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

Journées LPWAN - 2022

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

INSA de Lyon, INRIA, CITI Lab

amaury.paris@insa-lyon.fr

July 7, 2022

Multiple access in IoT

- **A rising number of connected devices**
  Recent interest in IoT, growing networks sizes and loads are making ALOHA-style access policies obsolete

- **The cost of data retransmission**
  Collisions and retransmissions creates latency, higher energy consumption and shorter devices lifespan

Multiple access in IoT

- **A rising number of connected devices**
  Recent interest in IoT, growing networks sizes and loads are making ALOHA-style access policies obsolete

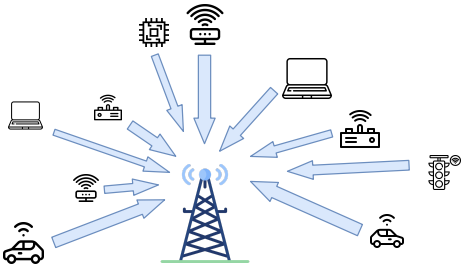
- **The cost of data retransmission**
  Collisions and retransmissions creates latency, higher energy consumption and shorter devices lifespan

- **New channel access policies**
  Slotted and Synchronized multi-Sources framework for new Channel Access Policies

S3-CAP framework

S3-CAP framework

- **Channel access policies**
  Development and evaluation of new channel access policies
- **Robust coding at the transmitter**
  Study of interference, non-orthogonal communication, error-correcting systems
- **Improved multi-level receiver**
  Successive interference cancellation, joint decoding



Framework Layout

Iot-like network :

- Star system;
- BS and sensors nodes;
- RF uplink channel;
- Time frame and slot.

Possibility to emulate sensors.

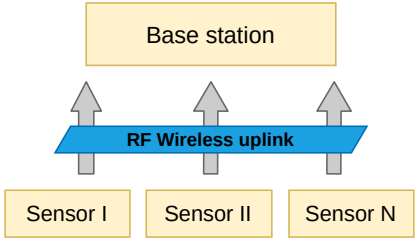


Figure: Star system with uplink channel

Framework Layout

Control channel :

- Uplink;
- Downlink;
- Error-free.

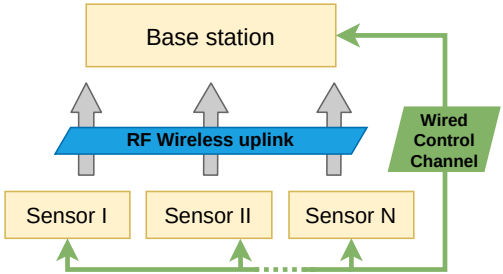


Figure: Error-free control channel

Framework Layout

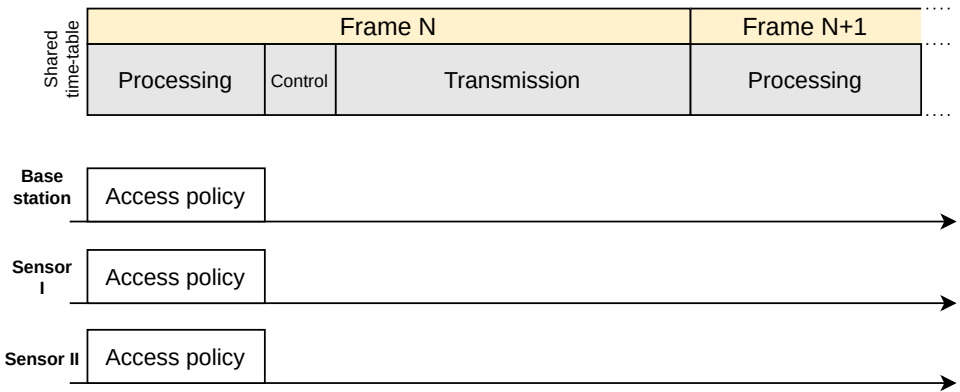


Figure: Shared time table

Framework Layout

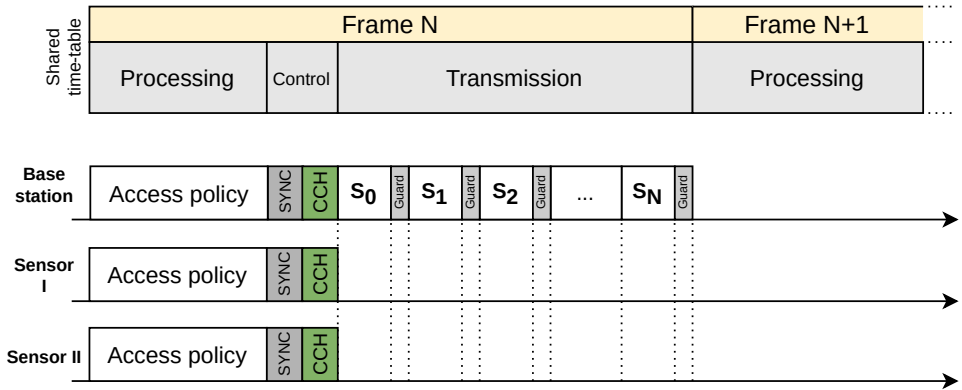


Figure: Shared time table

Framework Layout

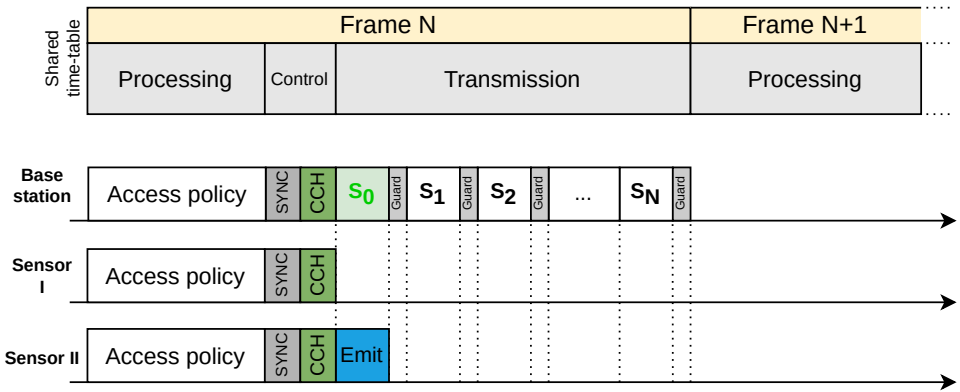


Figure: Shared time table

Framework Layout

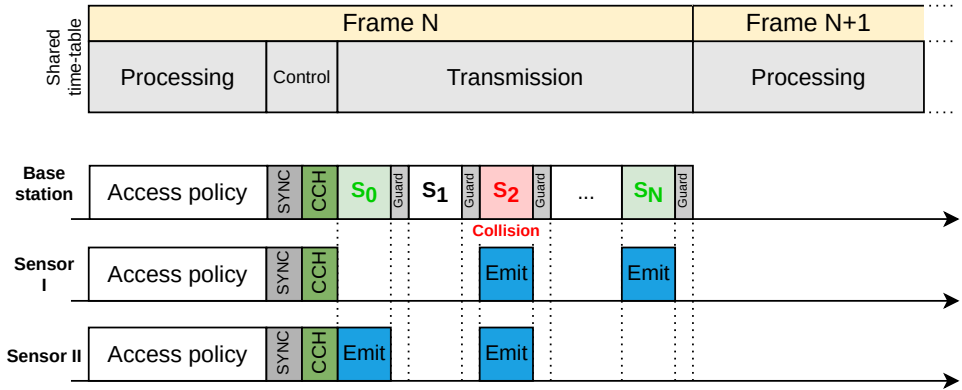


Figure: Shared time table

Framework Layout

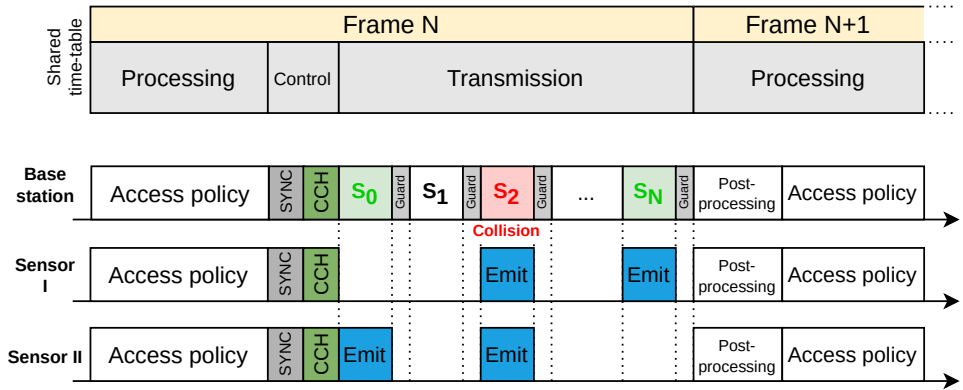


Figure: Shared time table

Your access policies

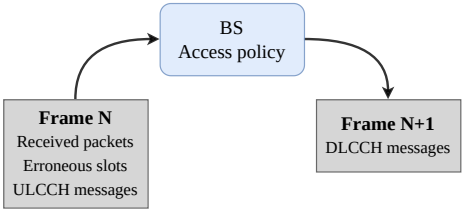


Figure: Base station

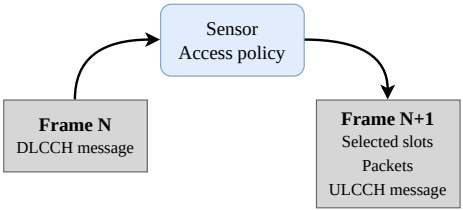


Figure: Sensor

Your access policies

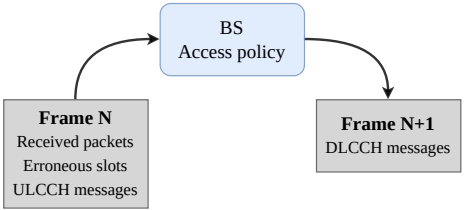


Figure: Base station

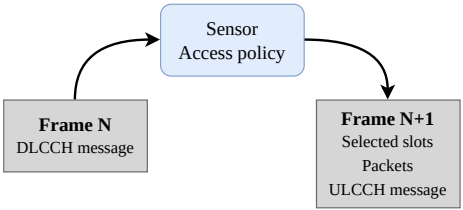


Figure: Sensor

The access policy is yours to implement

Adaptive implementation

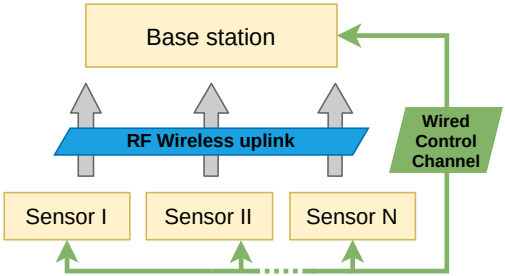


Figure: Framework layout

Adaptive implementation

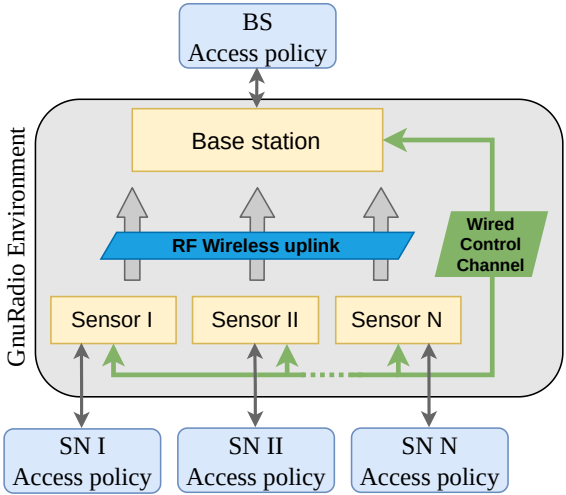


Figure: Decentralized access policy

Adaptive implementation

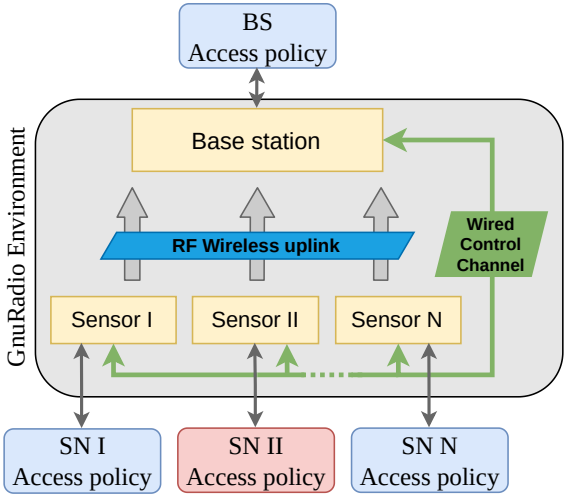


Figure: Decentralized access policy

Adaptive implementation

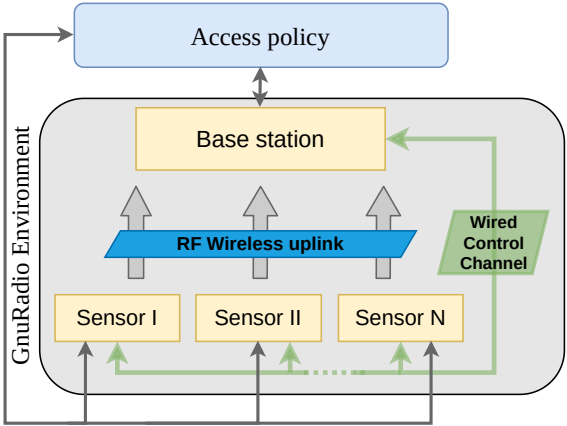


Figure: Centralized access policy

Modular physical layer

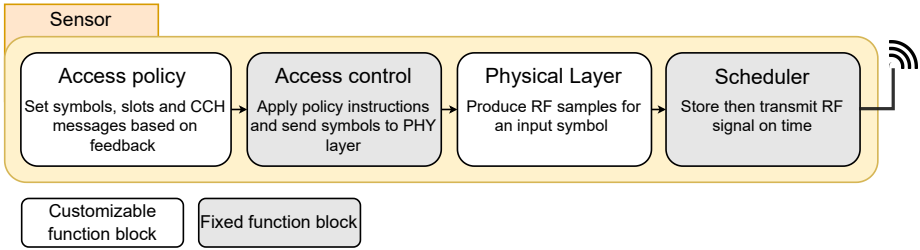


Figure: Sensor architecture

Available PHY layers :

- LoRa : Implemented and tested
- Nb-IoT : Under development
- Yours ...

Locally emulated network

Running the framework on your own local computer :

- Simulated channel - GnuRadio
- Development phase
- Theoretical results

FIT/CorteXlab - Needs and usages

Experimental testbed key elements for IoT/wireless communication

- "Test&try" environment : **free remote access with docker and ssh**;
- Reproducible results : **isolated from exterior interference**;
- Adaptable PHY layer : **Software Defined Radio (SDR)**.

Figure: USRP NI-2932, USRP N2944R, PicoSDR, Octoclock

Testbed for IoT/wireless communication

Figure: FIT/CorteXlab

Tutorial

Available on the Fit/CorteXlab wiki [under development]:

- Two nodes
- Random access policies

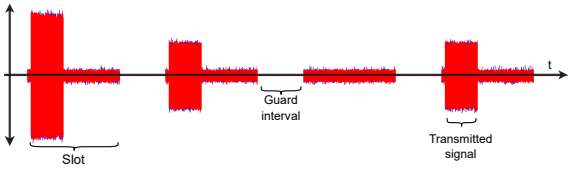


Figure: Error-free control channel

Tutorial

Available on the Fit/CorteXlab wiki [under development]:

- Two nodes
- Random access policies

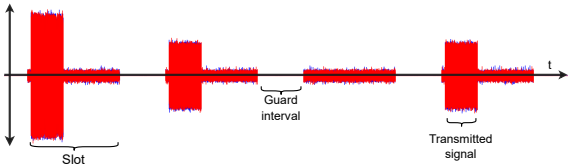


Figure: Error-free control channel

Available functions to send instructions/receive feedbacks
Written in python3

Instance of SN access policy

```
up = upper()

while True:
    pp = up.extract(up.recv())
    print("SN : %s, Frame : %s" % (pp["NODE"], pp["FRAME"]))

    if pp["DLCCH"] == "IoT":
        inst = up.create_SN_inst(   pp["NODE"], pp["FRAME"] +1,
                                    True, [[0, "X"], [1, False]],
                                    ulcch = "Hello there")
    else:
        inst = up.create_SN_inst(   pp["NODE"], pp["FRAME"] +1,
                                    False, [], ulcch = "Hello there")

    up.send(inst)
```

S3-CAP Framework

Current and future work :

- **Nb IoT** : Finalisation of the Nb Iot PHY layer
- **RL access policies** : Apply a RL access policies method designed for LoRa system
- **Interference statistics** : Study the distribution of baseband IQ samples during collision

Thanks !

If you have any additional questions or requests, please contact me at :

amaury.paris@insa-lyon.fr

- **FIT/CortexXlab** : `http://www.cortexlab.fr/`
- **Wiki FIT/CorteXlab** : `https://wiki.cortexlab.fr/doku.php`
- **S3-CAP tutorial** : `https://wiki.cortexlab.fr/doku.php?id=ephyl_framework_v2`