Slotted, synchronised and modular framework for the evaluation of channel access policies in dense IoT network using FIT/CorteXlab

Amaury Paris, Leonardo Sampaio Cardoso, Jean-Marie Gorce

EUROPEAN COOPERATION
IN SCIENCE
AND TECHNOLOGY

EURO-COST

SOURCE: INRIA, INSA Lyon, CITI

CA20120
TD(22)02TD(22)02056
Lyon, France
June 13-15, 2022

Slotted, synchronised and modular framework for the evaluation of channel access policies in dense IoT network using FIT/CorteXlab

Amaury Paris, Leonardo S. Cardoso, Jean-Marie Gorce

6 Av. des arts, Villeurbanne
Phone: 04 72 43 73 22
Fax: X
Email: amaury.paris@insa-lyon.fr

Slotted, synchronised and modular framework for the evaluation of channel access policies in dense IoT network using FIT/CorteXlab

Amaury Paris, Leonardo Cardoso, Jean-Marie Gorce
INRIA, INSA Lyon, Univ Lyon, CITI
firstname.lastname@insa-lyon.fr

Abstract

IoT communications may suffer from extensive collisions when an ALOHA-style transmission policy is used. The study of new decentralized channel access policies promises to reduce energy consumption, latency and errors due to retransmission induced by such collisions. In this paper we describe a Slotted and Synchronized multi-Sources experimental framework for the evaluation of new Channel Access Policies (S3-CAP). Each node transmission to the base station is made in a shared slotted time-frame where a decentralized access policy determines which slot to use. Then, a modular and replaceable physical (PHY) layer is used to create the transmitted signal.
The IoT network can be either emulated on a local computer with a simulated channel or deployed in the FIT/CorteXlab testbed where slot synchronisation between all nodes is provided by a clock distribution module. This open-source framework, designed in a GNU Radio environment, opens perspectives to design, evaluate and compare decentralised random access strategies for M2M communications.

Index Terms *- IoT, channel access policy, PHY, MAC, Software Defined Radio*

1. Introduction

The growing interest in IoT, followed by a rising number of connected devices, brought significant attention on the importance of efficient channel access policies. Indeed, ALOHA-style policies work well when adopted in small scale networks. However, when the number of devices or the traffic load increases, the transmission efficiency decreases. Data retransmission creates latency, leads to a higher energy consumption and reduces devices lifespan, making the development of new channel access policies a subject of interest.

Wireless communications research could profit from robust experimentation frameworks to test and validate new models. The S3-CAP framework presented herein offers a solution to emulate large IoT-like network with slotted transmissions. It is derived from the original EPHYL project [1] further extended in the ARBURST project, and is made to easily integrate new channel access policies for evaluation. It also ensures precise time synchronisation between nodes (actual or emulated), while presenting a modular PHY layer structure. Any access policies or GNU Radio PHY layer can be plugged in the proposed framework, to be evaluated either on a local computer with a simulated channel or within the FIT/CorteXlab radio testbed in a realist and controllable environment.

2. FIT CorteXlab testbed

FIT/CorteXlab [2, 3] is a radio testbed which offers a substantial set of software defined radio (SDR) nodes combined with synchronization equipments and remote task management tools. It is located in a shielded room, isolated from any outside interference. Each SDR node is composed of a local server, responsible for the signal processing part, paired with an universal software radio peripheral (USRP). It offers a wide flexibility of center frequencies, bandwidths, transmit powers combined with the possibility to implement nearly any desired modulation and coding schemes.

The shielded room serves a double role: it prevents interference coming from outside from disturbing the experiments as well as preventing internal signals from interfering to the outside, which lifts the limitation on experimenting with restricted frequencies. This effectively makes the transmission channel, and therefore any experiment, reproducible. Furthermore, all nodes have the possibility to use either an internal or an external clock reference, the latter done with the help of clock distribution module (octoclock from National Instruments), for sampling speed and time synchronisation.

This allows users to set up large wireless network for realistic testing in a controlled environment. The platform is accessible remotely through a service interface and open to anyone over the Internet [3].

3. Design architecture

3.1. Overview

The S3-CAP framework is built to develop and evaluate channel access policies in dense IoT network. The original implementation was designed with the same purpose and made to deploy an IoT-like network with its key features and two classes of nodes, Base Station (BS) and Sensor (SN), as seen in Fig. 1.

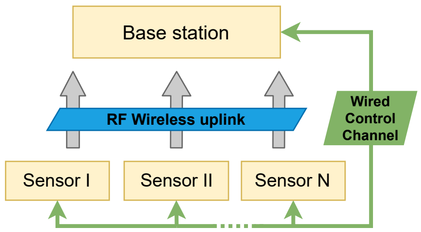

Figure 1. Framework layout

When deployed, each SN instance individually decides, based on an access policy, when to transmit its packet on the radio uplink channel. The content of the packet, one symbol, is to be decided or left to a default value for each transmission. BS and sensors also have the ability to exchange messages through a wired error-free control channel (CCH), in uplink and downlink. The content of those messages is to be decided based on the need of the implemented access policy, or left blank if not needed.

3.2. Slotted transmission

The RF uplink channel presented in this framework is organised in frames and slotted transmissions. As presented in Fig. 2, the time-table of each frame is composed of three parts.

- **Processing:** process RX packets (BS) and received CCH messages (BS and SN) from previous frame to set packets, slots and CCH messages of the next iteration, based on an access and content policy. Signals samples are generated and stored to be transmitted at the very start of the designated slots;
- **Control:** send uplink and downlink CCH messages and apply synchronisation protocol - see Sec. 3.4;
- **Transmission:** transmission on the RF uplink channel of defined packets on selected slots for each sensors. Each slot is followed by a guard interval for the BS to store the decoded data and to avoid overlapping.

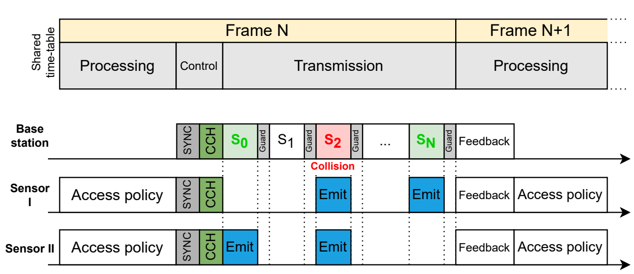

Figure 2. Shared time frame

The number of slots per frame and the duration attributed to each task in the time-table can be configured.

3.3. Control channel

As described on Fig. 1 and Fig. 2, on top of the wireless uplink channel, the framework has a wired and error free control channel, in uplink and downlink. At the beginning of each frame, individual messages can be transmitted from each sensor to the BS and from the BS to each sensor. The content of those messages, easy to program, is to be defined to serve the design and the implementation of the channel access policy.

3.4. Synchronisation

The framework can be tested either inside the FIT/CorteXlab testbed or on a local computer with a simulated channel. Both situations require a synchronisation mechanism to ensure that slots period occur simultaneously, on all sensors and BS.

In local mode, all clocks are inherently synchronized and each sensor continuously produces "zeros" samples between each transmission slots, at the given sample rate, to ensure the time continuity of the frame. In this context, a beacon is sent by the BS to all nodes. Each node can then compute the remaining number of samples to produce before the transmission period, to ensure synchronization with the rest of the network.

When running the framework in the FIT/CorteXlab testbed, no inherent synchronization is possible since multiple physical nodes with their own clocks are used. In this case, only the signal samples to be sent are produced and given by each sensor to its USRP with a timestamp. The value for this timestamp, indicating the time to emit the signal, is given by the BS to all sensors. The internal clock of each USRP is set by the clock distribution module (octoclock). As is, the framework presents an average 0.7 ms delay error[1].

Both beacon and timestamp values are sent at the beginning of the control period, as shown in Fig. 2

3.5. A modular physical layer

Each sensor is built as presented on Fig. 3. In this configuration, an access control block receives and prepares the access/content policy instructions. Symbols go through a PHY layer to produce the complex samples to be transmitted on the radio channel. Once those samples are generated, they are stored in a scheduler block. This block waits for the beginning of the designated slot to send the samples into the simulated channel or the USRP.

Each blocks has normalized inputs and outputs parameters, to simplify the implementation of various PHY layers and new channel access policies. As is, it is required that the sensor PHY layer produces a predictable number of complex samples.

[1] Average time delay between the beginning of the slot and the detection of the signal at the BS for a set of 700 transmissions and a slot duration of 150 ms

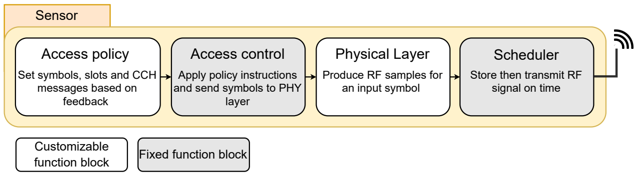

Figure 3. Sensor architecture

As of now, a LoRa[4] PHY layers has been implemented and tested, and a simplified NbIoT PHY layer is under development.

3.6. Power evaluation

A power threshold is implemented in the BS. It provides valuable information on whether packets were not properly demodulated. By default, the BS doesn't know if packets were transmitted [2] by sensors and therefore cannot list missing packets. This power threshold is computed on received samples, as shown in Fig. 4, and enables the detection of one or many erroneous transmissions on one slot. Most cases are due to packet collision because of multiple transmissions on a same slot. This indicator can be useful to compute a metric during the evaluation of a channel access policy.

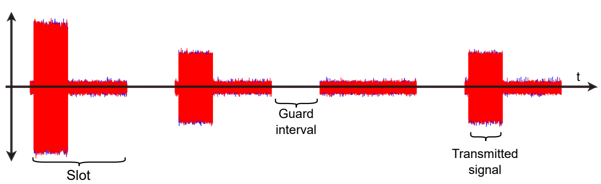

Figure 4. Base station received samples - real and imaginary values are represented in red and blue. All samples not received during slot period are ignored

4. Use case

This framework was built to create and evaluate new channel access policies in IoT networks, such as the one proposed in [5] for a LoRa PHY layer and a reinforcement learning based channel access policy. But its usage can be extended to other subjects related to : i) source traffic statistics, ii) robust source coding, e.g. distributed polar codes, iii) new access policies, based on learning iv) improved multi-level receivers (SIC, joint decoding, ML based).

This framework may also help to better understand interference statistics in IoT wireless communication. This is a challenging problem due to the short duration of packets and the uncoordinated nature of transmissions. At present, there are no experimental studies investigating the baseband IQ samples observed from interferers utilising the NB-IoT protocol. To this end, using the FIT/CorteXlab testbed paired with the slotted framework, we have obtained reproducible experimental data for the baseband IQ samples of simultaneous transmissions [6, 7].

[2]This information could however be communicated through CCH messages

5. Demonstration

The FIT/CorteXlab platform is open to all, you can create your own account and follow several tutorials available on the Fit/CorteXlab Wiki page [3]. All is required is an Internet connection, browser access and a Unix shell.

A demonstration of the presented implementation is available on the tutorial page as "S3-CAP framework". The proposed scenario, with two sensors randomly transmitting messages upon request to a base station, act as a basic example to visualize the project layout and how access policies are implemented and can be easily customized.

The project is also available[8] to be installed on a local computer and to be used with a simulated channel.

References

[1] O. Oubejja et al. "Framework for PHY-MAC layers Prototyping in Dense IoT Networks using FIT/CorteXlab Testbed". In: *Univ Lyon and INRIA CITI* (2019).

[2] Abdelbassat Massouri et al. *CorteXlab: An open FPGA-based facility for testing SDR & cognitive radio networks in a reproducible environment*. IEEE, 2014.

[3] INRIA CITI. *FIT/CorteXlab general information, access and usage*. URL: https://wiki.cortexlab.fr/doku.php.

[4] Amaury Paris, Leonardo Cardoso, and Jean-Marie Gorce. "Dynamic LoRa PHY layer for MAC experimentation using FIT/CorteXlab testbed". In: *European GNU Radio Days* (2021).

[5] Xu Huang et al. "A Reinforcement Learning Based Medium Access Control Method for LoRa Networks". In: *IEEE International Conference on Networking, Sensing and Control (ICNSC)* (2020).

[6] M. Egan et al. "Dynamic interference for uplink SCMA in large-scale wireless networks without coordination". In: *EURASIP Journal on Wireless Communications and Networking* (2018).

[7] M.A. Gritli. "Transmission Pattern Design in Uncoordinated IoT Networks". In: *Master thesis* (2021).

[8] *Framework repository*. URL: https://github.com/AmauryPARIS/gr-ephyl.