HomeCloud Framework for Priority aware VM Allocation and Network Bandwidth Provisioning

Chintoju Sidhardha¹, V.Srinivas², M.Gopal³, Rajesh Mothe⁴, Komuravelly Sudheer Kumar⁵

¹,²,³Sumathi Reddy Institute of Technology for Women, Warangal, India.
⁴,⁵SR Engineering College, Warangal, India.

¹sidhardhachintoju@gmail.com

Abstract: Various applications in a multi-tenant architecture based cloud computing comprises various requirements and priorities. Severe failure of a critical application may result if the network is congested in a data center. Hence, it is an important to use an efficient resource provisioning technique for ensuring transfer of high priority data to the other traffics still when the network is congested. Therefore, a new framework of Home Cloud architecture is proposed in this paper to provide an effective and flexible delivery of new applications in an edge-cloud architecture which integrates the two complementary technologies of Software-Defined Networking (SDN) and Network Function Virtualization (NFV). In SDN infrastructure, this presents a priority aware resource placement algorithms by taking into consideration of both host and network resources. The Virtual Machines (VMs) of high priority applications are placed at the strongly connected hosts by introducing the priority aware VM allocation (PAVA) algorithm for reducing the possibility of causing network congestion by other tenants. It also guarantees the critical applications for a required bandwidth by means of allocating bandwidth through the setting of priority queues for each network connection within the network of data center handled by SDN controller. The simulation result of the proposed system shows that the combination of proposed approaches will allocate sufficient resources for high-priority applications within a Home Cloud data centre to meet the QoS requirements.

Keywords: HomeCloud, Software-Defined Networking (SDN), Network Function Virtualization (NFV), virtual machines (VMs).

1. Introduction

The cloud computing [12] technology is utilized to provide commercial and flexible services by the most of latest Internet applications [1]. These different types of applications to the tenants are served by means of the various requirements by the provision of computing, memory, storage, and networking resources from Cloud providers. There is a need of various amounts of resources for those applications with different priorities. Excess of memory power and computing is needed compared to the networking resources for the compute-intensive applications of large scientific application whereas excess of networking bandwidth compared to computing power is required for intensive applications.

In order to satisfy different requirements in the data center, the Cloud providers must require effective allocating and provisioning resources. Furthermore, ensuring the requirements of various levels of Quality-of-Service (QoS) and reliability with the Service Level Agreement (SLA) is guaranteed to the customers by the cloud providers. That means in the cloud resource management, highly strict requirements are needed for the QoS critical applications (higher priority) deadline-
limited scientific applications for IoT applications for real-time disaster management or medical software for cyber surgery and where as the normal requirements are needed for the normal applications (lower priority). Cloud providers must efficiently provision the resources with various applications together with various QoS requirements which share the similar data center to meet the various QoS requirements for various applications. The software-based NFs replaced by the dedicated hardware of NFs using Network Function Virtualization (NFV) [2] happening on a generic compute resources can result in providing the cloning, transfer of data and further NF management for satisfying Service Level Agreements (SLAs) along with accommodating of dynamic traffic changes. In the meantime, a general method of supporting NFVs is provided by the Software Defined Networks (SDN) as it can dynamically direct the flow of traffic if required. A lot of interest is attracted from the both of academia and industry towards the proposed architecture of SDN + NFV also called as Software-defined NFV which is combination of that two (SDN and NFV) techniques.

One of the biggest challenges to achieve a dynamic NF management is compliance with adapted SLAs. For an instant, the processing states of original flow or packet must be cloned perfectly and absolutely throughout the cloning. Certain states installed during the original instance operation are generally not visible to the control system (SDN controller or NFV creative force). They are referred to as NF internal states and are lost in the newly created context. It is therefore unable to process the initial flows, correctly and results in network errors such as flow inconsistencies. This type of error condition, which can lead to an incorrect processing flow / packet state, is known as state violation. A state violation often arises in delivery of dynamic NFs and it is difficult to notice. Only after careful data control can it be recognised by the administrator or user.

2. Literature Review
Various methods are proposed for the placement of network-aware VMs and scheduling of SDN-based network flow. A Maple network-aware VM placement system which develops a predicted usage of bandwidth for VM is proposed by Wang in [3]. According to the experimental observations gathered from the servers, the authors of this Maple system computed the predicted usage of bandwidth. The approach of First Fit Decreasing is used in this Maple system to find out a host for the placement of VMs. This Maple method focused on network requirements for each VM and considers that all VMs have similar processing requirements. In addition, this Maple Project is further progressed to Maple Scheduler in which scheduling system of flows dynamically rescheduled the network flows depending on QoS requirement of each VM. Depending on the estimated usage of bandwidth, flows are relocated using the dynamic network flow scheduler. This device initially discovers a standard route with the Equivalent Multi Path Cost (ECMP) protocol, a widely used multi-path protocol, that also distributes network traffic equally across multiple pathways. Then the potential flows which can violate QoS are detected and an alternate path which satisfies the QoS is rescheduled in the system.

Another approach to the placement of VMs which is on the basis of QoS and energy efficiency is the Energy-efficient QoS-aware Virtual Machines Placement (EQVMP) [17]. In this EQVMP [4] approach the load in every one of the group is balanced as well as traffic of inter groups is reduced by partitioning the virtual machines into groups according to the traffic matrix. The energy consumption and number of required hosts are reduced by placing the virtual machine groups on hosts with the help of bin packaging algorithm. Links which were overloaded are detected across the network after the placement by performing load balancing [18] and then relocate that links in an alternate links. It is assumed that virtual machine’s capacity requirements are homogeneous and only consider the network requirements. The idea of providing tight SLA or QoS network control and service has been proposed by numerous existing NFV approaches of SIMPLE [7], Stratos [10], FlowTags [5], and connection acrobatics [6]. However, they focused only on controlling and coordinating the forwarding of traffic. Furthermore, as mentioned above, internal states are necessary to obtain a tight SLA or QoS service, but few works to address proprietary solutions have been suggested.
In 2003, the significance of states is explained by the VM or process replication. The potential errors generated by them are avoided since it permits the instance cloning totally and absolutely. Based on only the increased use states, still no one of network administrator and no one of user know about what accurately these states are. Subsequently, additional works proposed to offer NFs the possibility of moving, copying and sharing NF states, defined as Vendor-supplied controllers. They developed additional modules such as connecting NFs to expose and share states. However, states vary between NFs that means each NF has a proprietary abstraction, which makes network administration extremely complicated. NF control planes/systems have been developed such as Split/Merge [8], Pico Replication [9] and OpenNF to achieve complete network control with available internal states. They managed to gain some control over the internal states of NF and the states of the network. Split / Merge and Pico Replication have developed communal records that NFs can be used to create and modify the access of the internal state using a predefined API and Open NF API for the southern and northern bands along with a control module to achieve the goal in the network control system.

3. Proposed Architecture
A new framework of HomeCloud architecture along with its building blocks and the interaction between the building blocks in an edge-cloud atmosphere is presented in this paper to deliver new portable applications within a shared edge infrastructure in an efficient and effective manner. This was accomplished by consistently integrating two up-and-coming and opposite technologies of Network Software-Defined Networking (SDN) and Function Virtualization (NFV). SDN is integrated here with the implementation of Priority aware VM analysis that uses priority analysis along with the bandwidth analysis of edge-cloud network and it also suitable for networking, configuring, controlling and managing the Virtualized Network Functions (VNFs) in the edge-cloud working together with NFV. While in edge environment NFV is implemented for local computing, storage, and the future new applications which generate huge amounts of data and require low latency networking to place them near to the edge mobile devices.

3.1 Home Cloud Framework
Integration of SDN and NFV is the main thought of Home Cloud based environment. Various organizations conducted the research at early stages on the standardization, development, application of SDN and NFV. In this perspective, for achieving the overall idea of proposed method, the biggest challenge faced is how the NFV and SDN can be allowed to coexist and work consistently. This consistency method facilitates various researchers for deeply understand both of the SDN and NFV and although synergistically both of them developed quickly and independently. The home cloud corresponds to this is a big effort. The illustration on SDN and NFV ecosystem infrastructures is especially provided, that it is divided in to two associated domains called as stakeholders domain and technical domain. The stakeholder’s domain needs Internet Service Providers (ISP) and Application Service Providers (ASP) for good progress in this new framework [16] to mutually work together for getting mutual gain. In addition, ISPs and ASPs are highly motivated to keep their existence investments and the mechanism related to co-existence transition is important. The domain technique signifies the synergy and interaction between SDN and NFV, also the delivery of new applications for customers can be facilitate by automated management and orchestration functions.

Various technologies are used to intersect those two types of domains by connect them consistently. In particular, those used technologies are centralized control or configuration or programmability offered by SDN or virtualization offered by NFV. The proposed frame work, building blocks and the interactions between those building blocks are depicted in Figure (1) in a detailed manner. It mainly consists of two domains called SDN and NFV. The lower layer infrastructure of NFV contains the virtualized compute, network and storage which use a hypervisor layer working together with the hardware resources. In such context, NFV must require to provide network functions to the VNFs for creating virtual networks [14] to some particular new applications. Same as that of Physical network,
the virtual network also exactly works except with high flexibility, scalability and re-programmability.

Moreover, SDN layer contains three major layers of priority aware VM analysis, SDN controller and physical resources which are used for effective delivery of new feature applications. Elements corresponding to this SDN infrastructure are the VNFs in the NFV infrastructure. The SDN controller is specifically responsible by taking into consideration of controlling, configuring, monitoring and programming mechanisms for the VNFs. The SDN controller consists of three layers called application layer, control layer and data layer. There are several new applications to be implemented on the SDN controller. The Northbound Interface (NBI) of SDN is placed between the application and control layers where as Southbound Interface (SBI) of the SDN is placed between the control and data layer. This total operation is managed by the platform of application management and orchestration controller. It converts the Service Level Agreements (SLA) of ASPs for the new applications into executable schemes and attains the idea about further portability of applications.

3.2. Efficient Orchestration and Application Delivery:
In general, the providers and owners of cloud are massive societies and fundamental conventional cloud is proprietary. In the given "game changer" of the upcoming edge cloud, the main idea of proposed HomeCloud framework is that it breaks the monopoly network and promotes innovation to build an open and consistent platform. An approach called northbound is required for an application orchestration and delivery in the edge cloud frame work to convert the high level objectives of the SLAs which differ considerably from those of the centralized conventional cloud into the actual implementation of the application. A network scenario example for deploying these HomeCloud applications is described as follows. In this scenario, the edge clouds have multiple applications and runs NFV by virtualizing resources of compute, network and storage. For each of the new applications, there is at least one Virtualized Application Server (VAS) at the edge and more than for virtual reality applications, which must be data-intensive, processing-intensive or coordinated.
Generally, the NFV infrastructure launches the Virtualized Application Servers that are normally VNFs and which are considered as entry servers for users of application. In the edge cloud, this Virtualized Application Server also accesses virtualized computing, networks and edge cloud storage. The structure of NFV comprises a sequence of individual logical operating blocks: they are controller nodes, network nodes and computing node series. Generally launching of this Virtualized Application Servers is done closer to the users of an application. A variety of controllers and servers are there for HomeCloud framework. Specifically, the orchestration servers are between the owners of the application and the actual delivery process of the technical application for satisfying the requirements of the application owners. Later than lunching the Virtualized Application Servers, The accountability of the SDN controller is for monitoring, configuring, controlling and maintaining the Virtualized Application Servers. OpenFlow is used for the SDN controllers. Application controllers are the final set and the common functions of these controllers are maintaining the application level topologies, monitoring and controlling of functional entities at application level. In addition, interfaces among various functional components and interactions of internal logical functions are illustrated.

The main functions and requirements of some of the basic components are briefly listed and discussed as follows:

i. **New application Providers**: There are generally multiple ASPs which contains a deep understanding of customer needs and application plans are designed to meet them.

ii. **Application Orchestration Server**: It converts ASP requests into scripts that allocate and manage specific resources by synchronizing them with other servers and controllers.

iii. **Application controller**: It tracks, monitors and manages topology and the application level entities. Virtualized Application Servers are the one of the application-level entities.

iv. **NFV infrastructure Servers**: There are generally virtualization controllers assigned to and handled the various resources such as compute, network and storage in the edge cloud and those dynamically launch Virtualized Application Servers.

v. **SDN Controller**: It is responsible to control, configure, monitor and programming of the VAS launched in order to form a functional overlay network for a specific application. They are connected to VAS via protocol (typical example is OpenFlow) and network control interfaces.

### 3.3 Priority Aware VM Allocation (PAVA)

Priority-Aware VM Allocation (PAVA) algorithm is proposed for VM allocation with the information on network topology to allocate closely connected hosts to the critical new applications. The application’s priority and the connectivity of physical hosts on a network within a datacenter are taken into consideration for this PAVA method. This system first collects the data center’s network topology prior to the allocation of VM. According to the network connectivity each and every one of hosts within a data center are grouped. That means the hosts will be grouped together which corresponds to the same rack and which linked with the same edge switch. Also VMs are grouped and as per the application. The VM host mapping which is suitable for placing the VMs to critical applications is found after the grouping of hosts by running the PAVA algorithm.

The placement of set of VMs can be done by PAVA algorithm for critical applications into the host group which comprises of several networking and computing resources. Some virtual machines are placed in multiple host groups at nearest location i.e. VMs placed in the same pod under a different edge switches when the set of virtual machines does not fit into a single host group [17-23]. Thus the minimum number of hops will be maintained for network transmission by closely placing the VMs in
the same application. For example, the network traffic among the VMs placed in the same host is relocated by using host memory without occurring of any traffic for networking devices. In the same way, if the virtual machines are hosted on the same pod or in the same edge network then the network transmission cost among these virtual machines can be reduced with low probability of other applications interfering. PAVA not only assures the virtual machines for critical applications which are placed on the host with adequate capacity, but also assures the VMs on the same or closest hosts in the network topology for reducing the cost of communication.

3.4 Bandwidth Allocation for Priority Applications (BWA)

The applications of QoS [19] to provide a stable network bandwidth becomes an important in traffic congestion to evade the performance degradation caused by other tenants since different tenants share the same network infrastructure of cloud data center. Hence, bandwidth allocation (BWA) approach is presented to critical applications for allocating the necessary bandwidth. A feature of flow traffic management is used by the above approach for the virtualized network. The SDN controller can monitor the switches for allocating the necessary bandwidth by scheduling the prioritized waits on the switches to make sure that privileged traffic is transmitted over other low priority traffic. Once this VM allocation procedure is completed, frequency requirements as well as virtual network specifications are submitted to the SDN controller for a critical application. Then the SDN controller sets the priority queues (example: Linux disc and Hierarchy Token Bucket) for crucial application flows along the path at each switch. Internet traffic caused by virtual machines of critical applications uses the prioritized waits to get the bandwidth desired for critical applications.

3.5 Dynamic Flow Scheduling (DF)

The congested link which is overloaded with the data of new application is detected by the method of dynamic flow scheduling in multipath network topology and spillovers flows through dense traffic paths and subsequently that flows by the more capacity from congested link to an alternative link. Since the architecture of fat trees has been further developed and an over-subscription ratio of the network can be achieved up to 1:1, according to the observation even though this dynamic flow scheduling method is less effective for short-range flows in the topology of fat trees. The total number of downlinks of a connected host in a fat tree corresponds to the total number of uplinks for aggregation switches in edge switch environment. The transmission lines are uniformly dispersed according to the origin’s host address, destiny. Thus the network traffic between the links is always balanced among the edges and the grouping switches on same pod. Since the number of links between the edge and aggregation switches are more than that of links between aggregation and core switches, this algorithm still efficient for inter pod traffics. DF algorithm can be used to replace the flows if a link is congested to a core switch into an alternate link of leaving less traffic to the other core switches. This DF approach is considered as guideline for comparing with the approach of allocated frequencies. The proposed approach is used regularly to find the route [13, 14] with the lowest occupancy for traffic with the highest priority, while normal traffic continues to use the standard route set by Equal multi path cost protocol based on the source address.

4. Results

Simulation measurements of the proposed home cloud environment with that of cloud only environment is shown in the Figure (2) and Figure (3). The average response time of critical applications in their data centers [15] is measured and shown in the Figure (2). The average response time of an application can be deliberated as a time taken by the request send by the end-user to the application server through virtualized network functions until the response reaching the end-users. The virtualized network functions are used here to forward both of these requests and responses. As shown in Figure (2), the average response time is decreased by 27.3\% in Home cloud environment compared to that of Cloud-only environment from 1.1263sec to 0.8231sec. This reduced network latency is achieved with the virtualized network functions that are placed into the edge resources closest to the
end users.

Fig. 2: Average response time COMPARISION of CLOUD ONLY AND Home cloud environment

![Average response time comparison](image1.png)

Fig. 3: processing time of proposed network and existed network

![Processing time comparison](image2.png)

Also the processing time of virtual machine and network transmission at application server are measured and depicted in the Figure (3). The processing time of virtual machines at application server is correspondingly calculated at nearly 0.0093sec in a network transmission time. In addition, the proposed algorithm impact on the critical applications compared to the other algorithms is calculated in terms of average response time of higher-priority applications. The measured average response time of critical applications are shown in Table (1) for different algorithms.
Table 1: Average response time of different algorithms

| Algorithms                  | Average response time (sec) |
|-----------------------------|-----------------------------|
| Random Algorithm            | 6128.30                     |
| PAVA                        | 5674.47                     |
| PAVA+BWA                    | 5674.55                     |
| PAVA+BWA+DF (Proposed algorithm) | 4382.70                 |

From the table it can be concluded that the proposed PAVA, PAVA+BWA and DF algorithms had an improved performance for a higher-priority application with contrast to Random algorithm with a reduction of average response time. Briefly, even the performance of lower-priority applications can be maintained or improved by the proposed algorithms with the improved performance of critical applications.

Fig. 4: Comparison of Average response time for different algorithms

5. Conclusion
A new frame work of Home cloud architecture that integrates both SDN and NFV infrastructures is proposed in this paper to provide a flexible and efficient delivery of new applications in edge-cloud network by introducing a network traffic management method and priority based VM allocation followed by bandwidth allocation and dynamic flow path method. The main aim of this proposed frame work is to crack monopoly and permit small and medium sized ASPs to innovate in edge cloud with low cost and high elasticity. From the simulation results it can be shown that both the average response time and energy consumption are reduced for critical new applications since they are placed into the strongly connected hosts with the help of priority-aware VM allocation method. Moreover, in the case of network congestion, where other applications interfere with higher-priority traffic, the bandwidth allocation is particularly an efficient method. The proposed method is capable of dealing with the varying priorities and QoS requirements of various applications.

6. References
[1] S V Rossem, M Peuster, L Conceio, H R Kouchakhsaraei, W Tavernier, D Colle, et al., "A network service development kit supporting the end-to-end lifecycle of NFV-based telecom services" IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN) pp 1-2 2017
[2] M Peuster H Karl and S van Rossem "MeDICINE: Rapid prototyping of production-ready network services in multi-PoP environments" IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN) pp 148-153 2016

[3] R Wang, R Esteves, L Shi, J A Wickboldt, B Jennings and L Z Granville, “Network-aware placement of virtual machine ensembles using effective bandwidth estimation” in Proceedings of the 10th International Conference on Network and Service Management (CNSM) and Workshop Nov 2014 pp 100–108

[4] S-H Wang, P P W Huang, C H P Wen, and L C Wang, “EQVMP: Energy-efficient and qos-aware virtual machine placement for software defined datacenter networks” in The International Conference on Information Networking 2014 (ICOIN2014) Feb 2014 pp 220–225

[5] S K Fayazbakshsh, L Chiang, V Sekar, M Yu, and J C Mogul, “Enforcing network-wide policies in the presence of dynamic middlebox actions using flowtags” in 11th USENIX Symposium on Networked Systems Design and Implementation (NSDI 14) 2014 pp 543–546

[6] C Nicut, C Paasch, M Bagnulo, and C Raiciu, “Evolving the internet with connection acrobatics” in Proceedings of the 2013 workshop on Hot topics in middleboxes and network function virtualization, ACM 2013 pp 7–12

[7] Z A Qazi, C-C Tu, L Chiang, R Miao, V Sekar, and M Yu, “Simplefying middlebox policy enforcement using sdn” ACM SIGCOMM computer communication review vol 43 no 4 pp 27–38 2013

[8] S Rajagopalan, D Williams, A Warfield and H Jamjoom, “Split/ merge: System support for elastic execution in virtual middleboxes” in Presented as part of the 10th USENIX Symposium on Networked Systems Design and Implementation (NSDI 13) pp 227–240 2013

[9] H Jamjoom, D Williams and S Rajagopalan, “Pico replication: A high availability framework for middleboxes” in Proc. of the 4th ann. Symp. Cloud Computing, ACM 2013

[10] A Gember, R Grandl, A Anand, T Benson, and A Akella, “Stratos: Virtual middleboxes as first-class entities” UW-Madison TR1771 p 15

[11] Seena Naik K. and Sudarshan E., 2019 Smart healthcare monitoring system using raspberry Pi on IoT platform, ARPN Journal of Engineering and Applied Sciences, 14(4) 872-876

[12] R Buyya, C S Yeo, S Venugopal, J Broberg, and I Brandic, “Cloud computing and emerging it platforms: Vision hype and reality for delivering computing as the 5th utility” Future Generation Computer Systems vol 25 no 6 pp 599–616 2009.

[13] M Dobrescu, N Egi, K Argyraki, B-G Chun, K Fall, G Iannaccone, “Routebricks: exploiting parallelism to scale software routers” Proc. of the ACM SIGOPS 22nd symp. Oper. Syst. Prin. pp 15 2009

[14] Muntasir Raihan Rahman, N M Mosharaf, Raouf Boutaba and Kabir Chowdhury, “Virtual Network Embedding with Coordinated Node and Link Mapping” IEEE INFOCOM 2009.

[15] I Stoica, A Tavakoli and D A Joseph, “A policy-aware switching layer for data centers” in ACM SIGCOMM Computer Communication Review volume-38 numbr-4. ACM pp 51–62 2008

[16] Bikku T., Rao N.S and Akepogu A.R., 2019 A novel multi-class ensemble model based on feature selection using Hadoop framework for classifying imbalanced biomedical data, International Journal of Business Intelligence and Data Mining 14(43832) 25-39 10.1504/IJBIDM.2019.096801

[17] Sudarshan, E., Satyanarayana, C. and Bindu, C.S., 2006 A Compact Parallel Huffman Entropy Coding Technique On GPGPU Using CUDA

[18] R Zhang-Shen and H Liu, “On direct routing in the valiant load-balancing architecture” Glo. Tele. Confe. 2005. GLOBECOM’05. IEEE volume- 2 pp-6 2005

[19] Mahender K, Anil Kumar T and Ramesh KS 2017 SER and BER performance analysis of digital modulation schemes over multipath fading channels Journal of Advanced Research in Dynamical and Control Systems 9(2) 287-291
[20] Rajasri I, Gupta AVSSKS and Rao YVD 2013 A Comparative Study of a Few Tests for Isomorphism in Planetary gear trains *International Journal of Engineering and Technology Innovation* 3(2) 95-103

[21] Mounika, S. & Sampath Kumar, T. 2019 "Client –requirement fulfil QOS in multi server for max beneficial in cloud computing" *International Journal of Advanced Science and Technology* vol 28 no 7 pp 44-49

[22] Sudarshan E, Naik K.S, Kumar P.P 2020 Parallel approach for backward coding of wavelet trees with CUDA. *ARPN Journal of Engineering and Applied Sciences* 15(9), pp.1094-1100

[23] Sripada N.K., Sirikonda S., Kumar N.V and Siruvoru V., 2019 Support vector machines to identify information towards fixed-dimensional vector space *International Journal of Innovative Technology and Exploring Engineering* 8(10) 4452-4455 10.35940/ijitee.J9826.0881019