Resource Allocation for Network Slicing in WiFi Access Points

Matías Richart*†, Javier Baliosian*, Joan Serrat†,
Juan-Luis Gorricho†, Ramón Agüero‡ and Nazim Agoulmine§
*University of the Republic, Uruguay
†Universitat Politècnica de Catalunya, Spain
‡University of Cantabria, Spain
§Université d'Evry Val d'Essonne, France
{mrichart, javierba}@fing.edu.uy, serrat@tsc.upc.edu, juanluis@entel.upc.edu,
ramon@tlmat.unican.es, nazim.agoulmine@ufrst.univ-evry.fr

***Abstract*—Network slicing has recently appeared as one of the most important features that will be provided by 5G networks and is attracting considerable interest from industry and academia. At the wireless edge of these networks, most of the contributions in this area are related to cellular technologies leaving behind WiFi networks. In this work, we present a resource allocation mechanism based on airtime assignment to achieve infrastructure sharing and slicing in WiFi Access Points. The approach is simple and has the potential to be straightforwardly used within scenarios of wireless access infrastructure sharing.**

***Index Terms*—Wireless network slicing, wireless resource management, airtime allocation, 5G, WiFi.**

I. Introduction

Network slicing has recently appeared as one of the features 5G networks would eventually provide to achieve its objectives. It is indeed attracting considerable interest from research in industry and academia. With this approach, 5G will enable service providers to split network resources into *slices* with dedicated resources to be used by specific applications.

Network slicing is aimed to provide flexibility and efficiency to the operator as it permits to create dynamically and on-demand slices of resources to cope with specific services requirements. This is the key difference from existing similar proposals such as VPNs but a characteristic which makes it complex to deploy and manage. In addition, network slicing is very dependant on Software Define Networking (SDN) as it is a necessary technology to permit its management and deployment. SDN is indeed crucial to achieve the needed flexibility and programmability that is necessary for network slicing. As stated in [1], SDN and Network Function Virtualization (NFV) can be considered as fundamental enablers to network slicing, allowing its deployment and management.

Recent works on network slicing have mainly concentrated on the core of the network and on frameworks that enable slicing. Even more, most existing works consider slicing only for cellular networks and in particular for the LTE technology. On the contrary, our proposal is for the IEEE 802.11 (WiFi) technology, for which slicing has not been thoroughly studied. We focus in this paper on a slicing variant named *Infrastructure Sharing Slicing*. Its objective is to split and allocate network resources to slices proportionally to the requested share(see § II).

More precisely, we propose a novel mechanism for slicing a WiFi Access Point (AP) by considering the transmission time (airtime) as the resource to share. We propose a traffic queuing and airtime scheduling technique to split the AP transmission time and dynamically assign it to the different slices. The resource allocation achieved is efficient as each slice only receives the exact resources needed for its current load and freed resources can be used by other slices.

Network slicing is a new concept but many of its ideas have already been discussed extensively in the domain of network virtualization. Thus, some proposals for network virtualization can also be considered for network slicing. Some existing ideas share similarities with our proposal and are worth mentioning: [2] proposed an airtime resource allocation method through the management of the Contention Window (CW) size; [3] suggests to use traffic shaping to limit the use of resources of each slice and [4] proposed a queuing model with feedback control to guarantee throughput per slice. A more extensive review on existing proposals for slicing in WiFi can be found in our previous paper [1]. The idea of airtime scheduling in WiFi has been mainly studied as a means to overcome the *performance anomaly* [5]. As it will be explained in § III, our proposal adapts the airtime scheduling mechanism from a recent solution to the *performance anomaly* [6] so as to be used for network slicing.

Our proposal differentiates from previous works in that it does not need to control low-level MAC parameters, neither it needs feedback from the medium to achieve the required allocation. It only needs information on the airtime consumed, which can be obtained from the hardware driver. Furthermore, we do not do traffic shaping which, if not controlled properly, can led to waste of unused resources. Also, our queuing model takes into account the hardware behaviour to avoid queue buildup at lower layers and allows packet aggregation.

The rest of the paper is organized as follows: in § II we introduce our model of network slicing and define the concept of *Infrastructure Sharing Slicing*; in § III we describe our slicing mechanism; in § IV we show some experimental evaluation of our proposal and finally, in § V we present some concluding remarks and future work.

978-3-901882-98-2 © 2017 IFIP

II. Network Slicing

There seems to be no general definition of what is considered as a slice. Here we use a definition presented in our previous work [1]. We define a slice as a group of traffic flows which share some common characteristic. A traffic flow is a stream of packets which is identified by its source and destination. For example, a flow in the IP network is identified by the source and destination IP addresses and the source and destination ports. A slice is owned by an external entity (a *tenant*) and requires a fraction of network resources to be allocated. A slice can support flows of multiple *final users* (mobile clients of the network in our case), but at the same time, a final user can participate in multiple slices. However, a flow belongs to only one slice.

Some slice examples are: all the flows with certain type of device as source or destination (e.g., sensors); or the flows of a VoIP service; or, the flows from or to a user of a given operator. Depending on the specification of a slice, a final user can participate on different slices but, slices are always independent between each other.

A key aspect of slicing is *isolation*. Isolation consists on avoiding slices from disturbing each other, specifically avoiding performance degradation or the modification of allocated resources because of the behaviour of other slices.

In our opinion there exists two variants (or perspectives) of slicing, which differ substantially:

- *Quality of Service Slicing (QoSS):* slices offer different services and ensure Quality of Service within them.
- *Infrastructure Sharing Slicing (ISS):* similar to the traditional idea of network virtualization. There is a tenant (e.g. Mobile Virtual Network Operator), which is given a slice of the network. The tenant has complete control over the network infrastructure and network functions within the slice.

The major difference between these variants is that in QoSS it is required performance objectives for each flow in the slice while in the ISS the requirements are related to network resource allocations for the entire slice. Then, how to implement these two type of slices can vary significantly.

A. Wireless Infrastructure Sharing Slicing

The previous definition applies for slicing in general and for the entire network. However, as already mentioned, in this work we focus only on the slicing problem at the wireless edge of the network and in the ISS variant.

When instantiating the ISS variant in the wireless domain interesting challenges appear. As already mentioned, the ISS has many similarities with network virtualization where the slice requests do not specify performance objectives for the flows of the slice but demand resources to be allocated to the entire slice. In the wired domain it is common, for this approach, to allocate resources such as link bandwidth or CPU usage. This is possible because the available resources (as link capacity) are fixed and known. However, in the wireless domain the bandwidth of a link between the network and the final user is unknown as it depends on the wireless channel characteristics. This causes that sharing a resource such as link bandwidth in the wireless domain to be very complex. As a consequence, the possible resources to assign are limited and constrained to: the available radio spectrum (divided also in time, frequency or space) or the available transmission time (airtime).

The technique of allocating radio spectrum is vastly used in LTE slicing proposals, by extending the existent scheduling of LTE Physical Resource Blocks (PRBs) [7]. However, applying this approach to WiFi is not straightforward because of the differences on the usage of the radio spectrum. In WiFi there is no time or frequency division but instead each node transmits using a CSMA/CD approach utilizing the entire assigned frequency band. Therefore, for WiFi, transmission time sharing is a more appropriate solution. Airtime sharing consists on splitting the time each slice uses the medium, giving a fraction of the time to each slice. Nevertheless, this approach also presents its difficulties. For example, the airtime used by each transmission depends on the data rate used and the possible retransmissions that can happen in the MAC layer. The scheduling of airtime among the final users within a slice is also a challenge. It is expected that, while guaranteeing the airtime of the slice, each user receives a fair amount of this airtime. This is difficult considering each user can have different data rates and low data rates consume more airtime than high data rates for the same amount of data.

III. Our Resource Allocation Proposal

As already mentioned, our focus is on the edge of the network, more precisely on WiFi Access Points. The objective is to implement *Infrastructure Sharing Slicing* by allocating the airtime resource to the different slices. The model is simple, a slice requests a percentage of the total airtime in an AP, then, if enough resources are available, the airtime is allocated to the slice.

A. Proportional Time Deficit Round Robin

For the airtime allocation, we propose a queuing structure and a scheduling mechanism based on the solution presented in [6]. Differently from this previous work where the objective was to guarantee airtime fairness, our proposal allows slices to be defined with different airtime requests.

The mechanism is a modified version of *Deficit Round Robin* (DRR) [8] which we call *Proportional Time Deficit Round Robin* (PT-DRR). It presents some important differences with the original DRR that will be explained later.

The proposed mechanisms aims to (1) identifying the existing flows of traffic, (2) assigning a queue to each flow and (3) scheduling the service to each flow to meet the slice requests. The queue servicing consists on a round-robin scheduling with *quantums* on each queue. The *quantum* is a configurable parameter which controls how much airtime is allocated to each queue in a round. The current airtime assigned to a queue is maintained by a *deficit*. Each time a packet is dequeued, the deficit is decreased by the airtime consumed by the packet. Packets are dequeued only while a positive *deficit* remains. When the *deficit* is zero or less, no more packets can be dequeued and the algorithm switches to the next queue in a round robin manner. On every new round of the algorithm the *deficit* is increased by a *quantum*.

Counting the transmission time allows a direct control over the airtime used by each flow independently of the packet size or the rate. However, since packet aggregation and possible retries of packet transmission can happen after a packet is dequeued, the airtime consumed can only be calculated after the packet transmission succeed. Thus, the *deficit* of a queue is updated after the packet transmission, which leads to possible negative *deficits*.

Actually, two main differences between our proposed PT-DRR and the original DRR proposed in [8] exist and are the following:

- Instead of counting bytes, we count for the transmission time (airtime).
- We dequeue packets until the deficit reaches a value of zero or less.

B. PT-DRR Design

PT-DRR is envisioned to be implemented in the APs of a WiFi network and replaces the queuing and scheduling structure of the AP. For a descriptive explanation on how this can be implemented in hardware, the reader can see [6].

PT-DRR maintains a queue for the traffic flow of each user (WiFi station) within a slice. Hence, we have a queue per user and per slice. This means that if a user belongs to three slices, three queues will be created in the system for that user, each of them within a different slice (see Figure 1). Each queue defined in the system has its own quantum and keeps track of its deficit. Our current proposal applies only so far to the downlink traffic, but it is possible to extend it to consider also uplink traffic.

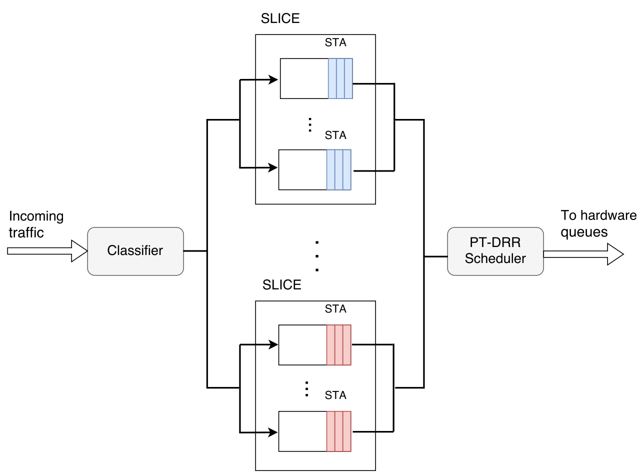

Fig. 1. PR-DRR simplified queueing architecture.

The design of PT-DRR follows two main objectives:

- Allocate to each slice the proportion of requested airtime.
- Allocate airtime to each user within a slice fairly.

To accomplish both objectives, the quantums of each queue must be carefully calculated. In the following we explain how we define the quantums.

For an AP where PT-DRR is running, lets define $\mathcal{S}$ as the set of all slices instantiated on the AP and $N_\mathcal{S}$ the number of slices in $\mathcal{S}$. Each slice $j \in \mathcal{S}$ requires a ratio of the airtime to be allocated denoted by $p_j \mid 0 < p_j \leq 1$. Lets also define $\mathcal{U}$ as the set of users associated with the AP, $\mathcal{U}_j$ the set of users that belong to slice j and N_j the number of users in $\mathcal{U}_j$. We denote by $q_{i,j}$ the quantum of the queue i in the slice j and define Q_j as the sum of all the quantums of slice j ($Q_j = \sum_{i=1}^{N_j} q_{i,j}$).

To achieve the first objective, the quantum Q_j assigned to a slice j must satisfy:

$$Q_j = p_j \sum_{l=1}^{N_\mathcal{S}} Q_l \quad \forall j \in \mathcal{S} \tag{1}$$

On the other hand, to achieve fairness among users of the same slice, all the quantums within the slice must be equal. Then, we have that the quantum of any queue i in a slice j that requests a proportion of the airtime p_j and has N_j users is:

$$q_{i,j} = q_j = \frac{Q_j}{N_j} \quad \forall j \in \mathcal{S} \tag{2}$$

Which can be rewritten as (we obviate the intermediate steps because of space constrains):

$$q_j = \frac{p_j \sum\limits_{l=1, l \neq j}^{N_\mathcal{S}} N_l q_l}{N_j(1-p_j)} \quad \forall j \in \mathcal{S} \tag{3}$$

From the above result it is important to note that each time a queue is created or deleted (i.e. when a user connects or disconnects) from the system, the quantums of all the queues are recalculated. Even more, equation 3 does not provide an absolute value for the quantums, but a relation between the quantums of the slices. To resolve this indetermination, we fix the minimum possible quantum allowable in the system and assign it to the minimum q_j.

A remarkable feature of PT-DRR resides in that it does not provide a static assignment of resource neither it constrains the use of resources, but properly schedule packets (if available) respecting the airtime assignment. For example, if a flow does not use all its assigned airtime, the unused time can be used by the other flows proportionally.

IV. Performance Evaluation

In order to asses the performance of PT-DRR, we implemented the algorithm in the NS3 Network Simulator [9] by modifying the existent transmission queuing structure of the wireless module. We conducted simulations in an attempt to evaluate the proposed resource allocation mechanism showing the correct airtime allocation.

All experiments consists on a deployment of one AP and 10 users randomly located around the AP. 3 slices are defined:

- Slice 1 requests the 20% of the airtime resource of the AP and there are 4 users associated with the slice (U1, U2, U3 and U4).
- Slice 2 requests the 20% of the airtime resource of the AP and there are 4 users associated with the slice (U4, U5, U6 and U7).
- Slice 3 requests the 60% of the airtime resource of the AP and there are 4 users associated with the slice (U7, U8, U9 and U10).

One should notice that U4 and U7 are belonging to two slices.

In this scenario, static users are deployed randomly around the AP with an average distance of 5m. The AP is configured to transmit Constant Bit Rate (CBR) UDP traffic to each user with a high rate so as to maintain all queues backlogged. Simulations were repeated 20 times varying the location of the users and each simulation runs during 60 seconds. We measured the airtime allocated to each slice in intervals of one second. We then calculated the ratio assigned to each slice as the proportion from the total airtime allocated to the three slices.

Figure 2 shows the airtime ratio assigned to each slice through one simulation execution. The median and the higher and lower airtimes assigned through the simulation time are depicted. As one can see, the requested proportion of airtime to each slice is in fact correctly allocated and with a very low variability. It is important to note that in the 20 instances executed the variability of the allocated airtime is analogous to the results shown.

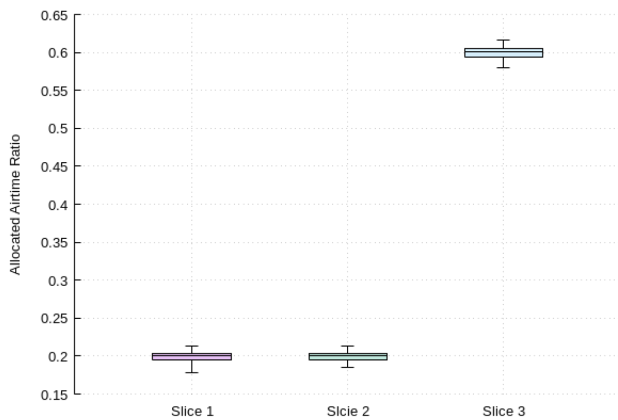

Fig. 2. Airtime assignment ratio to the slices.

We also evaluated the airtime assigned to each user for every slice so as to assess the fairness among users of the same slice. The results for one execution are shown in Figure 3. The figure highlights that the assignments for each user are noticeably fair. To better judge the fairness, we calculated the Jain's fairness index of the airtime assigned to each user within the slices for the 20 simulations executed. The results show that for all slices, the index is greater than 0.999 which is a very good result.

V. Conclusions

We have proposed a resource allocation mechanism based on airtime scheduling to achieve infrastructure sharing and slicing in WiFi Access Points. The approach is aimed to be simple and has the potential to be easily used in diverse infrastructure sharing scenarios.

Although airtime sharing has been previously studied in WiFi networks, the originality of our approach is in the consideration of the airtime as a resource for network slicing. Many existent works have proposed frameworks or high-level management solutions for slicing but there was a lack of ideas on how to specifically implement slicing at the WiFi APs. It is important to note that, to the best of our knowledge, we are the first to propose an airtime scheduling mechanisms for network slicing.

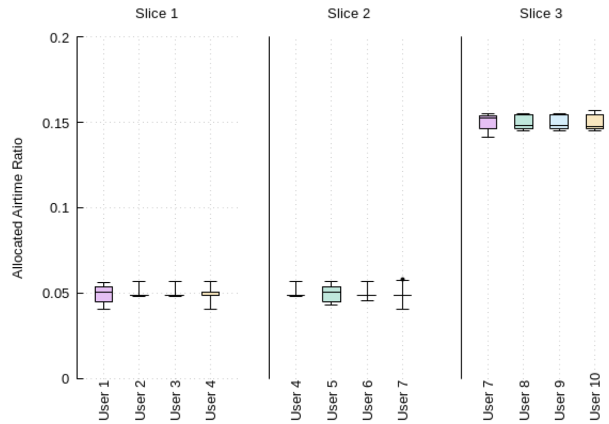

Fig. 3. Airtime assignment ratio to each user of the slices.

Furthermore, we implement our proposal on a simulator and demonstrate its key characteristics: airtime allocation based on slices requests, airtime fairness among users of a slice, efficient resource utilization and isolation between slices. We are aware that additional experimentation and study of the proposal is needed. Despite this we believe this initial results are promising.

We are currently working on a formal demonstration of the allocation guarantees provided by the algorithm as well as on more complex simulation scenarios. Moreover, in future research we plan to extend the mechanism to provide throughput and/or delay guarantees so as to also support *QoS Slicing*.

Acknowledgment

This work is being partially supported by the project ADVICE (TEC2015-71329-C2-2-R / MINECO-FEDER) and by the STIC AmSud project VNET.

References

[1] M. Richart, J. Baliosian, J. Serrat, and J.-L. Gorricho, "Resource slicing in virtual wireless networks: A survey," *IEEE Transactions on Network and Service Management*, vol. 13, no. 3, pp. 462–476, 2016.

[2] K. Nakauchi, Y. Shoji, and N. Nishinaga, "Airtime-based resource control in wireless lans for wireless network virtualization," in *Ubiquitous and Future Networks (ICUFN), 2012 Fourth International Conference on*. IEEE, 2012, pp. 166–169.

[3] G. Bhanage, D. Vete, I. Seskar, and D. Raychaudhuri, "Splitap: leveraging wireless network virtualization for flexible sharing of wlans," in *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*. IEEE, 2010, pp. 1–6.

[4] K. Katsalis, K. Choumas, T. Korakis, and L. Tassiulas, "Virtual 802.11 wireless networks with guaranteed throughout sharing," in *Computers and Communication (ISCC), 2015 IEEE Symposium on*. IEEE, 2015, pp. 845–850.

[5] M. Heusse, F. Rousseau, G. Berger-Sabbatel, and A. Duda, "Performance anomaly of 802.11 b," in *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, vol. 2. IEEE, 2003, pp. 836–843.

[6] T. Høiland-Jørgensen, M. Kazior, D. Täht, P. Hurtig, and A. Brunstrom, "Ending the anomaly: Achieving low latency and airtime fairness in wifi," in *2017 USENIX Annual Technical Conference (USENIX ATC 17)*. Santa Clara, CA: USENIX Association, 2017, pp. 139–151.

[7] Y. Zaki, L. Zhao, C. Goerg, and A. Timm-Giel, "Lte wireless virtualization and spectrum management," in *Wireless and Mobile Networking Conference (WMNC), 2010 Third Joint IFIP*. IEEE, 2010, pp. 1–6.

[8] M. Shreedhar and G. Varghese, "Efficient fair queueing using deficit round robin," in *ACM SIGCOMM Computer Communication Review*, vol. 25, no. 4. ACM, 1995, pp. 231–242.

[9] "NS-3 Network Simulator." [Online]. Available: http://www.nsnam.org/