Mixing MACs: An Introduction to Hybrid Radio Wireless Virtualization

Gautam Bhanage†
†Tech Report: Bhanage.com / GDB2017-005
gbhanage@gmail.com

Abstract—This study presents the design of the hybrid wireless virtualization (HWV) controller based network architecture. Using a HWV controller, an unified approach can be taken for provisioning and management of virtualized heterogeneous radios, irrespective of their MAC and PHY layer mechanisms. It is shown that the airtime occupancy by transmissions from different slices or groups can be used as a single metric for tying these virtualized platforms. The HWV controller can account and dynamically re-provision slice quotas, which can be used for maximizing the network operator’s revenue or aggregate system throughput performance. Results from simulations show that an HWV controller based infrastructure is able to improve the revenue generated from a single virtualized basestation and an AP by up to 40% under tested conditions.

Index Terms—Virtualization, network virtualization, wireless, wireless virtualization.

I. INTRODUCTION

Wireless virtualization can be defined as the approach by which the individual underlying physical network interfaces are abstracted by more than one virtual interfaces, which allow for better sharing of the physical interface and the medium used by the radio associated with that interface. The term virtualization itself was coined from the server systems area of virtualization and usually involves the application of three fundamental concepts for sharing the resource: (1) abstraction, (2) programmability, and (3) isolation. Abstraction ensures that the same interface is available to all the entities using the systems. Programmability ensures that each of the virtual entities (interfaces in our case) are able to affect properties of the underlying physical interface without conflicting with the requirements of other virtual entities. Finally, isolation is a property which ensures that the load on one virtual entity does not affect the other, and essentially, each of the interfaces are able to work oblivious of each other. In our wireless case, virtualization refers to running multiple virtual networks which are leased to mobile virtual network operators (MVNOs) by the mobile network operator (MNO) which owns the underlying physical networks.

Recent virtualization wireless efforts have shown how individual wireless network interfaces like those in a 4G cellular basestation [1], [2] and a cellular WiFi hotspot [3] can be virtualized. However, in this study we wish to look at a scenario beyond individual virtualized wireless components, where a central framework could be employed by a network operator for maximizing its revenue from the MVNOs. Consider the virtualized network architecture shown in Figure 1.

As seen in the figure, a MNO will have access to different types of virtualized radio platforms, and a mechanism is needed for establishing control and provisioning across these different radios. Our HWV controller design will be able to leverage such configurable virtualized radio platforms for revenue maximization of the MNO.

The contributions made by this study can be summarized as follows:

- Metric: We discuss and show how a single metric could be used for addressing the issue of resource accounting and allocation across a diverse set of radio interfaces.
- Framework: We present a mathematical optimization framework based on the above metric that will allow for enabling revenue ($/sec) or capacity maximization (bits/sec) of the physical network by dynamically re-provisioning individual virtualized wireless networks or interfaces.
- Prototype design: Guidelines are laid out for our work-in-progress HWV controller based network architecture design.
- Simulations: Finally, using simulations, we show some preliminary results that can be obtained by deploying

Fig. 1. An example of a virtualized network architecture. The figure shows two slices belonging to MVNO1 and MVNO2 which are hosted by the network operator. These slices are hosted across different virtualized wireless substrates.

1The ideas presented here are based on our issued patent [4] at one of my previous employers. This work has no affiliation or relation to my current or other past employers. Opinions and thoughts discussed here are my own and no one else. Please use this document to stoke your thinking and none of these should be construed as advice.
our HWV controller with virtualized wireless network systems.

Rest of this document is structured as follows. Section II discusses related work. Section III presents our hybrid wireless virtualization based network architecture and platform. Section IV presents results from our simulation of the controller with a virtualized AP and basestation. Finally, Section V presents conclusions and future directions.

II. RELATED WORK

Previous network virtualization [5], [6], [7] efforts can be broadly classified as those for wired networks and for wireless networks. In terms of wired networks, virtualization is used for running testbeds [8], [9], and also for dynamically re-configuring and maintaining routers [10]. A host of other studies have discussed how topologies can be managed across virtualized wired networks [11], [12], and some also for wireless networks [13].

Wireless device virtualization was first proposed for short range WiFi radios for virtual access points (VAPs) [14]. These VAPs would run as abstractions on a physical AP. One of the first virtual basestation designs is the VANU MultiRAN virtual basestation design [15]. It runs the entire virtualized basestation transceiver (BTS) stack from software. A similar approach is considered for the Open Basestation project [16] which relies on implementing a 2G BTS in software, though it does not support virtualization. Another study [17] discusses approaches in which the network architecture may be emulated using virtual machines, but does not deal with the emulation of the radio itself. An approach that modified a proprietary MAC scheduler for virtualizing a BTS is discussed in another previous work [18]. Another of our previous works has focused on mapping virtual wireless networks to physical wireless networks [11], [12], and some also for wireless networks [13].

Wireless device virtualization was first proposed for short range WiFi radios for virtual access points (VAPs) [14]. These VAPs would run as abstractions on a physical AP. One of the first virtual basestation designs is the VANU MultiRAN virtual basestation design [15]. It runs the entire virtualized basestation transceiver (BTS) stack from software. A similar approach is considered for the Open Basestation project [16] which relies on implementing a 2G BTS in software, though it does not support virtualization. Another study [17] discusses approaches in which the network architecture may be emulated using virtual machines, but does not deal with the emulation of the radio itself. An approach that modified a proprietary MAC scheduler for virtualizing a BTS is discussed in another previous work [18]. Another of our previous works has focused on mapping virtual wireless networks to physical wireless networks [11], [12], and some also for wireless networks [13].

For the purpose of link and user performance measurement, there are several widely used metrics like the throughput, goodput or even the delay performance experienced by the user. Due to their popularity and direct impact on the end user experience, these metrics are also used in resource provisioning on scheduled MAC mechanisms like 4G basestations. For example, in a WiMAX basestation, the system administrator can specify the number and types of flows in a service class, with each class having a minimum, maximum throughput and delay constraints associated with it. These help the basestation schedule links to better suit the application requirements. However, these metrics cannot be used for our HWV architecture. The main reasons for not being used in our framework are that these metrics do not translate into the use of the same amount of radio resources on any type of radio. For example, the same amount of throughput can be achieved by using different physical layer rates on two different links. However, the link with the slower physical layer rate uses more radio resources to achieve the same throughput as that by the faster radio link. In order to remove these, and other differences in resource usage induced by the MAC layer, such as channel access time, MAC and PHY overheads, MAC enhancements (like custom or standard aggregation mechanisms [20]) and the nature of the MAC itself (scheduled or unscheduled), we propose using radio airtime as a single metric for control of the virtualized network. Note that earlier studies [18] have maintained that rates for a single radio can be used as alternatives. However, as will be seen through further discussion, fairness across radio technologies can be truly represented through airtime fairness. Universality of $t_j$: The fraction of airtime $t_j$ used by any radio for wireless transmission directly translates into the resources on every wireless device. In a previous work, it has been shown that the reservation rate in a WiMAX scheduler can be used for specification of required time slots. We will further show that airtime can be used as a single metric for resource accounting on a scheduled as well as unscheduled MAC. Consider that $\mu_j$
represents the number of resource blocks in a 4G base stations MAC scheduler that are allocated to slice $j$. In this case, it can always be experimentally verified that:

$$t_j = \frac{\mu_j}{\sum_j \mu_j} = \frac{\text{rate}_{t_j}}{\text{rate}_{t_j}^{\text{phy}}},$$

(1)

This is the case because airtime utilization of a slice is a direct result of such MAC scheduling. In any radio, the airtime fraction is also equivalent to the aggregate throughput $\text{rate}_{t_j}$ achieved by slice $j$, as a fraction of the assigned average physical $\text{rate}_{t_j}^{\text{phy}}$ for the slice. Hence, irrespective of having a scheduled MAC (in a base station) or an unscheduled CSMA MAC (in an AP), airtime can be used as a single number for accounting and controlling slice radio resource usage.

With the advent of faster WiFi mechanisms like 802.11ac and 802.11ax, this metric should provide a way to account for MU—MIMO and OFDMA mechanisms. If the underlying OFDMA radio is divided on subcarriers, then this can still be translated into airtime by normalizing the airtime utilized by the number of subcarriers allotted over a period of time. Similarly, for MU—MIMO radios, the airtime can be normalized by the number of simultaneous transmissions to ensure that we account for fairness between MU and non—MU transmissions.

Based on this insight, we will now present the design of the HWV controller that will take into account resource usage across different virtualized network components, and will be responsible for dynamically reprovisioning these components for network operator revenue maximization or plain rate maximization.

C. HWV Controller Model

**Formulation:** The HWV controller model is responsible for getting airtime usage, and rate feedback from each of the virtualized components such as basestations and AP. It is also responsible for reprovisioning them so that the network operator’s profit is maximized. The virtual basestation [1] architecture and the modified SplitAP [3] framework enforce group airtime fairness across slices.

Let the airtime quota requested per slice across a group of basestations, or access points be specified by the set $Q = \{Q_1, \ldots, Q_n\}$. Let the airtime allocated per slice $j$ at every base station $k$ be given as $t_{j,k}$. Similar quotas $Q = \{Q_1, \ldots, Q_n\}$ are specified for the same set of slices over a set of access points. Let the airtime allocated per slice $j$ at every access point $k$ be given as $r_{j,k}$. Similarly, let the requested airtime per slice per base station or access point be given as $\hat{r}_{j,k}$ and $\hat{r}_{j}$ respectively. The requested airtime is updated at runtime for every iteration at which the optimization problem is solved. Corresponding usage flags for basestation and access points are given as: $u_{j,k}$ and $u_{j}$.

We define the overall revenue function across the set of basestations and access points owned by the operator as:

$$\text{Rev}(t_{j,k}) = t_{j,k} \times \Gamma_j(\hat{r}_{j,k})$$

(2)

$$\text{Rev}(\hat{t}_{j,k}) = \hat{t}_{j,k} \times \hat{\Gamma}_j(\hat{r}_{j,k})$$

(3)

Here, $\Gamma_j$ and $\hat{\Gamma}_j$ are the utility functions provided by MVNO $j$ for matching their traffic demands on a base station and access point respectively. For now, we can define the utility functions as a linear function of the allocated airtimes given as:

$$\Gamma_j(t_{j,k}) = C_j$$

(4)

$$\hat{\Gamma}_j(\hat{t}_{j,k}) = \hat{C}_j$$

(5)

These equations indicate a purely increasing utility with increasingly allocated capacity. Eventually, the values $C_j$ and $\hat{C}_j$ can be equated to the average physical rates available for the clients belonging to the slices $j$ on basestations and access points. Information of the average physical layer rates, and the overall traffic flowing to the clients is available at the controllers of both the virtual base station framework and the SplitAP design and these can be polled regularly by the HWV controller. In our initial evaluation of the setup, we will set $C_j$ and $\hat{C}_j$ equal to average of the slice physical layer rates to the clients. This eliminates the pricing component from the objective function, and the problem becomes a rate maximization problem. Eventually, we demonstrate the revenue maximization function of the controller by substituting linear objective functions of achieved rate for $C_j$ and $\hat{C}_j$. The problem being solved at the controller can finally be formulated as:

$$\text{maximize} \ \sum_j \sum_k \text{Rev}(t_{j,k}) + \text{Rev}(\hat{t}_{j,k})$$

subject to

$$\sum_j t_{j,k} \leq 1, \ j = 1, \ldots, m.$$  

$$\sum_k t_{j,k} \leq Q_j, \ k = 1, \ldots, n.$$  

$$\forall j, k \ \ t_{j,k} \geq \hat{t}_{j,k},$$

$$\sum_j t_{j,k} \leq 1, \ j = 1, \ldots, m.$$  

$$\sum_k t_{j,k} \leq \hat{Q}_j, \ k = 1, \ldots, n.$$  

$$\forall j, k \ \ \hat{t}_{j,k} \geq \hat{\delta}_{j,k},$$

In the above optimization formulation, the $\hat{\delta}_{j,k}$ is used to represent the minimum airtime reservation at the base station $k$ for the slice $j$. Since the objective function, and all of the constraints are convex, the formulation can be solved at the HWV controller using any standard convex optimization tool or heuristic.

D. Prototype Design

In this section we will discuss the work in progress for building the HWV controller and network architecture prototype. For this proof of concept architecture, we leverage the previously designed virtual wireless basestation (vBTS) prototype [1], and a modified SplitAP [3] based virtualized AP prototype that allows us to control downlink group airtime quotas. Both of these prototypes allow the network operator
Fig. 2. An example deployment of the HWV controller in a virtualized network architecture. Since the HWV controller is reachable by IP it can be placed wherever there is IP connectivity. However, for achieving fine grained control, it is better placed closer to the edge or near the cells that it controls.

dynamically controller the slice quotas. Both the vBTS framework and the SplitAP framework are connected to the HWV controller running on the network through an IP backhaul.

An example network layout is as shown in the Figure 2. As seen we propose deploying independent HWV controllers in access networks, and each of these HWV controllers are responsible for controlling a limited set of vBTS frameworks and SplitAPs. For every operator, these independent HWV controllers in turn will be connected to a HTTP based network wide policy database (NWPD), that advertises the operators revenue generation capabilities through different access mechanisms like cellular links or WiFi hotspots. The NWPD will also be responsible for advertising utility functions for each of the slices. We have made a provision for advertising the utility functions because we envision that these will change based on time varying agreements between the network operator (MNO) and the virtual network operators (MVNOs). Each of the HWV controllers will be configured at deployment time with the URL of the NWPD, and will be responsible for independently fetching operator policies and utility functions. Note that the fetch from the site-local HWV controller can happen very infrequently, and is dependent on the time of the lease agreements between the MNO and the MVNO. On the other hand, the control loop between each of the HWV controllers and their connected substrates will be on a much finer scale.

IV. SIMULATIONS FOR HWV EVALUATION

We present some initial results from our virtualization setup.

A. Setup

The setup consists of our centralized HWV controller that is connected to a virtualized access point [3], and virtualized WiMAX basestation [1]. The parameters for the simulation are as described in the Table I. In our preliminary evaluation we determine the performance of the system with the loads on each of the slices varying randomly. The change in load is based purely on the requirements for different airtimes to support the same amount of traffic, which is either due to the change in the link conditions to the access point or the basestation.

B. Unconstrained Setup with Varying Slice Rates

In our first experiment, we consider that the network administrator has placed no constraints on the quota allocations at either substrate (AP or the BTS). In this case we plot the weights or airtime allocation quotas computed by our architecture and the corresponding revenue generated. Note that in this experiment, the HWV controller is configured to purely solve a throughput maximization problem.

Figure 3 shows the airtime allocations calculated by the HWV controller based on varying load conditions.

| Parameter                      | Value |
|--------------------------------|-------|
| Simulation runs                | 1000  |
| APs                            | 1     |
| BTSs                           | 1     |
| AP peak throughput             | 36Mbps|
| BTS peak throughput            | 20Mbps|
| SLC1 airtime bid               | 0.6   |
| SLC2 airtime bid               | 1.0   |
| Slices per AP                  | 2     |
| Slices per BTS                 | 2     |

Fig. 3. Weights (airtime allocations for slices) calculated by the HWV controller based on varying load conditions.

1Numbers for throughput values of the BTS and AP are based on measurements on commercial hardware.
by the net airtime constraints for each of the slices, and the limitations of total airtime on each of the radios.

The CDF of the normalized rate generated from the corresponding experiment is as plotted in Figure 4. As seen in the results, we observe that on an average the revenue performance improves by at least 9.3%. In the best case, we observe that the revenue performance improve by up to 39.7%.

C. Constrained Setup with Varying Slice Rates

In the previous case, we measured performance of the HWV framework in the absence of any restrictions on the weights allocated to the individual substrates or the slices. Hence, in every case the controller, calculated a solution that maximizes aggregate rate across substrates without any external constraints and the results are influenced only by the utility functions for each of the substrates. However, our HWV controller allows us to impose constraints on the quotas allocated at each of the substrates, the AP and the basestation, for each of the slices. In this experiment, we impose a constraint of receiving a minimum allocation of 0.7 for the first slice on the AP.

Airtime allocations from the experiment are as shown in the Figure 5. We observe that the allocations change based on

the load conditions. However, we observe that the oscillations are not as much as those seen in Figure 3 because of the constraints imposed on the desired airtime quotas. The CDF of the rate allocations generated from the experiment are as shown in the Figure 6. This plot shows the CDF of the revenue generated with our HWV controller as opposed to using a static allocation. Results show that even in this case, despite of the airtime quota constraints the average revenue improves by 10%, and the best case revenue improves by up to 40%.

D. Revenue Performance

In all of the previous cases, we have shown how our architecture can solve an aggregate rate maximization problem for the network operator when pricing is not involved. In this study we show how the operator could include pricing information in the HWV architecture and use it to solve a revenue maximization problem. In this experiment, we assume that the revenue generated out of the 4G WiMAX BTS is twice as that obtained from the WiFi AP. Accordingly, this information is used to condition the objective function in the controller. Rest of the experiment is as before, and results are measured for iterations with randomly generated average slice rates.

Results from this experiment are as shown in the Figure 7. The results show a CDF of the revenue generated through a
static allocation strategy and that obtained through a dynamic re-allocation approach using our HWV controller. The results show that adding our HWV controllers re-allocation strategy for changing average rate conditions, for differently priced substrates allows the operator to do significantly better than a static slice allocation. In this case, we observe that the average improvement in revenue is 10%, with a best case improvement of up to 40%.

V. CONCLUSIONS AND FUTURE DIRECTIONS

This study presents the design and the initial evaluation of the hybrid wireless virtualization (HWV) controller mechanism. We show that the HWV controller operates across heterogeneous virtualized radio technologies and maximizes network operator revenue based on dynamic airtime quota re-allocations. We show that the objective for revenue maximization from multiple virtualized radio interfaces can be formulated as a convex optimization problem, under the conditions that the utility functions from each of the virtualized radios are convex. Results from preliminary evaluations show that in certain cases our setup can be improve revenue by up to 39% over static allocation schemes. The average performance improvement in an unconstrained case, with no minimum quota limitations for the slices is seen to be approximately 40%. Though the absolute numbers in specific cases will possibly vary based on load conditions, physical layer rates, and actual radio efficiencies, we see that our HWV controller is able to successfully improve the aggregate network rate or the revenue for the network operator by dynamically changing airtime allocations for slices at run time.

REFERENCES

[1] G. Bhanage, I. Seskar, R. Mahindra, and D. Raychaudhuri, “Virtual basestation architecture for an open shared wimax framework,” in ACM SIGCOMM Workshop on Virtualized Infrastructure Systems and Architectures, New Delhi, India, 8 2010.
[2] G. Bhanage, I. Seskar, and D. Raychaudhuri, “A virtualization architecture for mobile wimax networks,” ACM SIGMOBILE Mobile Computing and Communications Review, vol. 15, no. 4, pp. 26–37, 2012.
[3] G. Bhanage, D. Vete, I. Seskar, and D. Raychaudhuri, “SplitAP: leveraging wireless network virtualization for flexible sharing of WLANs;” in IEEE Globecom 2010 - Next Generation Networking Symposium (GC10 - NGN), Miami, Florida, USA, 12 2010.
[4] G. Bhanage and S. Ganu, “Hybrid wireless virtualization architecture,” Apr. 25 2013, uS Patent App. 13/489,253. [Online]. Available: http://www.google.com/patents/US20130100807
[5] H. Wen, P. K. Tiwary, and T. Le-Ngoc, “Wireless virtualization,” in Wireless Virtualization. Springer, 2013, pp. 41–81.
[6] C. Liang and F. R. Yu, “Wireless network virtualization: A survey, some research issues and challenges,” IEEE Communications Surveys & Tutorials, vol. 17, no. 1, pp. 358–380, 2015.
[7] G. Bhanage, “Network virtualization on the wireless edge,” Ph.D. dissertation, Rutgers University-Graduate School-New Brunswick, 2011.
[8] L. Peterson, S. Muir, T. Roscoe, and A. Klingaman, “PlanetLab Architecture: An Overview, Tech. Rep. PDN-06-031, May 2006.
[9] “VINI,” http://www.vini-veritas.net/
[10] Y. Wang, E. Keller, B. Biskeborn, J. van der Merwe, and J. Rexford, “Virtual routers on the move: live router migration as a network-management primitive,” SIGCOMM Comput. Commun. Rev., vol. 38, pp. 231–242, August 2008. [Online]. Available: http://doi.acm.org/10.1145/1402946.1402985
[11] M. Yu, Y. Yi, J. Rexford, and M. Chiang, “Rethinking virtual network embedding: substrate support for path splitting and migration,” SIGCOMM Comput. Commun. Rev., vol. 38, no. 2, pp. 17–29, 2008.