I J C T A, 9(18) 2016, pp. 8973-8980

A Novel Approach for Virtual Machine Migration in Cloud Computing

Yashveer Yadav[*] and C. Rama Krishna[*]

ABSTRACT

Live Virtual Machine migration is a process in which Virtual Machine (VM) migrates from one source host to another destination host while VM isstill in running state. In live migration, VM state, live data and data at rest is copied from source to destination. Pre and post copy VM migration techniques copylive data and data at rest in their continuous stages. Pre and post copy techniques generate heavy network traffic in short amount of time that ultimately leads to network congestion and packet loss.This further increasestotal migration time. In this paper, we have proposed a modified post copy approach called Shared Folder Based Migrationwhich generates less network traffic and reduce total migration time.

Keywords: VM Migration, Shared Folder Approach, KVM.

I. INTRODUCTION

Cloud computing is a computing paradigm where platform, scalable resources, data storage and IT services are provided over the internet [1]. Cloud computing has grown in the full pace in last few years. Cloud computing is gaining popularity amongst big, medium and small scale industries because of its pay-per-use model which is very economical. Cloud computing allows industries to fulfill their computing needs by providing them on-demand resources like networks, storage, servers and physical machines etc. The requirementsof clients are change dynamically according to their needs. The cloud platform dynamically adjusts according to client requirements. Cloud services are therefore, provided with minimum interruption by service providers.

In 2011, NIST [2] defined cloud computing as a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. Cloud computing provides many service oriented models but their base models remain the same, namely, Software as a Service(SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS). In SaaS model, software is provided to the clients over the internet. These software's are virtualized and mostly use web browser interface to execute. PaaS is a model where operating system is delivered to clients over the internet. The requested OS platform and its associated tools are bundled together. Then it is transferred to clients over the internet. These OS images and tools are machine independent. Clients can install their application on this guest OS without actually installing operating system. Clients do not need to worry about system's minimum requirement and compatibility issues of software with OS. In IaaS, client can avail complete network as a service which includes virtual machines, virtual server and virtual network etc.

[*] Ph.D. Student, Applied Science Deptt. (Compter Applications), I. K. Gujral Punjab Technical University, Kapurthala (Punjab), India, *E-mail: yadav.yashveer@gmail.com*

[**] Professor, Dept. of Computer Science and Engineering, National Institute of Technical Teacher Training and Reseach, Chandigarh, India, *E-mail: ramakrishna.challa@gmail.com*

In cloud computing, resources are provided on pay-per-use basis. These resources are more virtual than physical. In cloud computing,virtualization works on the concept of divide and distribute. In the rapid growth of cloud computing new players have joined the market. The key players in cloud computing are Rack Space, Amazon Web Services, Dimension Data, Apple, Cisco, Citrix, IBM, Google, Microsoft and Salesforce.com. Amazon was the first major cloud service provider whichprovided services with the name of Amazon Simple Storage Service (Amazon S3). Now, there are thousands of cloud hosting companies providing different cloud services to clients.

Cloud service providers are facing new challenges, they need to give maximum service to their clients withminimum cost.To reduce the cost, the maximum use of existing resources is desired with minimum response time. So cloud service providers need to balance both the maximum utilization of resources as well as have to give better cloud experience to clients.

The computing need of client fluctuates, sometimes they require high computing power and sometimesvery less computing power. To balance this fluctuating need of computing power, cloud service providersmigrate the VMsfrom high computing servers to low computing servers according to computing needs. Virtual machine migration also requires because of load balancing, hardware maintenance, energy reduction and dynamic resizing to increase system availability.

Virtual machine migration is a process in which virtual machine is transferred from one physical host to another physical host.Cloud provider tries to minimize any of three performance metrics, i.e, total migration time, total down time, and network load. For some cloud applications, minimumdowntime is of utmost importance [3] [4], while for others, optimizing network resource consumption is necessary [5] [6]. Additionally, VM migrations may have to meet a constraint on acceptable downtime [7].

Virtual machine migration can be of two types'offline migration and live migration. In offline migration, the virtual machine isshut down and image file of virtual machine is transferred from source physical host to destination physical host. In this migration, no vCPU or vRAM states are copied. In live migration, running instance of virtual machine is transferred from source physical host to destination physical host. The state of vCPU and vRAM are copied and dirty pages are also copied using iterative copy paging. The two most common techniques for live virtual machine migration are pre-copy and post-copy.

1.1. Pre-Copy Migration Technique

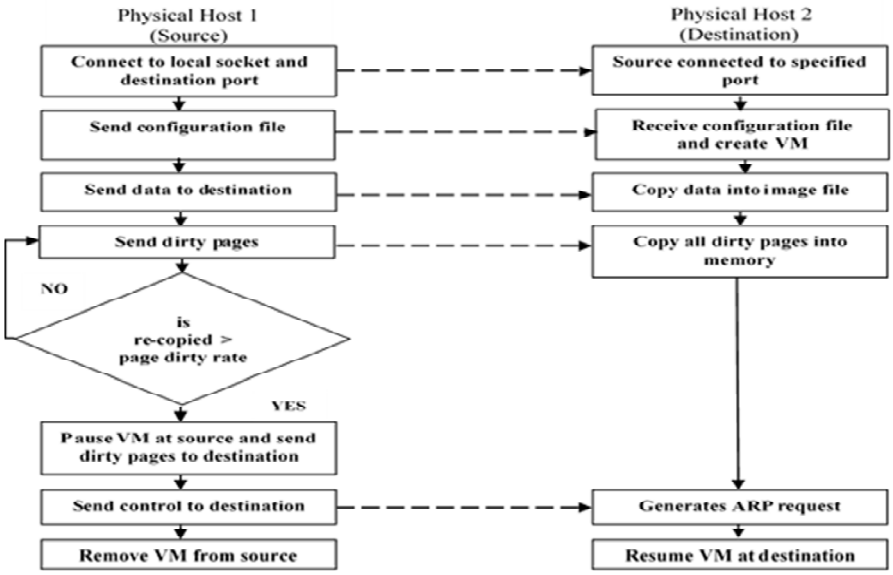

Figure 1: Pre-Copy Migration

As shown in Fig. 1, in pre-copy migration, virtual machine remains in running state while migration process starts in background. In the first attempt, the hypervisor copies all the memory pages from source host to destination host. Virtual machine is still in running state so there aresome pages which have been modified since the last copy to destination. These pages are called as dirty pages. They are re-copied and sent again to destination until the rate of re-copy is not less than the dirty page rate. All of the dirty pages are copied from source to destination and then virtual machine is stopped at the source.The most recent dirty pages are copied and control is transferred to destination host. The destination host generates an ARP request and assigns new IP address to the VM and resumes its functioning at the destination. When the VM is resumed at the destination then the VM is removed from the source. The time between pausing and resuming the virtual machine at destination is called down time. The time between starting the migration and ending the migration is called migration time.

1.2. Post-Copy Migration Technique

The drawback of the pre-copy migration is that it copies the data first and then resumes its functioning. In cloud environment, normally migration is required because of lack of resources on physical host. In pre-copy,VM keeps running on same physical host that increases the total migration time. To overcome this limitation inpost-copy migration, first it copies the minimum pages and thenresumes the VM at destination.Rest of the data is copied at a later stage as shown in Fig. 2. Sometimes it happens that client requests for a page which is still in process to be copied to destination. This condition is called as page fault or network fault. In post-copy migration, VM first resumes its work and rest of the data coping is done later. This method reduces the migration time but generates the risk that if source machine gets down before copying the complete data, the data may be lost for forever.

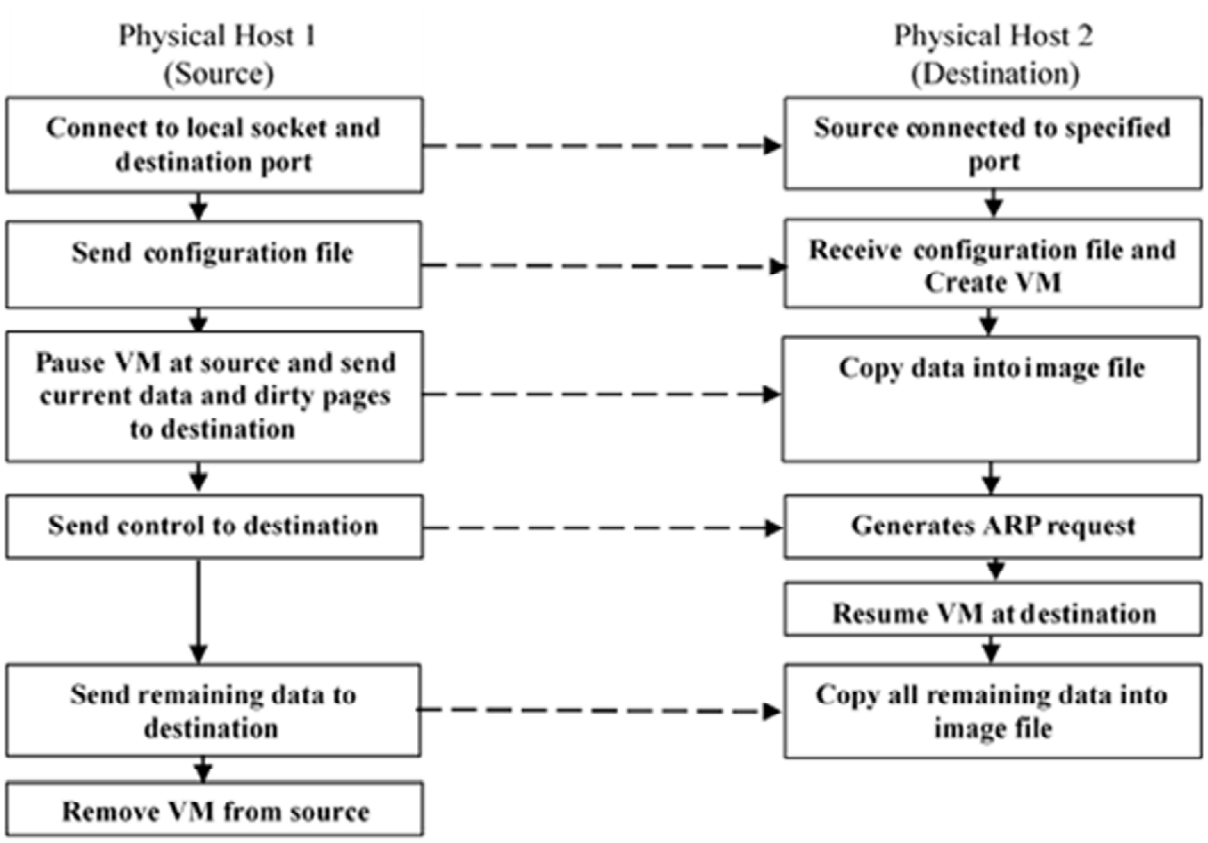

Figure 2: Post-Copy Migration

II. RELATED WORK

Live migration of VM needs to be done without any noticeable interrupt time. Several approaches are used to do so. C.Clarket et. al. [8] and Netson M. et al.[9] proposed pre-copy migration which copies current data from source host to destination host in number of iterative sessions. Data at rest is copied first and then current data is copied in iterative sessions until page_threshold_value>dirty_page_rate. This technique keeps using the same physical host until the complete migration has not taken place. To overcome this drawback, M.R. Hines et al.[10] and T. Hirofuchi et al.[11] proposed post-copy migration technique. Post-copy migration technique copies current data first then transferscontrol to destination host and data at rest copied later. The draw back with post-copy is that, it loses control on source host in very initial stage and if due to any reason source host gets down, data is lost forever. Auto-converge[12] is a new tool released under KVM hypervisor. Auto-converge puts some vCPU of VM on sleep so that dirty paging of data can be minimized. The one drawback of auto-converge is that it directly affects the performance of VM. To reduce the dirty paging rate and network traffic, KVM also released another tool called as Xor Based Zero Run Length Encoding (XBZRLE) [12]. It compares the previously sent pages with the current pagesandit pin point the pages with the most recent changes and sends them to destination. The drawback in XBZRLE is it uses additional vCPU and memory to compare and store the pages. V. Deshpande et al. [13] introduced scatter and gather technique, whichuses intermediate nodes between source and destination to scatter the data which uses post-copy approach. Scatter gather technique has two phases.In first phase, data is scatterd between intermediate nodes and control also sent to destination host with current data. In second phase, data is gathered from the intermediate nodes and transferred to destination host. J. Francisco et al.[14] proposed adaptive downtime for VM migration. They introduced the concept of dynamic calculation of downtime. They calculated the downtime first and then enter in stop and copy phase of VM migration. R. Cziva et al.[15] has proposed model of Software DefinedNetwork(SDN) based VM management. They have made modification in the switches and make them more compatible to support migration data. Their approach modifies the environment in which data flows. The approach does not make any change in any stage of VM migration. Results are also based on changes in environment of data flow.

The summery of the above said techniques/tools with respect to methodology, advantages and disadvantages/limitations are listedin Table 1.

Table 1
Findings from the Literature

Virtual machine migration technique / Tool	*Methodology*	*Advantages*	*Disadvantages/ Limitations*
Pre-Copy[8][9]	VM transferred to destination first and then resume its work	Sends the complete data first and then only resume working at destination	Total migration time is high because VM keeps running at same source until all dirty pages have not been transferred
Post-Copy[10][11]	VM resumes at destination with minimum required data and remaining data copied in later stages	Total Migration time is less than pre-copy migration	Any fault at later stage leads to data loss or corrupt the image file of VM
Auto-converge (KVM tool)[12]	In this technique some vCPU's are put on sleep so that dirty paging can be minimized	It reduces the dirty paging, whichalso reduces the total migration time due to less number of iterative re-copied attempt	Put some vCPU's on sleep that degrades the performance of VM. Client able to see the degradation in performance of VM

contd. table 1

Virtual machine migration technique / Tool	*Methodology*	*Advantages*	*Disadvantages/ Limitations*
XBZRLE (KVM tool)[12]	Compare the previously sent dirty pages with the current dirty pagesand send only the pages with the changes	Pages with the changes sent to destination. It speeds up the migration process. It also reduce network traffic	Comparison and sending only changes in dirty pages required more space and more vCPUs
Scatter Gather Technique [13]	Spreads data in between nodes from source and client. In gather phase this data is transferred to destination	Data is stored in between nodes so it takes comparatively less time to reach at destination	Any intermediate node failure can lead to data loss
Adoptive downtime technique(ADT)[14]	ADT had modified KVM live migration. They copy dirty pages in sequence of round and calculate the downtime as (*num_pages*)* ((*pending_pages*)+1)* (*page_threshold*)	This technique calculates the downtime in dynamic manner before entering the stop and copy phase. So it reduces downtime of VM.	Downtime calculation method is not appropriate for on-line application
Software Defined Network (SDN) based orchestration framework[15]	SDN has used S-Cross algorithm to reduce the communication cost from source to destination. In this framework they have suggested to make changes in switches to support the live migration	SDN based orchestration framework made changes in switches which are places between source and target. The dedicated switches have significant reduction in congestion and increase overall throughput by over six times. Results show 70% cost reduction by migrating less than 50% of the VM	SDN based orchestration framework has focused the work on how to make network switches more compatible with migration traffic. Their implementation doesn't modify or change any stage of standard migration process. Results are obtained only by changing the network environment. In some environments it is not possible to make changes in switches to support only one type of traffic

From literature survey, we have found that the large amount of data migrates from source to destination in form of Image file. This migration creates heavy network load and delay. Client normally does not work on whole data at the same time but whole data is transferred from source to destination. In shared folder based migration approach, we propose to transfer only VM state with current data. The data whenever is required in later stage is transferred on demand basis only. This will restrict to send whole data to destination and saves the time. It and also reduces network traffic because selected or required data is only transferred. In this paper, to overcome above mentioned limitations, authors have proposed shared folder based migration technique.

III. PROPOSED SHARED FOLDER BASED MIGRATION TECHNIQUE

In traditional migration systems, VM is migrated from source to destination with all data and VM state. This migration process includes transfer of current state of VM, current data, i.e., VM state, current data and data at rest (i.e., data which is not in process). Normally data at rest is much more thanthe current data and a VM image consists of OS and current data. Both pre-copy and post-copy techniques copy both OSand data at rest from source to destination. In pre-copy, the data at rest is copied first and then current data is copied. On other hand, inpost-copy current data is copied first and then data at rest is copied. In both

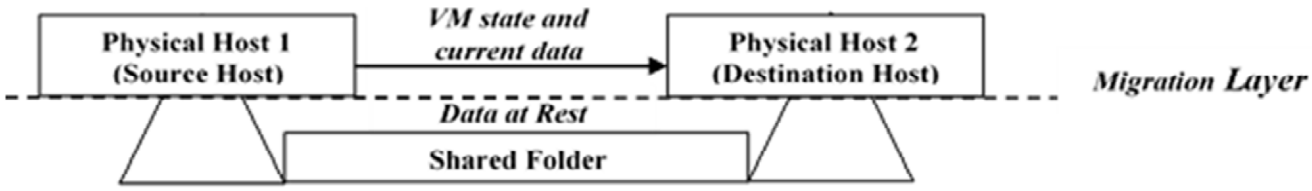

Figure 3: Shared Folder Based Migration Approach Migration

techniques, data at rest is copied from source host to destination host which takes longer time and thus increases the total migration time.

In our approach, only the current state of VM and current data is transmitted. Data at rest is stored in a shared folder as shown in Fig. 3. We store the image file in place where it is accessible by the source as well as by the destination.

We have used post copy migration technique in which we first pause the VM at source and then resumes VM at destination with minimum system requirements and current data. We do not copy data at rest because data at rest is already stored in shared folder as shown in Fig 3. Wherever data at rest is required, system copies only required block of data from source to destination. In initial stage, it copies only current data and the requested data is copied in later stages. The file which is not required will remain at the source. The working of our proposed technique under different scenarios depending on user requests for data is as under:

No data requested after VM migration:In thisscenario, it is assumed that the client will not request further data after the VM has been migrated from source to destination. In this scenario, our proposed approach copies the current data and VM state at the time of VM migration from source to destination. As the client makes no further request for the data from source host, hence no further network traffic is generated. Thus,it reduces load on the network. As fewer amounts of data are transferred during VM migration so the overall migration time is also reduced. In both pre-copy and post-copy VM migration techniques, whole data, i.e.,current data, VM state and data at rest are transferred from source to destination during the VM migration. Therefore, it generates heavy load on network. As an illustration assume, a client using VM with Ubuntu 14.04 LTS as OS and working on a file of size 200 MB with the total user data of 2 GB(let us assume 10 Files of different sizes). Using our proposed technique, current state of VM with 200 MB of data on which client is working will be migrated.Rest of the data will not be migrated. On the other hand, pre-copy and post-copy techniques will require migrating entire 2 GB of data with VM state.

Partial data requested after VM migration: In this scenario, after VM migration, clientmakesfurther request for data from data at rest. In this, our approach copies only the requested files. The remaining data will be stored at the source only. It will not be migrated till the next request of client. Thus, it generates less load on network therefore reduces total migration time. In both pre-copy and post-copy VM migration techniques, whole data, i.e., current data, VM state and data at rest are transferred from source to destination during VM migration. As an illustration assume, after VM migration with 200 MB of data, now client requests for 100 MB of data.The requested file of size 100 MB will be migrated from source to destination. Assuming 2 GB of data, out of which 200 MB was already migrated during VM migration and additional 100 MBis now requested by user.Therefore, it saves network overhead by 1.7GB (i.e., 2-(0.2+0.1) =1.7) as compared to the existing techniques where entire 2GB of the data is transferred.

Full data requested after VM migration:In the third scenario, the client generates multiple requests to copy all of the remaining data after the VM has been migrated from source to destination. The requests by client are made one after the other at different times. Using our approach, the client makesmultiple requests for files as per the requirement from source to destination. So, according to the request of the client file will be migrated from source to destination. In worst case scenario, all the files will be requested by client but

at different time span. So, entire 2 GB of data will be transferred but in different time span. This will not overload the network and willlead to negligible loss of packets. In both pre-copy and post-copy VM migration techniques, whole data (i.e., 2 GB including current data, VM state and data at rest)is transferred from source to destination during VM migration. Thus, it generates heavy network traffic at the same time. This leads to network congestion and ultimately results in packet loss. The lost packets will have to be re-transmitted which further worsens the situation. Re-transmission of lost packets increases delay which further increases the total migration time.

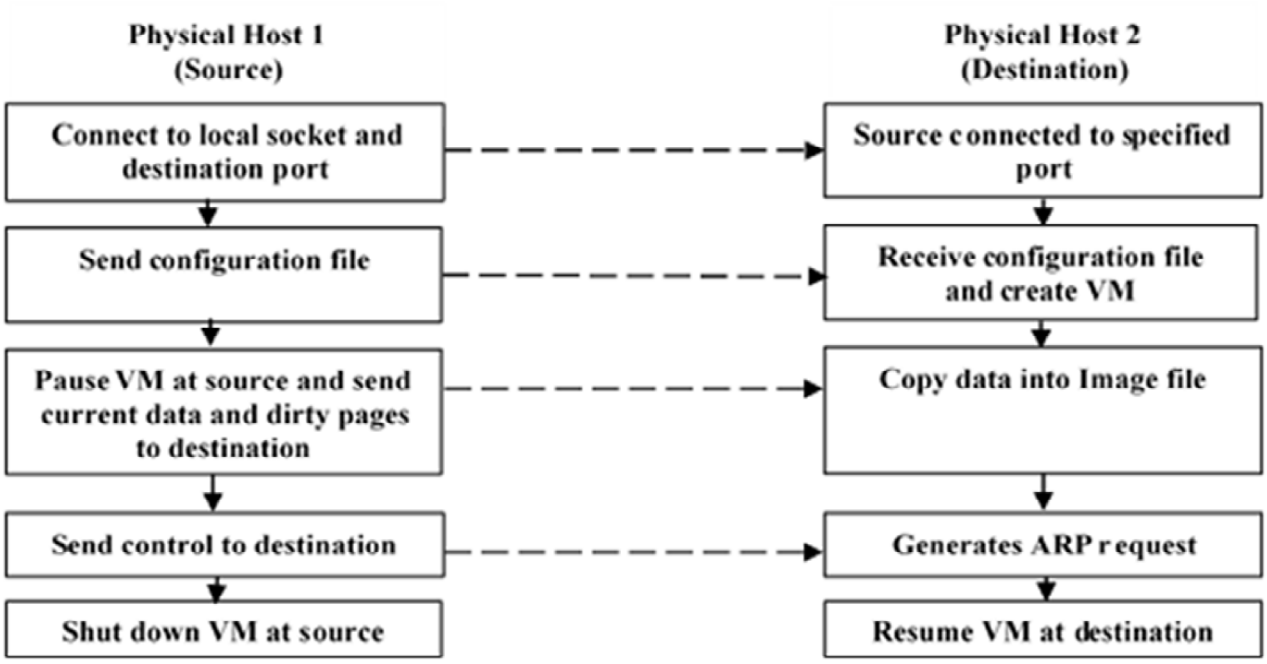

Figure 4: Conceptual Flow Diagram for Shared Folder Based Migration Approach

IV. CONCLUSION

Various authors have analyzed the performance of pre-copy and post-copy techniques for live migration of VM. Howeverthere are drawbacks in both of these techniques such as high down time, data loss and performance degradation. To tackle these drawbacks, in this paper we have proposed shared folder based migration technique. In our technique, data is logically shared between physical hosts which uses modified post copy migration technique. We first pause the VM at source and then resume VM at destination with minimum system requirements and current data. Data at rest is not copied because it is already stored in logically shared folder which is accessible to both source and destination hosts thus saving total down time and reducing network traffic. In future, we plan to extend the scope of study by addressing problem of selecting victim VM at source to be migrated. Also, an efficient method for selection of destination host for victim VM will be proposed.

REFERENCES

[1] A.V. Dastjerdi and R. Buyya, "An Autonomous Reliability-Aware Negotiation Strategy For Cloud Computing Environment", *12th IEEE International Symposium on Cluster, Cloud and Grid Computing, Ottawa*, ON, pp. 286-291, May 2012.

[2] NIST (National Institute of Standards and Technology), *TheNIST Definition of Cloud Computing. America: Special Publication* 800-145, 2011.

[3] S. Akoush, R. Sohan, A. Rice, A. W. Moore, and A. Hopper, "Predicting the performance of virtual machine migration," in *Proceedings Modeling, Analysis & Simulation of Computer and Telecommunication Systems (MASCOTS)*, 2010, pp. 37–46.

[4] M. Alicherry and T. V. Lakshman, "Optimizing data access latencies in cloud systems by intelligent virtual machine placement," in *Proceedings IEEE INFOCOM*, 2013, pp. 647–655.

[5] U. Mandal, M. F. Habib, S. Zhang, M. Tornatore, and B. Mukherjee, "Bandwidth and routing assignment for virtual machine migration in photonic cloud networks," in *Proceedings European Conference and Exhibition on Optical Communication (ECOC)*, Sept. 2013.

[6] U. Mandal, M. F. Habib, S. Zhang, M. Tornatore, and B. Mukherjee, "Heterogeneous bandwidth provisioning for virtual machine migration over SDN-enabled optical networks," in *Proceedings OFC*, March 2014.

[7] V. Mann, A. Gupta, P. Dutta, A. Vishnoi, P. Bhattacharya, R. Poddar, and A. Iyer, "Remedy: Network-aware steady state VM management for data centers," *NETWORKING 2012*, pp. 190–204, 2012.

[8] C. Clark, K. Fraser, S. Hand, J. Hansen, E. Jul, C. Limpach, I. Pratt,and A. Warfield, "Live Migration of Virtual Machines," in *NetworkSystem Design and Implementation*, 2005.

[9] M. Nelson, B. H. Lim, and G. Hutchins, "Fast Transparent Migrationfor Virtual Machines," in *USENIX Annual Technical Conference*,2005.

[10] M. R. Hines, U. Deshpande, and K. Gopalan, "Post-copy Live Migration of Virtual Machines," *SIGOPS Operating System Review*,vol. 43, no. 3, pp. 14–26, 2009.

[11] T. Hirofuchi and I. Yamahata, "Yabusame: Post-copy Live Migration for Qemu/KVM," in *KVM Forum*, 2011.

[12] A. Shah, "Live Migrating QEMU-KVM Virtual Machines", Available at: *http://developers.redhat.com/blog/2015/03/24/live-migrating-qemu-kvm-virtual-machines/*, accessed on (24-06-2016).

[13] U. Deshpande, D. Chan, S. Chan, K. Gopalan and N. Bila,"Scatter-Gather Live Migration of Virtual Machines", in *IEEE Transactions on Cloud Computing*, 2015 (Article in Press).

[14] F.J. Clemante-castle, Juan Carlos Fernández,R. Mayo and E. S. Quintana-Ortí," Adaptive Downtime for Live Migration of Virtual Machines",in *7th IEEE International Conference on Utility and Cloud Computing*, 2014, pp 457-464.

[15] R. Cziva, S. Jouet, D. Stapleton, F.P.Tso and P.Pezaros,"SDN Based Virtual Machine Management for Cloud Data Center", in *IEEE Journal on Network and Service Management*, 2015 (Article in Press).