Future Network and MobileSummit 2013 Conference Proceedings
Paul Cunningham and Miriam Cunningham (Eds)
IIMC International Information Management Corporation, 2013
ISBN: 978-1-905824-37-3

Making IEEE 802.11 Wireless Access Programmable

Evgeny Khorov[1], Artem Krasilov[1], Alexander Safonov[1],
Pablo Serrano[2], Ilenia Tinnirello[3]
[1]*Institute for Information Transmission Problems (IITP RAS), Bolshoy Karetny per. 19, 127994 Moscow, Russia, Email:{khorov,krasilov,safa}@iitp.ru*
[2]*University Carlos III of Madrid, Avda. Universidad 30, 28911 Madrid, Spain*
Email: pablo@it.uc3m.es
[3]*Universit'a di Palermo, 90128 Palermo, Italy*
Email:ilenia.tinnirello@tti.unipa.it

Abstract:
In this paper, we present a modular Layer 2 architecture which makes wireless access in IEEE 802.11 networks programmable and thus opens the door for broader range of enhancements. We show the power of the proposed architecture by presenting a number of innovative solutions for infrastructure, direct links and mesh cases. Early prototyping results are publicly available and can be used to develop solutions not so strictly bounded by legacy access rules, to quicker and more accurately meet evolving user demands.

Keywords: IEEE 802.11, flexible architecture, programmable MAC

1. Introduction

Communication networks connect the modern world, permeating the fabric of industrial, academic, and societal life. Ever increasing mobile traffic is one of the strongest drivers for evolution of all wireless networks. All forecasts agree that the mobile traffic pie is getting so big that even at the level of standardization bodies of cellular and local area networking we finally see the steps for cooperation replacing traditional competition, e.g. to facilitate mobile traffic offload into local area networks.

Another important driver is the requirements of wireless networks users becoming increasingly stringent. While IEEE 802.11 technology struggles for efficient use of wider spectrum, the user's expectations has evolved and the mission of networks has shifted from moving data towards providing services. Many applications today require high throughput indeed, but various services also require highly intelligent network protocol suite tailored for the specifics of those services. To say nothing about the revolution of services coming with Machine to Machine (M2M) communication with ultra-short packets, energy-constrains, sometimes critical delay requirements and unprecedented predicted population [1].

IEEE 802.11 technology has shown impressive progress in addressing these challenges. However, being designed to support a pre-established wireless access operation, it fails to adapt to the demands evolution in time. We believe that the shift from the antiquated networks moving data towards future networks providing services is restrained by rigid wireless interfaces suffering from the lack of flexibility in existing cards/devices. Even little modification of the interface is often only possible with hardware re-design associated with enormous cost and time. Indeed, out of all the work carried out to address the performance issues of existing standards in new application and networking scenarios, surprisingly, only a negligible part has migrated to real world deployments.

One the other hand, an extremely low level programming interface poses the risk of "fragmentation" in the space of approaches: the emergence of too many alternatives risks to prevent that convergence and critical mass is reached towards a small set of "best" ones. Indeed, the thorough understanding of what, specifically, should be programmable, and why (i.e. understanding what features future generation wireless interfaces should expose and what should instead remain confined in the manufacturing domain) is a difficult issue which we have addressed in the FLAVIA project [2].

Further, in Section 2, we overview a novel modular Layer 2 architecture allowing to take advantage of the innovative results of scientific communities and get ready for emerging services. In Section 3, we present selected original methods enabling intelligent medium access. Though impossible for implementation in the existing standard, those methods may be easily impletemneted with proposed architecture. We finally show some early prototyping and trial results in Section 4. Section 5 concludes the paper.

2. A Flexible Layer 2 Architecture

IEEE 802.11 standardization task groups have considered some of the extensions proposed by the research community by ratifying new features or new control frames with extensible information elements, but most brilliant solutions are still not supported. We propose a different solution based on the definition of a new architecture for wireless interfaces, in which reconfigurations are accomplished by means of a powerful programming interface which extends somehow the concept of *tuning knobs*. Rather than working on a pre-defined set of configurable parameters, such as the contention windows, the channel holding times, the handshake mode, and so on, we enable the possibility to work on a set of fixed primitives for managing hardware actions and events. The primitives can be composed into three layers of functionalities with different complexity (state machines, functions and services).

The resulting architecture is shown in Fig.1. At the lowest functional layer, there is a special architecture component called Wireless MAC Processor (WMP) which is responsible of interacting and scheduling actions on the hardware platform, by executing an abstract program specified in terms of a state machine working on the available primitives. On top of this layer, there is a layer called function container which runs data organization and elaboration functions in terms of frame forging, queue management, definition of signal messages, performance statistics and so on. This layer interacts with the lower functional layer by passing frames to be transmitted by the platform and/or by invoking the system primitives. At the highest layer, there is the service container in which different Layer 2 sub-tasks are executed in terms of independent (replaceable) modules. Each service may expose a customized interface to the higher layers and can invoke functionalities available at the lower layers.

A transversal control plane works on the control of the functional layers (i.e. on the service/function logic and data structures), dealing with: i) service configuration, that is adding/removing service and function modules; loading/unloading running instances of services and functions; modifying/updating/reconfiguring service and function parameters (on the fly or offline); ii) consistency management, that is exporting services definitions to other nodes; verifying service conflicts; guaranteeing inter-nodes

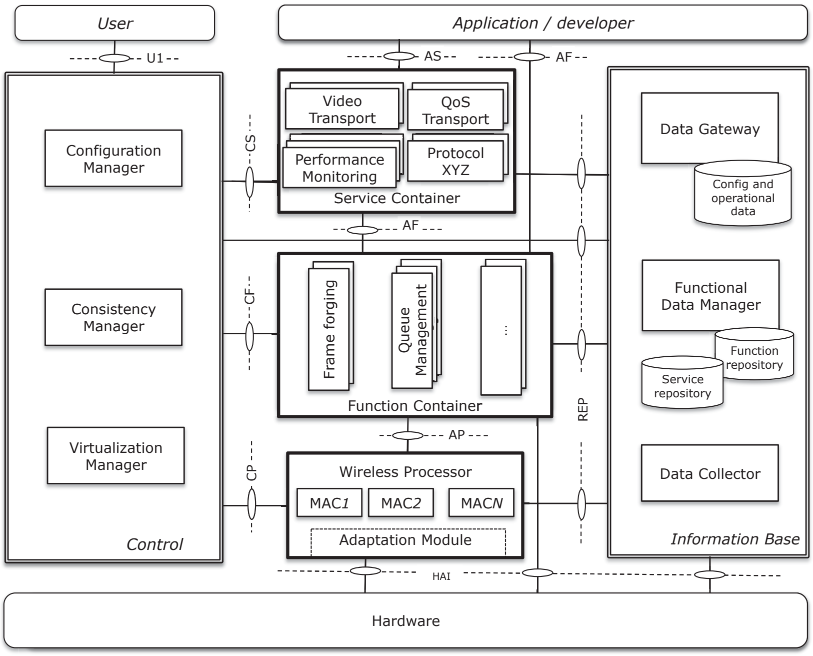

Figure 1: A high-level perspective of the proposed flexible wireless LAN architecture

and intra-node consistency; iii) resource virtualization, that is managing virtual interfaces; enabling virtualized services and functions; iv) data handling, that is collecting and storing system performance statistics; defining system state data; enabling function and service intercommunication. Details of the architecture can be found in the project report [3].

3. IEEE 802.11 technology enhancements

To show the power of the proposed architecture and its components, we have developed a number of solutions focusing on (i) improving the IEEE 802.11 data transport service in terms of throughput, energy consumption and reliability, and (ii) other important services such as virtualization, localization and parameterized QoS, which are not immanent to IEEE 802.11 technology, but emerged last years. In this section, due to space limitation we only present some selected ideas which details an interested reader can find in the references, covering the cases of infrastructure networking, direct link communication and mesh networking.

3.1 Enhancements for single-hop networks

3.1.1 Tuning the contention window to boost performance

In the IEEE 802.11 standard, one of the key parameters which determines the performance of the basic channel access method and thus network throughput is the minimal Contention Window (CW_{min}). Our study has shown that dynamic adaptation of CW_{min} along network conditions may result in relevant performance improvement. Specifically, by using game-theoretical [4] and control-theoretical [5] approaches we have developed algorithms for tuning CW_{min} to take into account current channel conditions and boost the network performance. Exploiting the ability to access CW_{min} through interface between Function Container and WMP, we have easily implemented those algorithms and experimentally validated their efficiency.

3.1.2 ACK piggybacking

According to the IEEE 802.11 standard [6], the basic data exchange between two stations comprises the transmission of a data frame itself and a control frame called acknowledgement (ACK). Though the data frame may be transmitted at any rate depending on the channel conditions, the ACK frame is always transmitted at the most robust rate, so the time of the ACK frame transmission is always not so negligible, and it is of the same order of magnitude as the time needed for transmission of short data frames, e.g. in the case of VoIP.

To reduce the overhead associated with ACK transmission we propose a simple modification of MAC operation which we call ACK piggybacking. Specifically, we propose to enable the data frame receiver to piggyback the ACK frame with the next data frame sent back to the transmitter (if any) instead of transmitting a separate ACK frame as in the legacy method. Though at first sight this modification looks pretty simple, it cannot be implemented in the current standard as it touches the core IEEE 802.11 functionality implemented at the firmware level. In contrast, with the proposed architecture the ACK piggybacking is easily implemented thanks to WMP which allows to alter low level MAC functionality.

Experimental results presented in [7] show that the network capacity measured in the number of voice calls doubles when the ACK piggybacking mechanism is in force. Another application of ACK piggybacking is for bidirectional TCP traffic, when short TCP acknowledgements are piggybacked to MAC acknowledgements. For this case the experimental results [8] show 20-40% improvement in data throughput.

3.1.3 Localization service

Currently the huge success of location-aware applications is pushing hand-set manufacturers and mobile operators into developing new mechanisms to extend or even outdo the Global Position System (GPS), especially for indoor environments, where the legacy GPS signal is poor. One of the approaches to localize Wi-Fi devices proposed recently in literature is based on the measurements of propagation delay between target station and several anchor Access Points (APs). Unfortunately, current off-the-shelf 802.11 cards expose to the driver a 1us clock resolution for measuring propagation delays which corresponds to a distance quantization error of 300m, obviously not suitable for deploying indoor localization. In contrast, the capability of WMP to access low level MAC events (i.e. start and end of frame transmission) allows to provide more accurate measurements. This advantage allows us to develop a new localization service which does not require any preliminary calibration or additional traffic overhead. Experimental results [9] show that in outdoor scenario absolute positioning errors is lower than 2.5m, while in indoor scenario lower than 5m in 80% of the tested positions.

3.2 Enhancements for direct link communications

3.2.1 DLS++: Multi-channel Direct Links

Direct Link Setup (DLS) is an 802.11 enhancement introduced for allowing station-to-station transmissions in infrastructure networks, to avoid the AP serving as relay in scenarios in which source and destination stations are in radio visibility. Thanks to our architecture, we extended DLS and made its management more flexible, by enabling each station to simultaneously work on two different channels. The primary channel is that of the AP network; the station has to periodically access it with legacy access rules for receiving beacons and retaining association. The secondary channel is ad-hoc set up and independently managed by the peer stations with customized access rules. For example, in case of low contention on the ad-hoc channel, the stations can negotiate more aggressive access behavior (with low contention window values or without contention).

In [10] we prove how different protocol variants can be immediately implemented and incrementally modified among the stations, with different synchronization and multi-threading solutions between the access to the two channels. Experimental results show that the customized direct-link access may bring dramatic throughput improvements in 802.11g networks, especially when it is managed on a secondary channel without contention (from about 12 Mbps of the normal DLS case to about 38 Mbps under the multi-channel DLS without backoff).

3.2.2 Self-Optimizing WLANs based on the Opportunistic use of Wi-Fi Direct

In current WLANs the well-known rate anomaly problem can drastically reduce the overall network performance. The use of opportunistic relaying can help to improve performance (e.g. [11]), but previous solutions suffer from at least one of the following issues: (i) power consumption increases, as the energy profile of devices is not taken into account; (ii) furthermore, stations not only diverge in power consumption but also in throughout requirement profiles, and they might be more or less willing to perform relaying depending on e.g. their battery status; (iii) finally, solutions typically involve tailored modifications and do not consider backwards compatibility, i.e., considering legacy stations in the maximization problem.

In contrast to previous approaches, we propose an optimization framework based on two key variables: the topology T and the schedule S. The former specifies the path every station uses to reach the Access Point (AP), and the later the amount of time each potential relay spends transmitting towards the AP and receiving data from other stations. Based on these, for every topology T the optimal schedule S can be computing by solving the following convex optimization formulation: $\max \sum_{n=1}^{N} U_n(X_n) - \sum_{n=1}^{N} L_n(Y_n)$, in which U_n is a concave throughput that maps the throughput X_n perceived by a user to a utility, and L_n is a convex function that maps the energy consumed Y_n to an incurred cost. The optimal topology can then be computed with an exhaustive search (in case of small deployments) or through the use of heuristics. Our results [12] show that proposed solution greatly improves network throughput (more than doubling it) and power consumption (up to 75% reduction) even in systems comprised mostly of vanilla stations and unmodified AP.

3.3 Enhancements for multi-hop networks

The vast commercial success of IEEE 802.11 technology has led to its extension to multihop case, and recently ratified 802.11s amendment defining mesh networking is already incorporated in the base standard [6]. Followed by 802.11aa amendment aiming to provide robust multimedia streaming in both infrastructure and mesh networks, 802.11s technology turned out to have limits of adaptation to the needs of such kind of traffic. We believe that the flexibility of our architecture allows to go far beyond those limits and enable parameterized QoS service by addressing the following issues.

3.3.1 Coordinated Channel Access

It is very well known that pure random channel access is poor in multihop environment due to interference from hidden stations, so, being limited by 802.11 core access rules, 802.11s amendment defines an additional reservation-based access method called MCCA. Those rules imply per-category frames queueing, so that frames of all streams which are to be transmitted with EDCA or MCCA are chosen by FIFO policy. To meet QoS requirement of particular stream, we propose in [13] *more intelligent queue management* and *modified access rules* for MCCA, so that each stream is transmitted independently. In addition, in [14] we show that MCCA reservations are not fully protected from so-called ACK-induced and two-hop interference which significantly diminish the value of MCCA reservations, and propose fixes for MCCA advertisement procedure and channel access rules, which are all easy to implement with the architecture described in Section 2.

3.3.2 Multiple Metrics Usage

The IEEE 802.11s amendment defines a frame forwarding framework allowing to replace default link metric and path selection protocol. What is not allowed and what is enabled by the proposed modular architecture, is the usage of *multiple metrics* as in [15] which is beneficial for QoS provisioning and would be very natural given multiple access methods, namely EDCA and MCCA which require totally different factors to be accounted by the metric. We propose an MCCA-oriented metric and show how multiple metrics improve the throughput and provide parametrized QoS in [16]. In addition, multiple metrics are required to deliver multimedia groupcast traffic as in [17].

3.3.3 Frame Forwarding Framework

Another core restriction of the frame forwarding framework defined in 802.11s amendment, which is overcome with the proposed modular architecture, is per-destination path selection. To meet QoS requirements of particular streams and to increase the total network throughput by load balancing, we propose in [18] a flexible protocol enabling *per-stream path selection*. The protocol also allows to divide end-to-end QoS requirements of the stream between the hops of the path and establish MCCA reservations at each hop, see [13], e.g. based on the method developed in [19].

3.4 Virtualization support

As mobile devices are becoming widespread and users increasingly prefer connecting to the Internet though wireless APs, Internet service providers are competing to provide wireless broadband services in popular locations such as airports, cafs, hotels, etc. As the infrastructure on such premises is usually managed by local businesses, network operators seeking to enable roaming services for their existing customers or to gain additional revenue from temporary users are often required to share the resources of a single AP with other parties (i.e. require virtualization support). A common practice for supporting virtualization is creating multiple logical networks (called Virtual Access Points, VAPs) over the same radio on a single radio channel. Logical virtualization is achieved thanks to the broadcast of multiple beacons, announcing each of them a logical network with particular security and characteristics. Although this approach solves the problem of sharing the resources, they do not provide fairness guarantees among VAPs that serve different number of clients. Specifically, as the IEEE 802.11 protocol grants stations equal opportunities of accessing the channel, in such scenarios the throughput performance of the VAPs will be proportional to their number of users. Fortunately, our proposed flexible architecture allows easily address the unfairness problem by utilizing its ability to tune CW_{min} parameter at different VAPs. Specifically, we develop a novel algorithm [20] which maximizes the total throughput shared by VAPs while providing fairness guarantees, by tuning CW_{min} at different VAPs.

It is worth to note that, in addition to common logical virtualization, our architecture is able to provide more intelligent virtualization support thanks to its capability to program low level MAC functionality. Specifically, as the WMP allows running in parallel several MAC programs on a wireless card and switching between them, it is possible to provide ***full*** virtualization of wireless resources with high level of isolation and support of heterogeneous medium access policies. For example, as shown in [10], it is possible to run two different virtual WLAN on single card, one of which uses a contention-based access (DCF or EDCA) and another deterministic access (TDM), with full isolation of resources between them.

4. Implementation and prototyping

The enhancements described in the previous sections could be implemented with tailored modifications of 802.11 firmware, as previous researches have done, for instance in the in the case of Idle Sense [21], in which authors were able to program new backoff operation, but in collaboration with Intel. A key advantage of our approach, in contrast to this ad-hoc tweaking of the interface, is that the modular framework provides open interfaces with enough flexibility to program these sorts of enhancements without the need to access the internals of the device, or to contact hardware vendors. In this way, not only the enhancements described before can be implemented, which alter substantially the operation of 802.11, but also new ones can be testbed with existing commercial devices.

A first version of the Wireless MAC Processor (WMP) is readily available in a `github` repository [22]. The processor has been extensively tested on a commercial wireless card, and the complete code consists of the following components (in addition to a detailed manual):

- The WMP Editor, which is a graphical tool that supports programming new MAC protocols by drawing the state machine (including conditions and actions).
- The WMP bytecode manager, required to compile the resulting graphical state machines into a readable machine code.
- The WMP firmware, which is the "core" component of the WMP. This firmware should substitute the original firmware of the 802.11 card. The firmware is provided as closed source, but the source of the code is available upon request.

In addition to the above components, a set of MAC examples are provided to illustrate the basic "programming cycle" of a MAC extension in the above framework.

Similarly, we are about to release a first version of the `mac80211++` framework. This software, which basically consists of the function container and service scheduler, standardizes the registration and scheduling of different modules running over the same interface. In this way, if the available hardware does not support the WMP, or the envisioned extensions do not require to alter the behavior of the low MAC layer, with `mac80211++` researchers are provided with a modular framework in which programming and extending the standard functionality is much simplified. For instance, the framework already implements consistency checks to test in advance if a given set of operations are supported by the hardware/driver, and provides the means to orchestrate different system operations to prevent e.g. simultaneous calls or race conditions.

Finally, here we list some examples of extensions presented in the previous sections, which have been prototyped using one of the above frameworks:

- For the cases of ACK piggybacking [7] and [8] and MAC virtualization [10], given the need to alter the functionality of the low level layers of the protocol stack, the prototyping was based on the WMP.

- For the cases of the tuning of the CW_{min} [5] and [20], on the other hand, the main requirement is the ability to change the EDCA parameters, which is already supported by recent devices, and therefore the prototyping (that basically consists on triggering the update of the parameters upon the reception of long-timescale events) was based on the mac80211++ framework.

- Finally, to implement the Self-Optimizing proposal [12] there is no need to significantly alter the MAC operation, but a proper scheduling of the sniffing, forwarding and power saving activities is required. Because of this, the mac80211++ resulted extremely adequate to develop the prototype.

5. Conclusion

By developing the modular architecture of wireless access interface and making IEEE 802.11 wireless access programmable we pursue two goals. First, to open the door for efficiency improvements for the basic data transport service, which were difficult, if possible at all, to implement with today's monolithic wireless interface. Second, to enable testing and implementation of new innovative ideas to support new services emerging due to evolution of the needs of operators and users, as well as evolution of the users themselves. A first version of the Wireless MAC Processor – the key component of proposed architecture – is available on-line, including nice graphical tool to program the access operation by simply drawing state machines and bytecode manager to compile the graphical state machines into a readable machine code. We hope that developed architecture will help to increase the impact of the scientific community research on real-life deployments.

References

[1] S. Lucero and S. Carlaw, "Cellular M2M connectivity service providers - the market opportunity for MNOs, MMOs, and MNVOs," *ABI Research Report*, 2009.

[2] *FLexible Architecture for Virtualizable future wireless Internet Access (FLAVIA), http://www.ict-flavia.eu/*.

[3] *FLAVIA project Deliverable Report D2.2.2: Architecture Specification, available at http://www.ict-flavia.eu*.

[4] I. Tinnirello, L. Giarre, and G. Neglia, "MAC design for WiFi infrastructure networks: A game-theoretic approach," *IEEE Transactions on Wireless Communications*, vol. 10, pp. 2510–2522, August 2011.

[5] P. Serrano, P. Patras, A. Mannocci, V. Mancuso, and A. Banchs, "Control theoretic optimization of 802.11 wlans: Implementation and experimental evaluation," *Computer Networks*, no. 0, 2012.

[6] *IEEE Standard for Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements – Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, 2012.

[7] P. Salvador, F. Gringoli, V. Mancuso, P. Serrano, A. Mannocci, and A. Banchs, "Voipiggy: Implementation and evaluation of a mechanism to boost voice capacity in 802.11 WLANs," in *INFOCOM*, 2012.

[8] I. Tinnirello, G. Bianchi, P. Gallo, D. Garlisi, F. Giuliano, and F. Gringoli, "Wireless MAC processors: Programming MAC protocols on commodity hardware," in *INFOCOM*, 2012.

[9] P. Gallo, D. Garlisi, F. Giuliano, F. Gringoli, and I. Tinnirello, "WMPS: A Positioning System for Localizing Legacy 802.11 Devices," *submitted to IEEE Transactions on Smart Processing and Computing*.

[10] G. Bianchi, P. Gallo, D. Garlisi, F. Giuliano, F. Gringoli, and I. Tinnirello, "Maclets: active MAC protocols over hard-coded devices," in *CoNEXT*, 2012.

[11] P. Bahl, R. Chandra, P. Lee, V. Misra, J. Padhye, D. Rubenstein, and Y. Yu, "Opportunistic use of client repeaters to improve performance of WLANs," *IEEE/ACM Transactions on Networking*, vol. 17, pp. 1160 –1171, aug. 2009.

[12] A. Garcia-Saavedra, B. Rengarajan, P. Serrano, D. Camps-Mur, and X. Pérez-Costa, "Solor: Self-optimizing wlans with legacy-friendly opportunistic relays," *Submitted work*.

[13] A. Safonov, A. Lyakhov, and A. Krasilov, "Parameterized QoS for 802.11s mesh networks," in *online: https://mentor.ieee.org/802.11/dcn/12/11-12-0422-01-0wng-parameterized-qos-for-802-11s-mesh-networks.ppt*.

[14] A. Krasilov, A. Lyakhov, and A. Safonov, "Interference, even with MCCA channel access method in ieee 802.11s mesh networks," in *Proceedings of IEEE Eighth International Conference on Mobile Ad-Hoc and Sensor Systems*, pp. 752–757, 2011.

[15] E. Khorov and A. Safonov, "Multiple metrics in MANET with end-to-end QoS support for unicast and multicast traffic," *Lecture Notes in Computer Science*, vol. 6235, pp. 251–262, 2010.

[16] E. Khorov, A. Lyakhov, and A. Safonov, "Flexibility of routing framework architecture in IEEE 802.11s mesh networks," in *Proceedings of IEEE Eighth International Conference on Mobile Ad-Hoc and Sensor Systems*, pp. 777–782, 2011.

[17] A. Safonov, A. Lyakhov, A. Urgenson, and O. Sokolova, "Wireless groupcast routing with palette of transmission methods," *Lecture Notes in Computer Science*, vol. 7642, pp. 97–108, 2012.

[18] A. Lyakhov, P. Nekrasov, D. Ostrovsky, A. Safonov, and E. Khorov, "Analysis of the joint use of the proactive and reactive methods of the topology information dissemination in ad-hoc wireless networks," *Journal of Communications Technology and Electronics*, vol. 57, pp. 1322–1330, 2012.

[19] E. Shvets and A. Lyakhov, "Mathematical model of MCCA-based streaming process in mesh networks in the presence of noise," in *Proceedings of Wireless Communications and Networking Conference (WCNC)*, pp. 1887–1892, 2012.

[20] A. Banchs, P. Serrano, P. Patras, and M. Natkaniec, "Providing throughput and fairness guarantees in virtualized WLANs through control theory," *Mob. Netw. Appl.*, vol. 17, pp. 435–446, Aug. 2012.

[21] Y. Grunenberger, M. Heusse, F. Rousseau, and A. Duda, "Experience with an implementation of the idle sense wireless access method," in *Proceedings of the 2007 ACM CoNEXT conference*, CoNEXT '07, (New York, NY, USA), pp. 24:1–24:12, ACM, 2007.

[22] *Wireless MAC Processor source code repository, available at https://github.com/ict-flavia/Wireless-MAC-Processor*.

