Each VM during the migration process in Proposed and Competitive Approach

After analysing the figures 2.3 and 4 it can be conclude that our proposed approach transmitted less number of pages in stop and copy phase. Hence proposed approach minimizes the total downtime. This is because due to the setting of threshold value on the maximum dirty page. In the proposed approach after checking the mandatory condition (to_skip=0 and to_send =1), page can be transferred only when is number of zero is more number of one.

Experiment results say that proposed approach gives the better result. It has minimum down time, minimum total migration time and minimum number of page transferred before applying stop and copy as compare to the competitive approach.

5. Conclusion

Total migration time and down time play an important role in the VM migration. This work provides an enhancement on existing pre-copy based live migration techniques and presents a new idea for live migration of virtual machine in cloud computing environment which reduces the time taken for migrate the virtual machine or application from one physical host to another host. Proposed Approach modifies the existing “Time Series Based Approach”. Main problem in “Time Series Based Approach” is that it is very difficult to calculate the K/N ratio where K is the threshold and N is the number of history in each pages. Our method eliminates the problem of taking threshold. Our method calculates the number of 0 and number of 1 and then by comparing total number of 0 and 1 decide whether the page is send or not in the current iteration. That means we check how many times a page is modified and if it is modified more times than unmodified times, page will not transferred. It improves results by reduced virtual machine downtime and total migration time.

Acknowledgements

We would like to thank all the anonymous reviewers for providing their reviews and comments for the improvement of this paper.

References

- [1] R. Buyya et al., “Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility”, *Future Generation Computer Systems*, 2011.
- [2] Sosinsky et al.” *Cloud Computing Bible*”, Wiley Publishing Inc 2012.
- [3] R. K. Gupta et al., “A Complete Theoretical Review on Virtual Machine Migration in Cloud Environment”, *International Journal of Cloud Computing and Services Science (IJ-CLOSER)*, Vol.3, No.3, June 2014, pp. 172-178.
- [4] Barnham et al., “Xen and the art of virtualization”, *Proceedings in 19th ACM Symposium on Operating System Principles (SOSP’03)*, 2003, pp. 164–177.
- [5] VMware, VMware ESX and VMware ESXi. [Online]. Available: <http://www.vmware.com/files/pdf/VMware-ESX-and-VMware-ESXi-DS-EN.pdf>
- [6] C. Clark et al., “Live migration of virtual machines,” in *Proceeding NS DI’05 Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation*, vol. 2, pp. 273– 286, 2010.
- [7] R. K. Gupta et al.,” *Survey on Virtual Machine Placement Techniques in Cloud Computing Environment*”, *International Journal on Cloud Computing: Services and Architecture (IJCCSA)* ,Vol. 4, No. 4, August 2014, pp. 1-7.
- [8] M. R. Hines and K. Gopalan, “Post-Copy Based Live Virtual Machine Migration Using Adaptive Pre-Paging and Dynamic Self-Ballooning”, *ACM*, 978-1-60558-375-4/09/03,PP 51-60, March 11–13, 2009.

- [9] W. Cui, M. Song, "Live Memory Migration with Matrix Bitmap Algorithm", IEEE, 978-1-4244-6359-6/10, pp 277-281, 2010.
- [10] F. Ma, F. Liu, Z. Liu, "Live Virtual Machine Migration based on Improved Pre-copy Approach", IEEE, Vol.7 No.10, pp-230-233, 2010.
- [11] B.Hu, Z. Lei, Y. Lei, D. Xu, J. Lei, "A Time-Series Based Precopy Approach for Live Migration of Virtual Machines", IEEE 17th International Conference on Parallel and Distributed Systems, pp-947-952, 2011.
- [12] R. Calheiros, R Ranjan, C ésar A. F. De Rose, R. Buyya, "CloudSim: A Novel Framework for Modeling and Simulation of Cloud Computing Infrastructures and Services", 2011.
- [13] S.Goyal "Public vs Private vs Hybrid vs Community - Cloud Computing: A Critical Review", published online February 2014 in IJCNIS, pp-20-29.
- [14] Md.Imran et al.. "A Comprehensive Survey on Cloud Computing" Published Online January 2015 in IJITCS, pp-68-79.
- [15] S. Sharma et al.. "Two Fold Optimization of Precopy Based Virtual Machine Live Migration", Published Online September 2015 in IJITCS, pp-8-14.

Authors' Profiles



Praveen Jain has completed his Masters in Cyber Forensic from School of Information Technology, R.G.P.V. Bhopal. His dissertation work was carried out in the field of virtual machine migration in the cloud environment under the guidance of Asst. Proff. Ratish Agarwal of University Institute of Technology, R.G.P.V.Bhopal.



Ratish Agarwal has completed his Doctorate in Networking and Communication from R.G.P.V. Bhopal. His Research intresets includes wireless communication and networking and has published number of papers in various international journals and confrences.

How to cite this paper: Praveen Jain, Ratish Agrawal, "An Improved Pre-copy Approach for Transferring the VM Data during the Virtual Machine Migration for the Cloud Environment", International Journal of Engineering and Manufacturing(IJEM), Vol.6, No.6, pp.51-60, 2016.DOI: 10.5815/ijem.2016.06.05