

Towards a Hybrid Management Protocol Stack in Programmable Wireless Networks

Thomas Le Tarnec and Georgios Z. Papadopoulos

*IMT Atlantique,
IRISA, UMR CNRS 6074,
Rennes, France
thomas.le-tarnec@imt-atlantique.fr
georgios.papadopoulos@imt-atlantique.fr*

Guillaume Le Gall

*University of Rennes,
IRISA, UMR CNRS 6074,
Rennes, France
guillaume.le-gall@irisa.fr*

Julien Montavont and Thomas Noël

*University of Strasbourg,
ICube, UMR CNRS 7357,
Strasbourg, France
montavont@unistra.fr
noel@unistra.fr*

Abstract—Reconfiguring the protocol stack of deployed wireless devices remains a critical challenge in programmable wireless networks. Existing approaches, like Over-The-Air (OTA) firmware updates or Software-Defined Network (SDN)-based solutions, rely on the data plane as their transport channel, creating a circular dependency that risks permanent device isolation during reconfiguration. To address this limitation, we propose the Hybrid Management Protocol Stack (HYMA-PS), a protocol stack for Software-Defined Radio (SDR)-enabled devices that provides a persistent and dedicated management channel, decoupled from the data plane and independent of the wireless technology in use. The key idea is to leverage SDR and network virtualization to host a dedicated management stack alongside the data stack on the same hardware, activated periodically over a single-hop star topology. This design eliminates the circular dependency without requiring additional hardware, while allowing the data plane to adopt any topology suited to the application. We evaluate HYMA-PS through ns-3 simulations, showing that jointly tuning the Beacon Interval and Management Time enables a knee-point operating condition achieving a convergence time of 2.3 s with a 14% PDR loss. These preliminary results are promising and constitute a first step toward a fully programmable wireless management plane, giving incentive to further investigate this solution.

Index Terms—SDR, SDN, Programmable Wireless Networks, Protocol Stack Reconfiguration, Management Plane

I. INTRODUCTION

In computer networks, functionality is divided into two planes: the data plane, which is responsible for forwarding data packets from their source to their destination, and the control plane, which computes the forwarding rules. These two planes are co-located within network devices (routers and switches) to distribute processing load and enhance robustness against node failures. However, this tight coupling limits flexibility and makes network-wide optimization more complex. Software-Defined Network (SDN) addresses these limitations by decoupling the control plane from the data plane, allowing forwarding devices to focus solely on packet forwarding while a logically centralized controller computes and installs forwarding rules across the network [1]. This separation also introduces a third functional component: the management

plane, which is responsible for configuring, monitoring, and orchestrating the overall network behavior.

Applying the SDN paradigm to wireless networks remains challenging. In particular, the management and control functions rely on wireless links, which are inherently less stable and more variable than wired ones. As a result, collecting consistent and up-to-date network state information becomes more difficult due to fluctuating connectivity, interference, and packet loss. Moreover, the Medium Access Control (MAC) layer plays a central role in overall performance, as it governs channel access, contention, and scheduling.

However, rule-based forwarding decisions in traditional SDN abstractions primarily operate at the network layer and cannot directly influence MAC-layer mechanisms. This limits the extent to which SDN control logic can optimize key performance factors such as airtime usage, collision probability, or link reliability. However, recent advances in Software-Defined Radio (SDR) and network virtualization provide the necessary building blocks to enable a fully programmable network stack. This programmability can span all layers, from the physical layer (e.g., channel selection, modulation, coding schemes) up to routing and transport functions. For instance, the communication protocol stack can be decomposed into lightweight virtual network functions that are instantiated and managed dynamically, as proposed in [2]. These functions can be updated during operation, including through Over-The-Air (OTA) reconfiguration, enabling networks to adapt their behavior in response to changing channel conditions, traffic demands, or application requirements.

With these capabilities, it becomes possible to deploy distinct communication stacks for different functional planes, rather than relying on a single homogeneous network design. In particular, the data plane and the management plane can be optimized independently, since they have fundamentally different requirements. Typically, the data plane can be designed for energy efficiency and scalability, in a multi-hop, low-power wireless network representative of constrained IoT deployments. In contrast, the management plane can be built on a high-throughput, low-latency star topology, which simplifies connectivity and improves reliability for control and monitoring operations. This separation allows

This research has been funded by the French National Research Agency (ANR) under project ANR-23-CE25-0008. With a view to open access publication, the author has applied for an open access CC-BY licence for any manuscript accepted for publication (AAM) as a result of this submission.

the management infrastructure to remain stable and efficient, while the data plane is optimized for energy-aware and large-scale wireless communication.

In this context, wireless nodes are required to dynamically switch between the data plane and the management plane depending on network conditions and operational needs. In this article, we investigate how such transitions can be performed in an efficient manner.

In particular, we aim to identify the best trade-off between reconfiguration reactivity and data plane preservation, where reactivity refers to how quickly nodes converge to a new configuration, and preservation refers to minimizing disruptions such as packet loss and increased delay for ongoing data traffic during the transition process. More specifically, the contributions of this paper are as follows:

- We propose the Hybrid Management Protocol Stack (HYMA-PS), a protocol stack for SDR-enabled devices that provides a persistent and dedicated management channel, which decouples configuration delivery from the data plane state without requiring additional hardware.
- We define a MAC-layer protocol on top of a dedicated radio channel, based on a time-division scheme that shares the single SDR interface between the management and data stacks. This scheme is governed by two tunable parameters that control the trade-off between reconfiguration reactivity and data plane availability.
- We evaluate the HYMA-PS through ns-3 simulations, first analyzing how the management period frequency and duration each affect the convergence time, then characterizing the resulting trade-off between reconfiguration reactivity and data plane preservation, and identifying the operating points that offer the best compromise.

II. RELATED WORK

A. SDN for wireless networks

1) *SDN for Wireless Networks*: Originally proposed for wired networks, the SDN paradigm has been progressively extended to wireless environments. Early efforts focused on bringing SDN programmability to IEEE 802.11 infrastructure, extending the control plane to support wireless port abstractions, introducing physical and logical wireless ports to expose radio capabilities, statistics, and events to the controller, and demonstrating controller-driven handoff management [3].

In cellular and heterogeneous networks, SDN has been leveraged to orchestrate multi-Radio Access Technologies (RAT) deployments through virtualized radio access technologies, enabling predictive handover with reduced latency and improving backhaul resilience through centralized self-healing mechanisms [4]. More recent work has addressed throughput-optimal routing and scheduling under interference constraints in wireless SDN, demonstrating that Universal Max-Weight algorithms can be made compatible with SDN flow installation while doubling throughput compared to shortest-path routing [5]. The scalability of multi-hop routing protocols in

SDN-based 5G/D2D environments has also been studied, highlighting trade-offs between proactive and reactive approaches under varying node density and mobility [6].

In constrained wireless networks, several solutions have been proposed to bring SDN to low-power multi-hop topologies. TinySDN [7], SDN-WISE [8], Atomic-SDN [9], and SDN-TSCH [10] each address the challenge of reliably delivering control traffic over lossy multi-hop topologies, as surveyed in [11]. These approaches adapt the SDN control channel to operate under the constraints of low-power links, limited bandwidth, and energy budgets typical of IoT deployments. Closer to our work, VERO-SDN [12] introduces a dedicated out-of-band control interface over a single-hop star topology, eliminating multi-hop constraints, but relies on a second radio interface.

However, across all these domains, SDN solutions remain bound to the data plane channel and tied to a specific wireless standard, limiting their applicability to heterogeneous and fully reconfigurable network deployments.

2) *OTA Firmware Update*: Reconfiguring deployed devices has traditionally relied on over-the-air firmware updates delivered over the existing data plane [13], [14], standardized by the IETF SUIT architecture [15]. These approaches are however limited to application-level updates and do not support reconfiguration of Physical (PHY) and MAC layers, which remain tightly coupled to hardware. More critically, as documented in [14], they exhibit an absolute dependence on the underlying transport medium: using the data plane to reconfigure the data plane creates a circular dependency that risks permanent device isolation if the stack transitions mid-delivery. No existing solution simultaneously provides a dedicated management channel, independent of the data plane, without additional hardware, and capable of delivering full protocol stack configurations. The HYMA-PS addresses this gap by maintaining a persistent, standard-agnostic management channel that decouples configuration delivery from the data plane state.

III. PROBLEM STATEMENT AND SYSTEM MODEL

A. Problem Statement

Today, modifying the PHY and MAC layers of a deployed device requires physical hardware replacement, as these layers remain tightly coupled to dedicated radio hardware, driving premature network obsolescence and preventing operators from adopting more efficient protocols. The combination of SDR and network virtualization addresses this limitation by enabling the protocol stack to be fully implemented, abstracted, and dynamically reconfigured in software on top of programmable hardware platforms. Together, these technologies make it possible to envision a fully programmable network stack, spanning from the physical layer (e.g., modulation, channel selection) to higher-layer functions such as MAC, routing, and transport. In such a context, previously impractical scenarios become achievable, such as deploying an optimized MAC protocol to hundreds of devices without service interruption, or simultaneously transitioning an Internet of

Things (IoT) network to a new protocol stack in a coordinated manner, preventing fragmentation into incompatible subsets.

However, this flexibility introduces a critical operational risk: most existing approaches, including OTA firmware mechanisms and SDN-based solutions, use the data plane itself as the transport substrate for management operations. This coupling is fundamentally fragile, as any reconfiguration of the data plane may temporarily degrade, or even break, the very communication channel required to control it. In extreme cases, the network can become partially or completely isolated during the transition, preventing completion of the update process. This creates a circular dependency: the data plane must be operational to reconfigure itself, yet reconfiguration is precisely what may disrupt its operation.

To address this limitation, we argue for a strict separation between two dedicated protocol stacks: a stable, resilient stack reserved for the management plane, ensuring continuous and reliable connectivity during reconfiguration phases, and a fully reconfigurable stack dedicated to the data plane, optimized for application-driven performance requirements. This separation avoids the circular dependency and enables safe, network-wide protocol evolution.

This leads to the central problem addressed in this article: given a network of SDR-enabled devices, how can a controller maintain a persistent management channel, independent of the data plane technology in use, in order to reliably deliver protocol stack configurations and orchestrate their synchronized activation, without depending on the data plane being reconfigured?

B. System model

To address this problem, we consider a wireless network composed of two types of entities: a single controller, responsible for orchestrating all protocol stack reconfigurations, and a set of end-devices, which receive and apply configurations on their data plane. This model is based on three assumptions:

- **Homogeneous nodes:** all end-devices share identical hardware and software capabilities, ensuring any configuration delivered by the controller is uniformly applicable.
- **SDR capabilities:** all devices are SDR-enabled, with sufficient PHY-layer throughput and range to deliver configuration payloads to all devices within a single hop if necessary.
- **Single-hop reachability:** every end-device is reachable by the controller within a single wireless hop on the management channel, which contributes to reducing and stabilizing management-plane latency.

Building on this system model, the HYMA-PS is designed around three core objectives:

- **Persistent connectivity:** the management plane must remain operational regardless of the state of the data plane, ensuring that every device can always be reached and reconfigured.
- **Protocol-agnostic interoperability:** the management plane must operate independently of the wireless technology

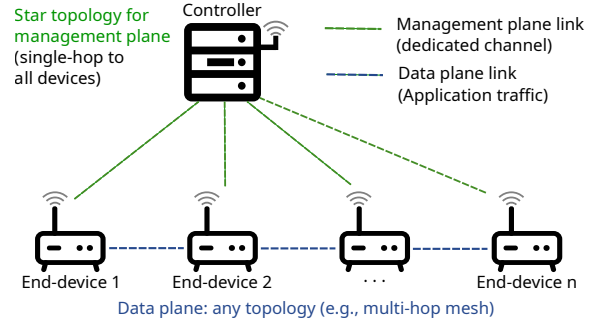


Fig. 1: Architecture overview.

used in the data plane, enabling coordinated transitions between heterogeneous communication stacks.

- **Synchronized reconfiguration:** protocol stack updates must be deployed and activated in a coordinated manner across all target devices, preventing involuntary network heterogeneity.

IV. HYBRID MANAGEMENT PROTOCOL STACK

Building on the three-plane separation introduced in Section I, where the management plane relies on a dedicated channel between the controller and end-devices distinct from the data plane, the HYMA-PS proposed in this article instantiates this concept in SDR-enabled wireless networks. HYMA-PS provides a persistent, protocol-agnostic communication channel between the controller and end-devices, independent of the state of the data plane. Its fundamental property is that any device equipped with the management stack remains reachable and reconfigurable by the controller, even while its data-plane stack is being replaced or transitioned to a new wireless standard.

A. Design Rationale

The HYMA-PS is built around two core design choices, namely an out-of-band management channel and a star topology for the management plane, each motivated by the objectives defined in Section III.

In-band SDN approaches suffer from a fundamental coupling between the management and data planes. Any disruption of the data plane, such as that induced by a protocol stack reconfiguration, may also interrupt the management channel. Existing out-of-band solutions address this limitation by relying on a dedicated second radio interface, but at the cost of additional hardware and an increased energy budget, making them more expensive in wireless deployments and often incompatible with resource-constrained devices [12]. The HYMA-PS overcomes this trade-off by leveraging SDR to host a secondary protocol stack on the same radio hardware, which is activated periodically and independently of the data-plane stack, thereby satisfying the first objective defined above without requiring additional hardware.

While multi-hop mesh topologies are commonly used in wireless data planes to extend coverage and improve energy

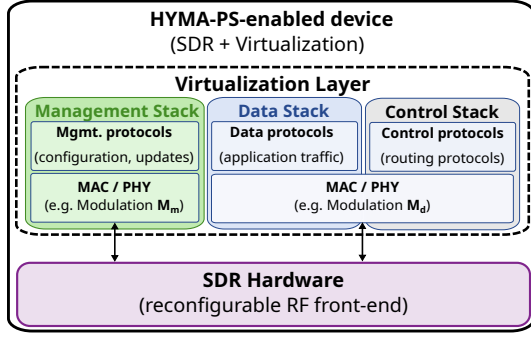


Fig. 2: HYMA-PS-enabled Device.

efficiency, they introduce routing instability, parent-loss vulnerabilities, and forwarding overhead.

Although these characteristics may be acceptable for application traffic, they are undesirable for the management plane, which primary requirements are reliable and predictable reachability, as well as sufficient throughput to rapidly disseminate protocol-stack updates and reconfiguration commands. To satisfy these requirements, the HYMA-PS adopts a star topology for the management plane, in which every end-device can be reached by the controller within a single wireless hop, as illustrated in Fig. 1. This relies on the assumption, stated in our system model, that the SDR provides sufficient PHY-layer range and throughput. This design choice is independent of the data-plane topology, which remains unconstrained and may therefore be optimized for application requirements, including multi-hop mesh deployments. By eliminating routing dependencies from the management plane, this architecture avoids route convergence and parent-selection issues, reduces path variability, and contributes to lowering and stabilizing management-plane latency. Furthermore, single-hop communication allows the management channel to devote its capacity directly to control traffic, enabling faster dissemination of large protocol-stack updates. As a result, the controller maintains a direct, predictable, and high-capacity communication channel with every device, regardless of the topology currently used by the data plane.

B. Architecture Overview

Each HYMA-PS-enabled device uses SDR and network virtualization to host multiple coexisting protocol stacks entirely in software; the management stack operates on a dedicated channel and handles all configuration exchanges with the controller, while the data stack carries application traffic on the data plane, as illustrated in Fig. 2. The operation of these virtualized stacks follows a hybrid, time-shared scheme: rather than dedicating a separate radio to each stack, a single radio interface alternates between management and data periods, activating the corresponding protocol stack in each time window along with its appropriate channel and PHY-layer configuration such as modulation and coding schemes. The alternation between these two stacks follows a periodic temporal structure, which is detailed in the following.

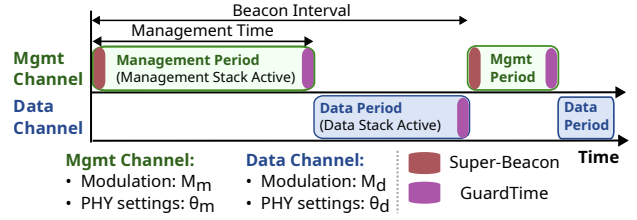


Fig. 3: Time-division structure of the HYMA-PS management period.

C. Management Period Structure

The temporal organization of the HYMA-PS is illustrated in Fig. 3. Each cycle begins with the controller broadcasting a Super-Beacon on the management channel, which announces the upcoming management period and synchronizes all devices. Following the Super-Beacon, a Management Time window opens during which management exchanges take place. At the end of this window, a Guard Time compensates for clock drift before all devices simultaneously switch to the data plane channel for the Data Plane Period.

Two parameters govern this temporal structure: the Beacon Interval (BI), which defines the duration between two consecutive Super-Beacons, and the Management Time (MT), which defines the duration of the management window.

These two parameters define a fundamental trade-off: a longer MT relative to the BI speeds up configuration delivery but reduces data plane availability, while a shorter one preserves the data plane at the cost of slower convergence. The Super-Beacon includes the list of devices the controller intends to reach during the upcoming management period; a device absent from this list and with no pending transmission immediately enters sleep mode after receiving the Super-Beacon, reducing the management overhead on the network.

D. Frame Types

The HYMA-PS defines a small set of frame types exchanged on the management plane:

- Association Request: initiates the joining procedure and embeds the list of configuration identifiers already installed on the device, enabling the controller to immediately determine missing configurations or reject the device based on its protocol inventory.
- Configuration Delivery: delivers missing configurations as a list of pairs $\{id, length\}$ followed by their respective payloads. The MAC layer treats payloads as opaque byte sequences, delegating their interpretation to upper-layer components.
- Configuration Set: signals that all configurations have been delivered and triggers their activation, specifying the protocols and configurations the device must use on its data stack.
- Super-Beacon: provides the periodic synchronization and scheduling framework within which all exchanges take place.

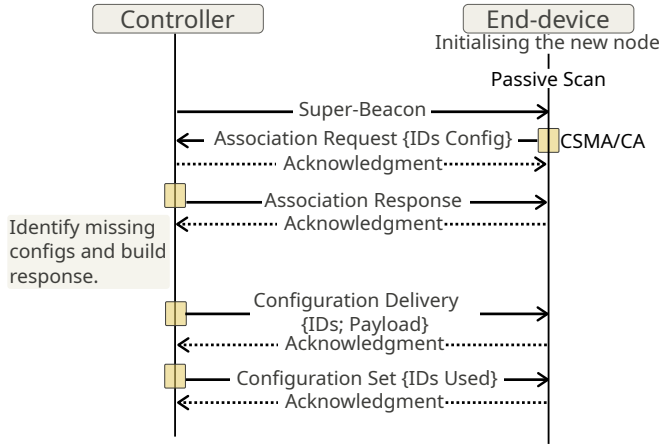


Fig. 4: Sequence diagram for new device integration.

Additional frames (Configuration Request, Configuration Error, Disassociation) handle error recovery and device life cycle management but are not central to the evaluated scenario. Having defined the frame types and their respective roles, we now describe how they are used by the controller and end-devices when a new device joins the network.

E. Typical Messages Sequence : New Device Joining

We now describe the core procedure evaluated in this paper: the integration of a new device joining the network. Upon initialization, the device switches to the management stack and passively listens on the management channel until a Super-Beacon is received. It then issues an Association Request message embedding its installed configuration inventory. The controller immediately identifies missing configurations and delivers them via one or more Configuration Delivery messages, as the upper-layer component is responsible for segmenting configuration payloads to fit within the maximum frame size. Each message is acknowledged, and the controller retransmits up to a maximum number of retries upon failure before proceeding.

Once all configurations have been delivered, the controller sends a Configuration Set message specifying the protocols and configurations the device must activate on its data stack, completing its integration. The device is then fully integrated and begins exchanging data traffic on the data plane. Configuration state management, i.e., tracking which configurations are installed on each device, is delegated to an upper-layer component, keeping the HYMA-PS MAC layer agnostic to configuration management logic and preserving full implementation flexibility. Having defined the HYMA-PS architecture and its operating procedures, we now evaluate the protocol behavior through simulation, focusing on the impact of the BI and MT parameters on the convergence time, as well as the impact induced by the management plane on an active data plane.

TABLE I: Simulation Setup

Management Plane	Value
PHY model	IEEE Std 802.11b-like
Data rate	20 Mbps (2000 ksymbol/s)
Frequency band	2.4 GHz
Max. frame size	2303 bytes
Channel access	Unicast CSMA/CA
Topology	Star (single-hop)
Data Plane	Value
PHY/MAC model	IEEE Std 802.15.4
Data rate	250 kb/s
Topology	Binary tree (multi-hop)
Node spacing	10 m
Traffic pattern	CBR to root (50% saturation)
Scenario	Value
Number of end-devices	25
Configuration sizes	500 B – 32 kB
Repetitions	30 (95% CI)

V. PERFORMANCE EVALUATION

A. Simulation Setup

We evaluate HYMA-PS using the ns-3 network simulator, in which we implemented the protocol by deriving it from the existing Low-power, Low-Rate Wireless Personal Area Network (LR-WPAN) ns-3 module. The complete set of simulation parameters is summarized in Table I. They are selected to represent a heterogeneous Low-power Lossy Wireless Network (LLWN) scenario where the data plane relies on a constrained low-power wireless technology, while the management plane uses a high-rate channel for network monitoring and reconfiguration. The data-plane configuration follows IEEE Std 802.15.4 characteristics, with a 250 kb/s data rate and a multi-hop binary-tree topology representative of LLWN deployments. The 10 m node spacing models a moderately dense deployment, resulting in a network diameter compatible with the considered low-power communication range. The management plane relies on an IEEE Std 802.11b-like PHY operating in the 2.4 GHz band, providing sufficient capacity for exchanging monitoring information and configuration updates. The considered 25-node scenario and configuration sizes from 500 B to 32 kB are selected to evaluate the scalability of the proposed reconfiguration mechanism, from small parameter updates to larger software component transfers. Signal propagation is modeled using a log-distance path loss model.

B. Simulation Results

In this section, the performance evaluation of HYMA-PS is presented through two experiments. The first assesses the impact of the Beacon Interval and Management Time parameters on the convergence time, under varying configuration payload sizes. The second characterizes the trade-off between convergence time and the impact on an active data plane.

1) *Convergence Time*: The convergence time is defined as the time interval between the reception of the first Association Request by the controller and the reception of

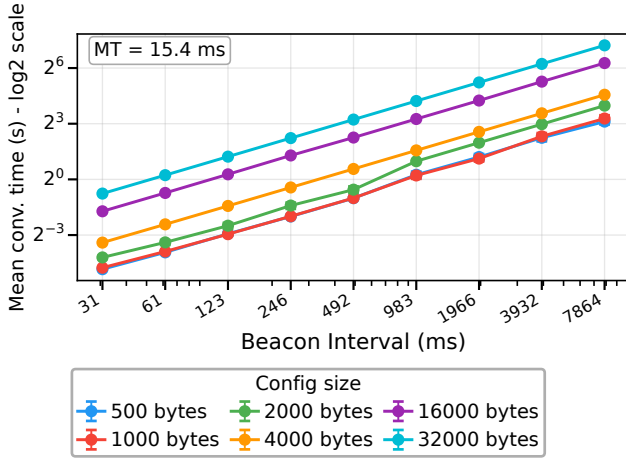


Fig. 5: Mean convergence time as a function of Beacon Interval (MT = 15.4 ms, 25 end-devices, 95% Confidence Interval).

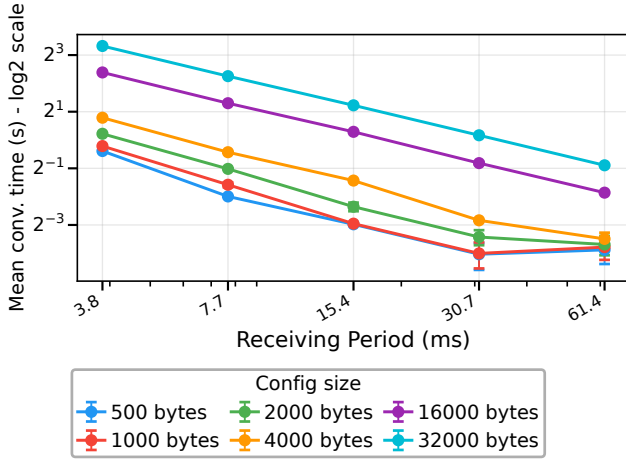


Fig. 6: Mean convergence time as a function of Management Time (BI = 123 ms, 25 end-devices, 95% Confidence Interval).

the last Configuration Set acknowledgment, confirming that all devices are fully configured for data plane participation. Experiments are conducted with 25 end-devices starting simultaneously, all initiating their passive scan at the same time, and configuration payload sizes ranging from 500 to 32,000 bytes, doubling at each step. The simultaneous start of all devices generates initial contention on the management channel, as all Association Requests are issued within the same management period, requiring the controller to serve devices sequentially via unicast Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) exchanges. Fig. 5 shows the convergence time as a function of BI (with MT = 15.4 ms), expressed in milliseconds. As expected, convergence time increases monotonically with BI: since delivering large configurations to 25 devices requires multiple Configuration Delivery frames per device, multiple management periods are necessary to configure all the devices. A larger BI increases the duration between consecutive management windows, directly extending the total time required for all devices to receive their

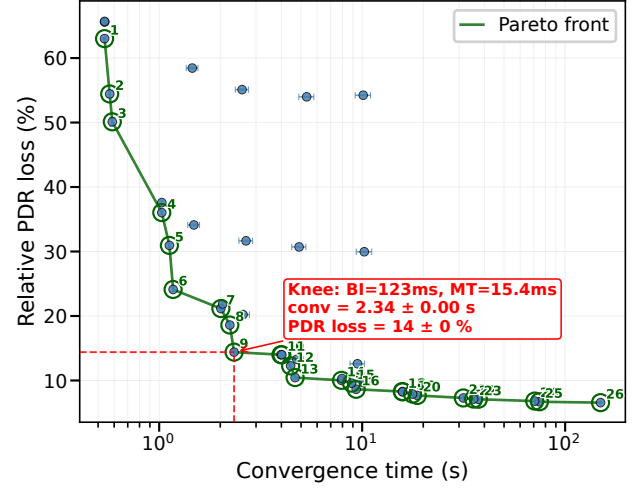


Fig. 7: Convergence time vs. data plane impact trade-off.

configurations.

Fig. 6 shows convergence time as a function of MT (with BI = 123 ms). Increasing the MT extends the management window duration, allowing more Configuration Delivery frames to be transmitted within a single period, and thus reducing the number of periods required. Convergence time therefore decreases as MT increases. In practice, both a shorter BI and a longer MT reduce convergence time; however, both also increase the share of airtime taken from the data plane, which motivates the joint trade-off analysis presented next.

2) *Data Plane Impact*: The Beacon Interval and Management Time govern a fundamental trade-off: more time allocated to the management plane accelerates reconfiguration but reduces the airtime left to the data plane. To capture it, we jointly evaluate, for each (BI, MT) setting, the convergence time and the impact on an active data plane.

The data plane impact quantifies how much the management plane degrades data delivery. We define it as the relative Packet Delivery Ratio (PDR) loss between two scenarios: a baseline where the data plane runs alone (no management plane), and the same data plane running alongside an active management plane. Writing PDR_{base} and PDR_{mgmt} for the global PDR measured in each case, the impact is:

$$PDR_{loss} = \frac{PDR_{base} - PDR_{mgmt}}{PDR_{base}} \times 100 [\%] \quad (1)$$

A higher value thus indicates a stronger degradation of the data plane caused by the management activity.

For this experiment, each device generates Constant Bitrate (CBR) traffic toward the root at 50% channel saturation. The data plane impact is the relative PDR loss caused by the management plane, where the global PDR is the total received over the total sent, aggregated over all 25 links, relative to a baseline without management activity.

Fig. 7 shows the resulting trade-off: each point is a (BI, MT) setting, and the green curve is the Pareto front of non-dominated settings, also reported in Table II. The *knee*

TABLE II: Pareto-optimal (BI, MT) settings at 50% saturation.

N°	BI (ms)	MT (ms)	Conv. (s)	Impact (%)
3	31	15	0.6	50
5	123	31	1.1	31
9*	123	15	2.3	14
15	983	31	8.9	10
26	7864	15	149.4	7

* Knee point of the Pareto front.

point offers the best compromise, reached at $BI = 123$ ms and $MT = 15$ ms with a convergence time of 2.3 s for a 14% PDR loss. Settings favoring fast convergence incur a severe data plane penalty, while those minimizing impact require disproportionately long convergence, confirming that BI and MT must be jointly tuned.

VI. CONCLUSIONS

In this paper, we addressed the challenge of reconfiguring the protocol stack of deployed wireless devices in programmable wireless networks. We identified a fundamental limitation shared by existing OTA firmware update and SDN-based approaches: their reliance on the data plane as the transport channel for management operations, creating a circular dependency that risks permanent device isolation when a reconfiguration disrupts the active stack. To overcome this limitation, we proposed HYMA-PS, a persistent and protocol-agnostic management channel for SDR-enabled wireless networks. By leveraging SDR and network virtualization, HYMA-PS hosts a dedicated management stack alongside the data stack on the same radio hardware, activated periodically over a single-hop star topology and independently of the data plane state. This hybrid design eliminates the circular dependency without requiring additional hardware, while allowing the data plane to adopt any topology suited to the application. We evaluated HYMA-PS through ns-3 simulations, characterizing the trade-off between reconfiguration reactivity and data plane availability governed by the Beacon Interval and Management Time parameters, and quantifying the overhead induced on an active data plane. Our results confirm that the management plane can be operated with a negligible impact on data plane performance when its parameters are properly tuned. These results are however preliminary: HYMA-PS has not yet been compared against existing solutions, and a thorough comparative evaluation is needed to fully assess its benefits and limitations.

Several directions remain open for future work. A first one concerns the network-wide reconfiguration scenario, in which the controller migrates the entire network to a new protocol stack. Beyond this, the current architecture relies exclusively on unicast exchanges, which limits the delivery speed of large configurations that may reach several megabytes; we plan to introduce multicast delivery, along with a suitable acknowledgment mechanism. We further intend to exploit the

synchronization provided by the Super-Beacon to introduce a Time Division Multiple Access (TDMA) mechanism, by splitting the Management Time into two sub-periods: one dedicated to uplink traffic using CSMA/CA, and one dedicated to downlink traffic using TDMA. Finally, we aim to deploy HYMA-PS on a real testbed to validate our simulation results under realistic conditions.

REFERENCES

- [1] D. Kreutz, F. M. V. Ramos, P. E. Verissimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," *Proceedings of the IEEE*, vol. 103, no. 1, pp. 14–76, 2015.
- [2] A. Mahmod, J. Montavont, and T. Noël, "PENSIL: Programmable network stack for low-power lossy IoT networks using lightweight virtualization," *Elsevier Internet of Things Journal*, 2026.
- [3] M. Yan, J. Casey, P. Shome, A. Sprintson, and A. Sutton, "Etherflow: Principled wireless support in SDN," in *2015 IEEE 23rd International Conference on Network Protocols (ICNP)*, 2015, pp. 432–437.
- [4] K. Wang, Y. Wang, D. Zeng, and S. Guo, "An SDN-based architecture for next-generation wireless networks," *IEEE Wireless Communications*, vol. 24, no. 1, pp. 25–31, Feb. 2017.
- [5] Q. M. Nguyen, M. S. Rahman, X. Fu, S. Kompella, J. P. Macker, and E. H. Modiano, "An optimal network control framework for wireless SDN: From theory to implementation," in *MILCOM 2022 – 2022 IEEE Military Communications Conference (MILCOM)*, Rockville, MD, USA, 2022, pp. 102–107.
- [6] S. Ashtari, M. Abdollahi, M. Abolhasan, N. Shariati, and J. Lipman, "Performance analysis of multi-hop routing protocols in SDN-based wireless networks," *Computers and Electrical Engineering*, vol. 97, p. 107393, 2022.
- [7] B. T. de Oliveira, C. B. Margi, and L. B. Gabriel, "TinySDN: Enabling multiple controllers for software-defined wireless sensor networks," in *2014 IEEE Latin-America Conference on Communications (LATINCOM)*, 2014.
- [8] L. Galluccio, S. Milardo, G. Morabito, and S. Palazzo, "SDN-WISE: Design, prototyping and experimentation of a stateful SDN solution for Wireless Sensor networks," in *Proceedings of IEEE INFOCOM*, 2015, pp. 513–521.
- [9] M. Baddeley, R. Nejabati, G. Oikonomou, M. Sooriyabandara, and D. Simeonidou, "Atomic-SDN: Is synchronous flooding the solution to software-defined networking in IoT?" *IEEE Access*, vol. 7, pp. 97 159–97 177, 2019.
- [10] F. Veisi, J. Montavont, and F. Théoleyre, "SDN-TSCH: Enabling software defined networking for scheduled wireless networks with traffic isolation," in *2022 IEEE Symposium on Computers and Communications (ISCC)*, 2022.
- [11] M. Baddeley, R. Nejabati, G. Oikonomou, M. Sooriyabandara, and D. Simeonidou, "Evolving SDN for low-power IoT networks," in *2018 4th IEEE Conference on Network Softwarization and Workshops (NetSoft)*, 2018, pp. 71–79.
- [12] T. Theodorou and L. Mamatras, "A versatile out-of-band software-defined networking solution for the Internet of Things," *IEEE Access*, vol. 8, pp. 140 739–140 755, 2020.
- [13] K. Zandberg, K. Schleiser, F. Acosta, H. Tschofenig, and E. Baccelli, "Secure firmware updates for constrained IoT devices using open standards: A reality check," *IEEE Access*, vol. 7, pp. 71 907–71 920, 2019.
- [14] J. Gu, S.-S. Lee, and H. Kang, "DSME-FOTA: Firmware over-the-air update framework for IEEE 802.15.4 DSME MAC to enable large-scale multi-hop industrial IoT networks," *Internet of Things*, vol. 27, p. 101239, 2024.
- [15] B. Moran, H. Tschofenig, D. Brown, and M. Meriac, "A firmware update architecture for Internet of Things," Internet Engineering Task Force, RFC 9019, 2021.