Queuing analysis of dynamic resource allocation for virtual routers

M. Said Seddiki, Nefzi Bilel, Ye-Qiong Song, Mounir Frikha

Queuing analysis of dynamic resource allocation for virtual routers

M. Said Seddiki † *, Bilel Nefzi *, Ye-Qiong Song *, Mounir Frikha†
† Higher School of Communications of Tunis, University of Carthage, Tunisia
* LORIA Research Laboratory, University of Lorraine, France
†{seddiki.said, m.frikha} @supcom.rnu.tn
*{bilel.nefzi, ye-qiong.song} @loria.fr

***Abstract*—The most critical issue in network virtualization is the dynamic resource allocation of the physical substrate. There is a need to monitor the running virtual routers in order to allow an adaptive change in the resource allocation. In this paper, we focus on the router data plane virtualization and we explore this issue by presenting a new dynamic allocation approach through queuing theory. We consider the problem where multiple instances of virtual routers (VRs) that have some quality of service (QoS) requirements are sharing different physical resources. We propose a novel router architecture that offers a strong isolation between concurrent VRs and provides a dynamic allocation scheme in order to guarantee the provided QoS to each of them. Our approach aims at providing a higher isolation for the concurrent virtual routers sharing the same infrastructure. We propose a dynamic Weighted round robin (WRR) scheduler for each physical resource and an algorithm for adjusting the weight of each VR in order to reduce the delay of the packet processing and avoid the bottlenecks. We also propose an admission control mechanism that estimates the current load of the physical node and decides either to accept or reject a creation request of a new VR. Simulation results show that the proposed approach achieves good performance in terms of delay minimization inside the virtual router.**

***Keywords*-Network virtualization; Dynamic resource allocation; Queuing theory; Weighted round robin**

I. Introduction

Network virtualization represents the abstraction of a physical network, where its resources are partitioned and utilized by multiple isolated virtual networks (VNs). Due to the dynamic workload of networks, managing the virtual node and the virtual link is a difficult task for the infrastructure provider [1]. However, progress in network technologies has made it possible to provide on-demand physical network resources for multiple service providers with different requests. Dynamic and static approaches to resource allocation have been proposed in the literature [2]. In the dynamic allocation, the resources are allocated as needed, and are released when they are no longer needed. In the static allocation, the virtual router receives a small fraction of each resource and cannot ask for more even if it is needed.

This study addresses the issue of node allocation in network virtualization. We believe that this allocation should be dynamic, in order to ensure higher performance and optimal utilization of the physical infrastructure. Through this approach, we aim at minimizing the packet processing delays that occur when multiple isolated virtual networks are sharing the limited resources on the same physical node.

Queuing theory provides an interesting mathematical tool to manage problems involving waiting lines [3]. This theory provides a good model for analyzing the physical router, where a packet needs to be processed by multiple pipelined resources from the router's input port to the output port.

The main difference between the existing approaches and our work is that they virtualize only the control plane without providing any isolation between all the flows of the concurrent VRs at the data plane. Isolation is crucial constraint in Network virtualization. This constraint ensures that each VR is completely independent of all other VRs running on the same physical router. In this paper, we present a fine-grained efficient resource allocation in the router's data plane that supports the isolation of the VR traffic flows. In our study, the input flow for each VR is maintained in a separate queue in order to isolate each VR's flow. We also consider that each resource has a dynamic Weighted round robin scheduler and the weights depend on the current workload of the physical infrastructure and the QoS requested by the VR. We suppose that every resource is a multiple input queue with a single server Weighted round robin (WRR) queuing system, where the service rate of the resource is deterministic and the arrival rate is an exponential distribution with λ as a parameter. At every instant, we attempt to find the optimal weight for each VR, in each resource, with the QoS constraint consideration in order to prevent under-utilization of the physical infrastructure. The proposed allocation scheme aims at providing a better utilization of the physical resources in order to optimize the packet processing delays inside the router.

The remainder of the paper is organized as follows. In section II, we expose the background and related work. In section III, we present our router architecture and we introduce our approach for admitting a new request for creation of a new VR. We also propose an algorithm for dynamic resource allocation for multiple VRs. In section IV, the simulation results of our approach are presented and discussed. Finally, in section V, we conclude this paper and summarize our findings.

* This work has been partially supported by ANR Quasimodo (under No. ANR 2010 INTB 0206 01)

II. Background and related work

Network virtualization is a new technology that aims at allowing the sharing of a physical network infrastructure among multiple virtual networks. One of its advantages is to isolate logical environments from each other [4]. Resource allocation in network virtualization has been addressed in many studies in order to provide better utilization of the physical substrate. Most of the proposed approaches offer a heuristic solution, since the problem is NP-hard [5].

There have been a number of efforts which have approached this issue by proposing a new router's architecture that allow the dynamic change of the fraction of the physical resources received by each VRs according to their workloads.

Bozakov [6] presented an flexible architecture for virtual routers that transparently manipulates the forwarding tables of a set of distributed forwarding elements allowing them to be operated as a single entity. The author also proposed an embedding algorithm for the optimal allocation of resources with minimum allocation cost as a flow network problem. The proposed approach offers the live migration for VRs data plane to quickly adapt resources to changing demands.

McIlroy and Sventek [7] proposed a router architecture which consists of a number of virtual machines, called QoS Routelet, which are assigned to route a single QoS flow. A proportion of the available physical resource is allocated to each network flow requiring QoS guarantees. The proposed approach ensures that each routelet can only access its allocated resources in order to prevent from the interference with the others routlets while they are processing their flows. The authors claimed that the proposed approach allows the partitioning of a routers resources between flow streams and avoids the over-provisioning. The proposed approach offers a good dynamic resource allocation between multiple flows in terms of latency, inter-packet jitter and single flow throughput but does not support a strict isolation between the flows in the data plane.

Choi et al. [8] developed a user-space load aware virtual router monitor to deploy VRs virtual routers atop a commodity multi-core architecture, where each of its component supports different implementation. The proposed work offers the ability to manage the dynamic resource allocation of the VRs based on their data traffic loads. The approach supports the load balancing between the cores inside the physical router and provides good performance in terms of throughput and latency.

Egi et al. [9] also proposed a design of a new platform for virtual routers on commodity hardware that is mainly driven by performance and flexibility for packet processing. The authors use Xen virtualization technology to host the guest domains for packet processing and forwarding in order to indentify the performance bottlenecks associated with the currently available commodity hardware. The proposed design supposes that the routers graph organization can be decomposed into a series of forwarding trees. It provides high packet forwarding rates and support the isolation of each packet flow.

Bourguiba et al. [10] also explored the resource allocation over commodity hardware. The authors proposed a new mechanism based on packets aggregation techniques for transferring packets between the driver domain and the virtual routers. They presented an analytical model of the new packets transferring mechanism that determines the maximum achievable throughput taking into consideration only the packet delay constraints. In a previous work [11], we studied the dynamic resource allocation issue from different perspectives in order to control the workload and avoid service degradation. We modeled this issue as a non-cooperative game and we presented a prediction-based algorithm that seeks the best strategy to take that maximizes the utilization of router's resources.

Most of the current commercial routers provide a static partition of the physical router resources into multiple logical routers, each of which has its own configuration [12]. This partition does not support any flexibility to customize their resource management. Many researchers addressed this issue without taking into consideration the isolation between the concurrent VRs at the router's data plane. By presenting the related works we show that our paper provides a different point of view to the dynamic resource allocation for virtual routers. Our work aims at providing a strict isolation between the flows that are processed by the physical resources inside the router. The objective of this research is to propose a new router architecture that offers a dynamic change of the current resource allocation according to an estimation of the current workload of each VR. This architecture needs an hypervisor to create and run the VRs. This hypervisor is also responsible of managing the execution of the VRs and the partition of the physical resources between them.

III. Router architecture and dynamic resource allocation scheme

The objective of the proposed architecture is to provide a higher level of isolation for each instance of VR. Every physical resource, at a fixed instant, seems to be totally dedicated to a VR. The shared routers resources can be either the computational resources in network processor architecture or the internal or the external memory such as SRAM or a DRAM. For example we can consider each of the Task Optimized Processors (TOPs) of an Ezchip network processor [13] as a single resource. These processors are programed in a dedicated assembly language and offer a parallel packet processing. They are responsible a specific task such as parsing, searching, resolving, and modifying the packet to perform the IP lookup. A packet traverses these resources in order to be processed from the ingress port to an egress port. The main functionality of a router is to perform this task. The ingress line card prepares the packet in order to be forwarded by the switch fabric. Then the switch fabric is responsible for forwarding the packet to the egress line card that contains the output port. The switch fabric does not perform any packet processing and usually has a queue for the packets that are waiting to be forwarded.

We consider there are N virtual routers competing for M resources, such as processing engine and memory. We assume that each physical resource maintains N separate infinite

queues, each queue corresponding to each flow of a VR. Each virtual router VR_i is fed by an arrival stream for which packet inter-arrival time is exponential, with rate λ_i. We assign each VR a global weight, Gw_i, which depends on QoS requested. The higher is the requested QoS, the greater is the global weight of the VR. The virtual routers' requests are subject to an admission control that is based on the current load of the physical infrastructure and the stability condition of the queues in all the resources. The admission control decides whether to admit the request of a VR and prevents from the deterioration of the performance of the active VRs. Once a new VR is admitted, the dynamic WRR algorithm assigns, every interval, a new weight in each resource that is proportional to its average waiting packet and the sum of all the average waiting packets of all the active VRs using that resource.

A. The proposed router architecture

The main functionality of a router is to forward packets between different networks. It consists of three main components: the line cards, which physically connect the router to different networks; a control processor, which builds the routing tables and runs routing protocols and the backplane, which connects the line cards and the control processor together. In our study, we use a switch fabric as a backplane in order to allow packets to traverse the backplane in parallel and to increase the workload that a router could cope with.

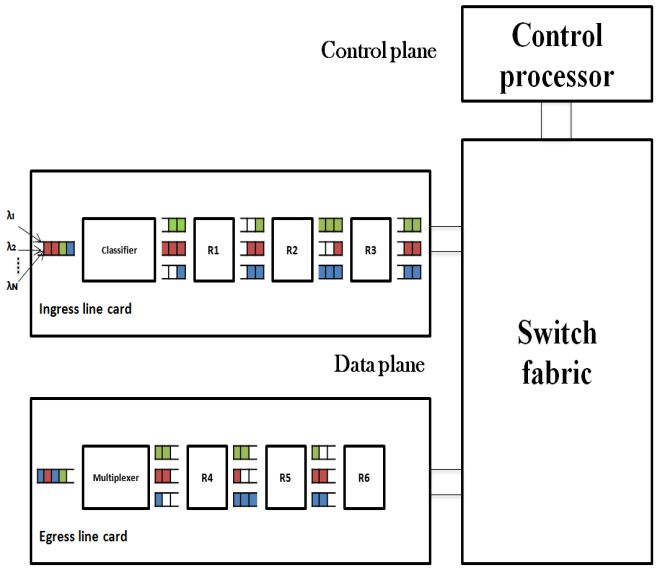

Fig. 1: The physical router architecture

In our architecture, as shown in Figure 1, the physical router maintains a different infinite queue for each flow coming to each VR. In all the physical resources, we have a WRR packet scheduler in which the weights are dynamic according to the number of waiting packets in the queues.
In the ingress line card, the classifier sorts the incoming packet flow from the input port into different flows for the different VRs. The function of the multiplexer in the egress line card is to combine the flow from different VRs into the same FIFO queue to the output port. By proposing this router architecture, we attempt to provide a fair and better usage of the physical resource according the the mean of allocated fraction of each resource while ensuring to provide a higher level of isolation for all the concurrent VRs. In this work, we consider that the multiplexer and the switch fabric are fast enough to process the packets without a need to queue. We also suppose that the packet processing delays by the multiplexer are equal to zero. The classifier is a multiple input queue with a single-server queuing system with an exponential service time. The packets enter to the system according to a Poisson process with a rate parameter λ equal to the sum of all the input rates of all the active virtual routers. The packets are queued on a first-come first-served basis in order to be processed by the classifier to determine which virtual router's flow they belong to.

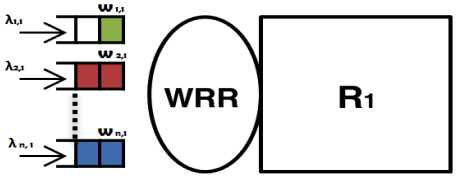

Fig. 2: The multiple queues and the single WRR sever of the shared resource R_1

The departure process of a stable queue with an exponential service rate and arrival rate λ is a Poisson process rate. Thus, the first resource R_1 is considered as a queuing system with multiple queues and a single sever. The queuing system is illustrated in Figure 2. In the system, the arrival rate is exponential and the service rate is deterministic. The arrival rate in each queue is equal to the sum of all the input rate λ_i divided by the number of active VRs. Arriving packets for each VR that follow an exponential distribution are classified and queued. The WRR scheduler serves the flows according to their weights $w_{i,1}$. The service time is deterministic and each VR receives a fraction of the resource. The weight of every VR_i is dynamically updated according to the algorithm that will be presented later in this paper.
The rest of the router's physical resources are considered a multiple input queues with a single-server queuing system and the WRR scheduler. In our approach, we consider the fact that packets are discrete units of data. We suppose that a packet's processing will not be interrupted to begin processing another packet.

B. The admission control mechanism

In this section, we develop an admission control algorithm that will be used to limit the number of active VRs. It is performed at the ingress port of the physical router. Whether the creation of a new VR is allowed depends on the capacity of the physical router's resources and the incoming workload for each VR.

The admission control mechanism is activated by the reception of a new creation request for a VR. We suppose that

the system already knows the maximum arrival rate for each VR. The creation request for a new VR is accompanied by information about the maximum arrival rate requested. The system computes a fraction based on all the maximum arrival rates, the number of active VRs, the capacity of each resource, and the capacity of the classifier. When the fraction is greater than one the request is rejected; otherwise the request is accepted. The admission controller decides to accept a new instance of a VR according to the queue stability condition [14] in each queue of the system. Algorithm 1 presents the different steps of the admission control algorithm.

Algorithm 1 Admission control algorithm

Input Virtual router VR_1, VR_2,..., VR_n
Resource R_1, R_2,..., R_m
Resource's capacities C_1, C_2,..., C_m
Classifier R_c
Classifier's capacity $C_{classifier}$
Weight $w_{i,j}$
Number of active virtual routers k
Maximum input rate λ_i^{max}

Begin
For each $req_{i'}$ = creation of a new $VR_{i'}$ ($\lambda_{i'}^{max}$) do
For each resource R_j **If** $\frac{\sum_{i=1}^{k} \lambda_i^{max} + \lambda_{i'}^{max}}{k * C_j} < 1$ and $\frac{\sum_{i=1}^{k} \lambda_i^{max} + \lambda_{i'}^{max}}{C_{classifier}} < 1$
then admits $VR_{i'}$
$w_{i',j} = \frac{\sum_{i=1}^{k} w_{k,j}}{k+1}$
else
rejects $VR_{i'}$
$w_{i',j} = 0$
end If
end For
end For
end

C. Dynamic resource allocation algorithm

This work aims at developing multiple schedulers for all the router's physical resources based on Weighted round robin. This scheduler is able to predict the requirements for different flows of packets that need to be processed by different VRs. The goal of this work is to provide higher isolation for different VRs and allocate the physical resources in a fair, adaptive, and efficient way.

We suppose that the capacity of a resource j with a WRR scheduler is C_j. If all the concurrent VRs are active, then each of them receives a fraction of that capacity. The minimum guaranteed service rate is given by the assigned weight for each VR and is equal to:

$$C_{i,j}^T = \frac{w_{i,j}^T * C_j}{\sum_{k=1}^{N} w_{k,j}^T}$$

We use the exponential moving average to estimate the current input rate for each VR in each resource at *T* instant. The estimation of the input rate is given by the following equation:

$$\bar{\lambda}_{i,j}^T = (1-\alpha) * \bar{\lambda}_{i,j}^{T-1} + \alpha * \lambda_{i,j}^T$$

Since the classifier is considered a multiple input queue with a single-server queuing system with both exponential service rate and arrival rate, the estimation of the arrival rate at the system is given by the following equation, where k is the number of active VRs:

$$\bar{\lambda}_c^T = \bar{\lambda}_1^T + \bar{\lambda}_2^T + ... + \bar{\lambda}_k^T$$

In our study, the resource R_1 is considered a queuing system with multiple queues and a single sever. The service rate is deterministic and the arrival rate is exponential. The estimation of the input arrival to the system is equal to the estimation of the input at the classifier divided by the number of active VRs. It is equal to:

$$\bar{\lambda}_{i,1}^T = \frac{\bar{\lambda}_c^T}{k}$$

In order to update the weights of each virtual router VR_i in the resource R_1, the dynamic WRR algorithm uses an estimation of the average number of packets in each queue at T instant according the estimation of the input packet arrival. Let $A_{i,j}^T$ be the average number of waiting packets in the queue related to the virtual router VR_i at *T* instant in order to be processed by the resource R_1. According to Pollaczek-Khinchin formula [15], it is defined as:

$$A_{i,1}^T = \frac{(\bar{\lambda}_{i,1}^T)^2}{2\, C_{i,1}^T * (C_{i,1}^T - \bar{\lambda}_{i,1}^T)}$$

The packet delay inside the router has a big impact on QoS. In our work, we try to minimize this delay in order to provide good performance for each VR. The average packet delay in the classifier is given by the following equation [16]:

$$D_{i,C}^T = \frac{1}{C_{i,C}^T - \bar{\lambda}_{i,j}^T}$$

We define as $D_{i,1}^T$ as the average delay for a packet for a given virtual router VR_i spent in the resource R_1 at *T* instant. This delay is equal to the average waiting delay plus the average service delay of the resource. The average packet delay in a resource R_1 is given by the following equation [17]:

$$D_{i,1}^T = \frac{1 - (\bar{\lambda}_{i,1}^T / 2 * C_{i,1}^T)}{C_{i,1}^T - \bar{\lambda}_{i,1}^T}$$

The sum of all packet delays in all resources is equal to the delay of a packet inside the whole physical infrastructure. This delay is the time spent by the VR to process a packet from the input interface to the output interface.

Every control interval each virtual router VR_i receives a new weight in the resource R_1 equal to:

$$w_{i,1}^{T+1} = \left(\frac{Gw_i}{\sum_{k=1}^{N} Gw_k} + \frac{A_{i,1}^T}{\sum_{k=1}^{N} A_{k,1}^T}\right) \frac{C_1}{2}$$

For the rest of the resource, at *T + 1* instant every virtual router VR_i receives a new weight equals to:

$$w_{i,j\neq 1}^{T+1} = \left(\frac{Gw_i}{\sum_{k=1}^{N} Gw_k} + \frac{C_{i,j-1}^{T}}{\sum_{k=1}^{N} C_{k,j-1}^{T}}\right)\frac{C_{j\neq 1}}{2}$$

The different steps of the algorithm for the dynamic weight assignment are presented in Algorithm 2. The algorithm ensures that every instant, each VR receives a new weight for each resource that is proportional to its global weight, the average length of its queue, and the average length of all the queues in our queuing system.

Algorithm 2 Dynamic weight assignment algorithm

Input Virtual router VR_1, VR_2,..., VR_n
Resource $R_1, R_2, ..., R_m$
Capacities $C_1, C_2, ..., C_m$
Global weight Gw_i
Weight at T instant $w_{i,j}^{T}$
Average waiting packet $A_{i,j}^{T}$
Exponential moving average $\bar{\lambda}_{i,j}$
Output Weight at T + *1* instant $w_{i,j}^{T+1}$

Begin
For each virtual router VR_i do
For each resource R_j do
Compute $(\bar{\lambda}_{i,j}^{T})$
If $j = 1$
$$w_{i,1}^{T+1} = \left(\frac{Gw_i}{\sum_{k=1}^{N} Gw_k} + \frac{A_{i,1}^{T}}{\sum_{k=1}^{N} A_{k,1}^{T}}\right)\frac{C_1}{2}$$
else
$$w_{i,j\neq 1}^{T+1} = \left(\frac{Gw_i}{\sum_{k=1}^{N} Gw_k} + \frac{C_{i,j-1}^{T}}{\sum_{k=1}^{N} C_{k,j-1}^{T}}\right)\frac{C_{j\neq 1}}{2}$$
end If
end For
end For
end

IV. Simulation Results

In this section, we perform simulations to validate our approach. We use Matlab as the simulation tool. We try to simulate the resource allocated to each VR. We consider 3 virtual routers (VR_1, VR_2, VR_3) sharing 4 resources (R_1, R_2, R_3, R_4), inside a single physical router, with different capacities and with poisson arrivals at rate λ_1, λ_2, λ_3. We suppose that these resources are computational resources, such as a network processor. In every stage of the simulation, a VR receives a fraction expressed in packets per milliseconds. We assign 3 global weights (Gw_1, Gw_2, Gw_3) for the 3 virtual routers. We assume that $p_1 > p_2 > p_3$, and the assigned weights to the VRs depend on the level of the QoS requested. We use the dynamic weight in order to dynamically update the weight of each VR every control interval that is equal to 1ms. We compare the results obtained by the static, where only the global weight is used, and dynamic Weighted round robin when the arrival rate is the same as the input rate. The performance of our simulation is expressed in terms of packet delays, the number of waiting packets in different queues of the system, and the efficient usage of the router resource.

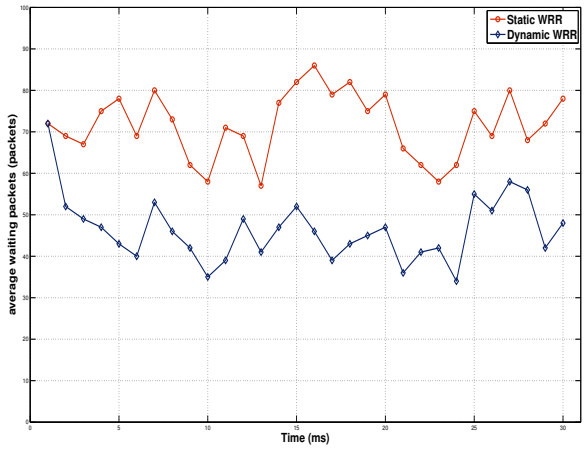

Fig. 3: The average waiting packets for VR_1 in the resource R_1

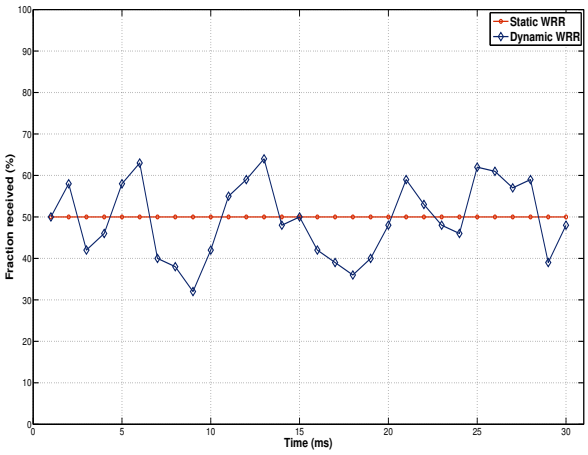

Fig. 4: The fractions of the resource R_1 received by VR_1

As shown in Figure 3, the dynamic WRR achieves better results than the static one in terms of number of waiting packets for the virtual router VR_1. Indeed, the dynamic algorithm updates the weight of each VR depending on the number of waiting packets in its queue and the sum all of the waiting packets in all the queues waiting to be processed by the resource j. In fact, in the first iteration, the number of waiting packets is equal because both static and dynamic weights are equal.
Figure 4 illustrates the fraction of R_1 received by the VR_1. It's very interesting to use the dynamic weight algorithm to avoid the under-utilization of the physical resource. For example, at 9 ms, in the static WRR allocation, VR_1 receives a fraction

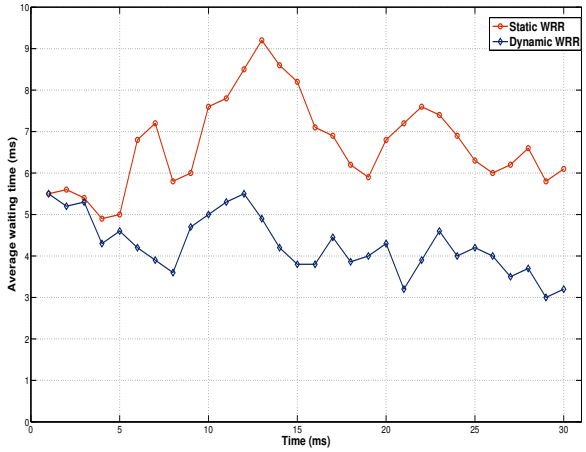

Fig. 5: The average packet delay for VR_1 in the resource R_1

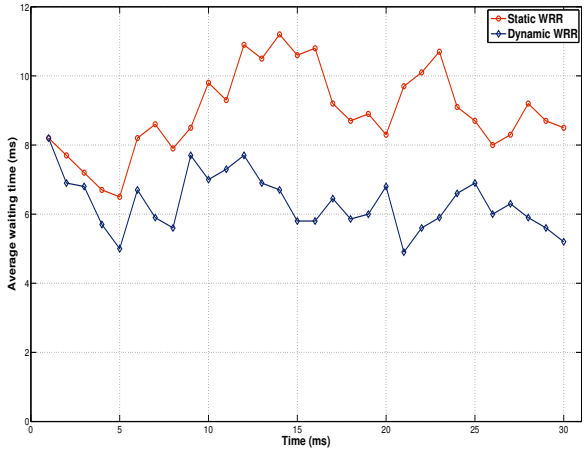

Fig. 6: The average packet delay for VR_1 in all the resources

equal to 50% of the capacity according to its weight. But in fact, it needs only 32% of this resource to process the packets waiting in its queue.

In order to simulate the total packet delay inside the physical router and compare the static and dynamic approaches, we suppose that the arrival rates of R_2, R_3, R_4 are uniform. Since the admission control maintains stability in all the queues in each resource, the queuing delays in R_2, R_3, R_4 are equal to zero. The packet delay in each resource is equal to the service delay of the queuing system.

Figure 5 shows the average packet delay for VR_1 in the resource R_1 and Figure 6 illustrates the average packet delay for VR_1 in the physical infrastructure. For a given single resource and for the whole router resources, the dynamic Weighted round robin achieves better performance than the static. In fact, the delay for a packet is minimized when the weights are not updated every instant according to the incoming workload for all the VRs.

V. Conclusion

In this study, we addressed the problem of dynamic resource allocation for multiple VRs sharing the same physical infrastructure. We proposed a novel router architecture that takes into consideration the strong isolation constraint between the VRs. We presented an admission control mechanism that aims at limiting the number of active VRs and to avoid performance deterioration. We also presented a dynamic weight algorithm that updates the weight of every VR to receive a higher fraction and to minimize delay of its packets to be processed from the input interface to the output interface. We have further conducted simulation experiments to validate our approach. As a future study, we will explore other distributions of packet inter-arrival rates at each VR that could be more realistic for Internet-like traffic [18]. We also plan to implement and evaluate the proposed approach in a real environment and discuss its performance.

References

[1] N. M. M. K. Chowdhury and R. Boutaba, "A survey of network virtualization", *Journal Computer Networks*, Volume 54, Issue 5, 2010, pp. 862-876.

[2] A. Haider, R. Potter and A. Nakao, "Challenges in Resource Allocation in Network Virtualization", In *proc. of 20th ITC Specialist Seminar*, 2009.

[3] B. J. Watson, M. Marwah, D. Gmach, Y. Chen, M. F. Arlitt, and Z. Wang, "Probabilistic performance modeling of virtualized resource allocation", in *Proc. of ICAC conference*, 2010, pp. 99-108.

[4] M. F. Mattos, L. H. G. Ferraz, L. H. M. K. Costa, and O. C. M. B. Duarte - "Evaluating Virtual Router Performance for a Pluralist Future Internet";, in *Proc. of ICICS conference*, 2012, pp. 1-7.

[5] A. Belbekkouche, M. M. Hasan, A. Karmouch, "Resource Discovery and Allocation in Network Virtualization", *IEEE Communications Surveys and Tutorials*, Volume: PP, Issue 99, 2012, pp. 1-15.

[6] Z. Bozakov, Towards Virtual Routers as a Service, in *Proc. of 6th GI/ITG KuVS Workshop on Future Internet*, 2010.

[7] R. McIlory and J. Sventek, "Resource virtualisation of network routers" , in *Proc. of IEEE HPSR workshop*, 2006.

[8] H.F.W. Choi and P.P.C. Lee, "An Extensible Design of a Load-Aware Virtual Router Monitor in User Space", in *Proc. ICPP Workshops*, 2011, pp.361-370.

[9] N. Egi, A. Greenhalgh, M. Handley, M. Hoerdt, F. Huici, and L. Mathy, "Towards high performance virtual routers on commodity hardware". in *ACM CoNEXT Conference*, 2008, pp. 20 -32.

[10] M. Bourguiba, K. Haddadou and G. Pujolle, "A Container-Based Fast Bridge for Virtual Routers on Commodity Hardware", in *Proc. IEEE GLOBECOM conference*, 2010, , pp. 1-6.

[11] M. S. Seddiki and M. Frikha, "Resource Allocation for Virtual Routers through Non-Cooperative Games", in *Proc. IEEE ICCCN conference*, 2012, pp. 1-6.

[12] Cisco Systems, "Introduction to Cisco ASR 9000 Series Network Virtualization Technology", 2011, Available : http://www.cisco.com/

[13] Z. Guo-sheng, Y. Shao-hua, and Y. Jin-you, "Design and implementation of scalable IPv4-IPv6 internetworking gateway", in *Proc. of the SPIE*, Volume 7137, 2008, pp. 713705-713705.

[14] H. L. Liang and V.G. Kulkarni, "Stability condition for a single-server retrial queue", *Advances in Applied Probability*, Volume 25 , Issue 3, 1993, pp. 690-701.

[15] P. Pochec and W. Mardini, "Modelling with queues: an empirical study," in *Proc. IEEE CCECE conference*, 2001, pp. 685-689.

[16] P. Hajipour, N. Amani, A. Dehestani, M. Mazoochi , "Measurements and Analysis of M/M/1and M/M/c Queuing Delay Models of the Two IP-PBXs in Various Remote Location", in *Proc 6th ICNC conference*, 2009, pp. 1-7.

[17] B. C. Shin and C. K. Un, "Performance Analysis of a Quasi-M/D/1 Cut-Through Switching Network with Noisy Channels", *IEEE Transactions on Communications*, Volume 34 , Issue 9, 1986, pp. 882-889.

[18] A. Adas, "Traffic models in broadband networks", *IEEE Communications Magazine*, vol.35, no.7, 1997, pp.82-89.