Advanced and disruptive IoT/AI technologies targeting
the smallholder community for increased resilience

Performance of LoRaWAN CSMA in Dense LoRa Networks

Guillaume Gaillard, Congduc Pham

@univ-pau.fr

Université de Pau et des Pays de l’Adour, E2S
UPPA, LIUPPA, Anglet, France

MSWiM 2025

Barcelona, Spain

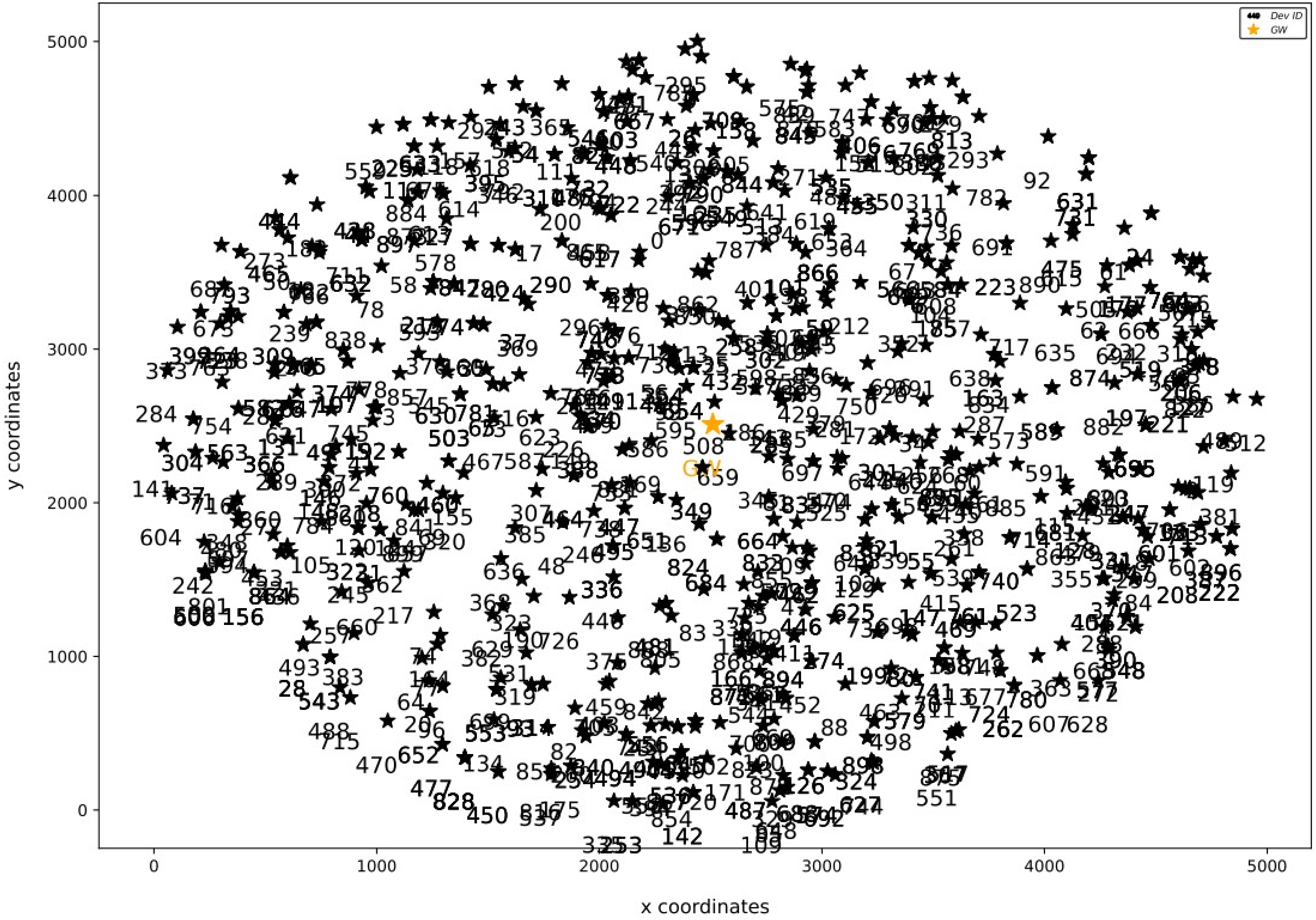
October 27th – 31st, 2025

The 27th International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems

Agenda

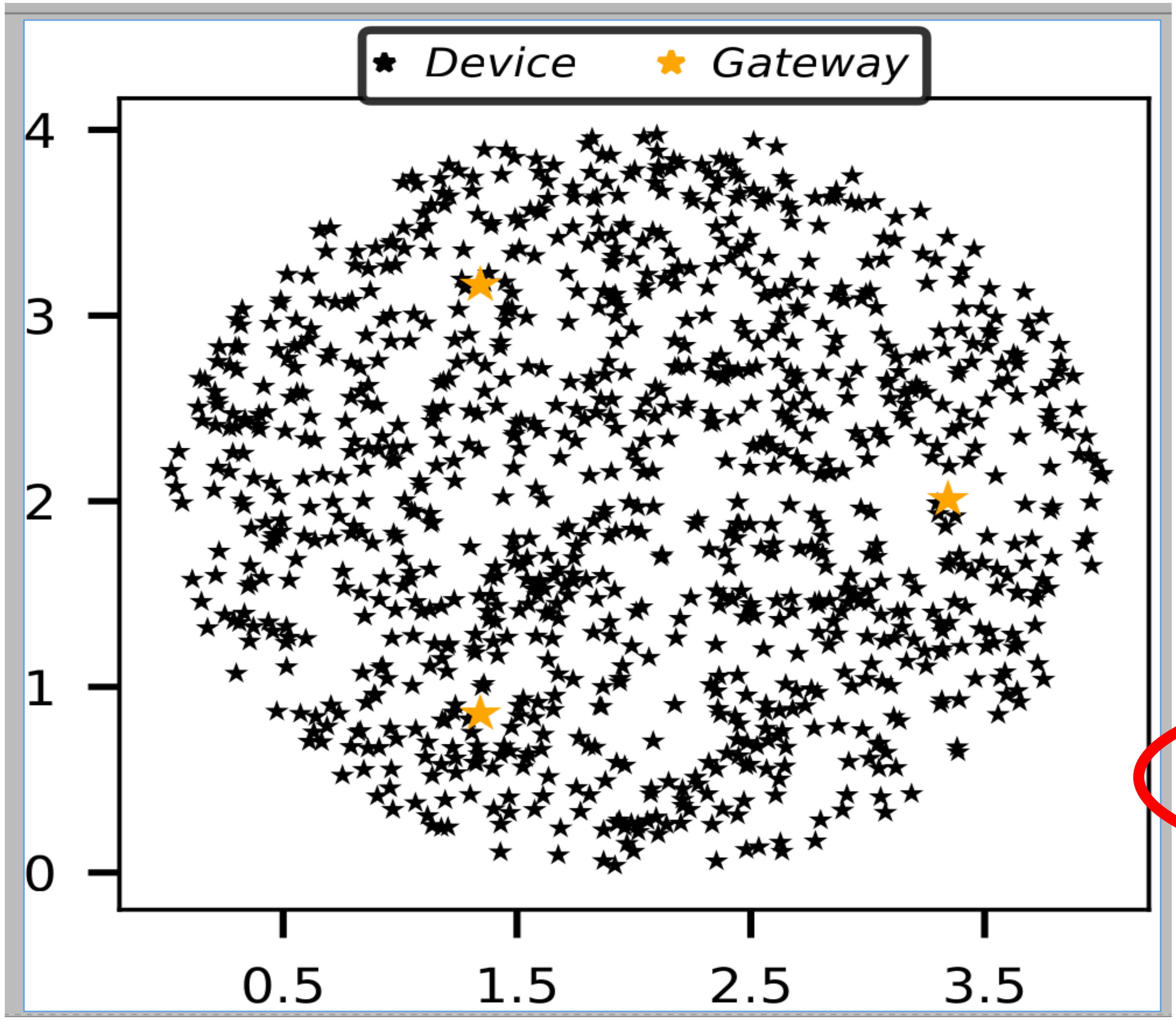
- CSMA for LoRaWAN, context
- LoRa-CSMA-Sim, our SimPy simulation framework
- LoRaWAN CSMA
  - LoRa Alliance Technical Report TR013 (Sept 2023)
- Improving LoRaWAN CSMA
- Conclusions/Perspective

Dense network = dense collision domain



- LPWAN for cities
- Around one gateway, one among…
- LoRa CSS still dominant in the unlicensed LPWAN

LoRa(WAN) in this dense collision domain

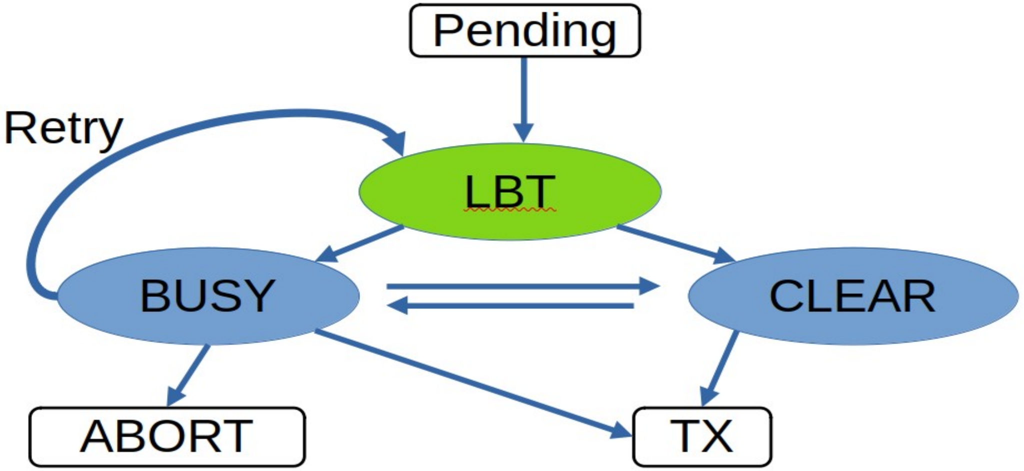


- Asynchronous
- No downlink needed
- SF12
- Interference
- Obstruction
- 8 channels * 125 kHz
- Random CH sequence/dev
- Multiple GWs

Carrier Sense Multiple Access (CSMA)

- Listen-Before-Talk
  - Is channel free?
- Dense LoRa => no feedback
- Actions a device can do:
  - "Listen"
  - "Wait" (Backoff, BO)
  - Change channel
  - Transmit
  - Discard

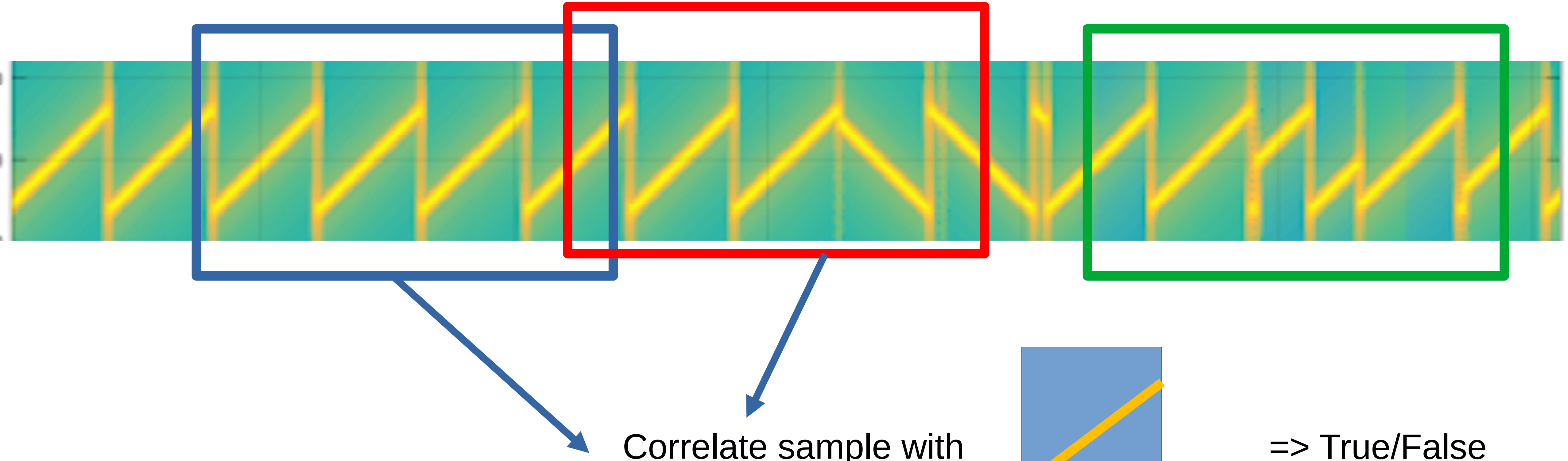
retry



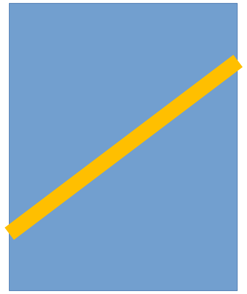
Listen in LoRa = Semtech's Channel Activity Detection (CAD)

- Not RSSI based
  - LoRa operates/interferes/collides below noise floor
  - No downlink
  - => based on symbol correlation
- Semtech AN1200.48 CAD performance Evaluation (2019)
  - Energy efficient (more than RX mode)
  - Reliable enough
  - Channel/SF specific

Channel Activity Detection (CAD)



Correlate sample with



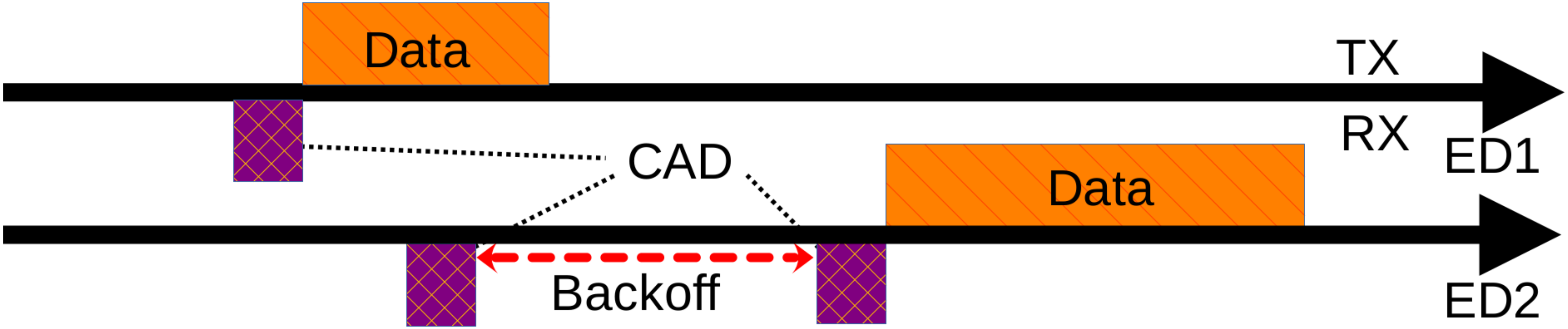
=> True/False

“Energy efficient” and fast

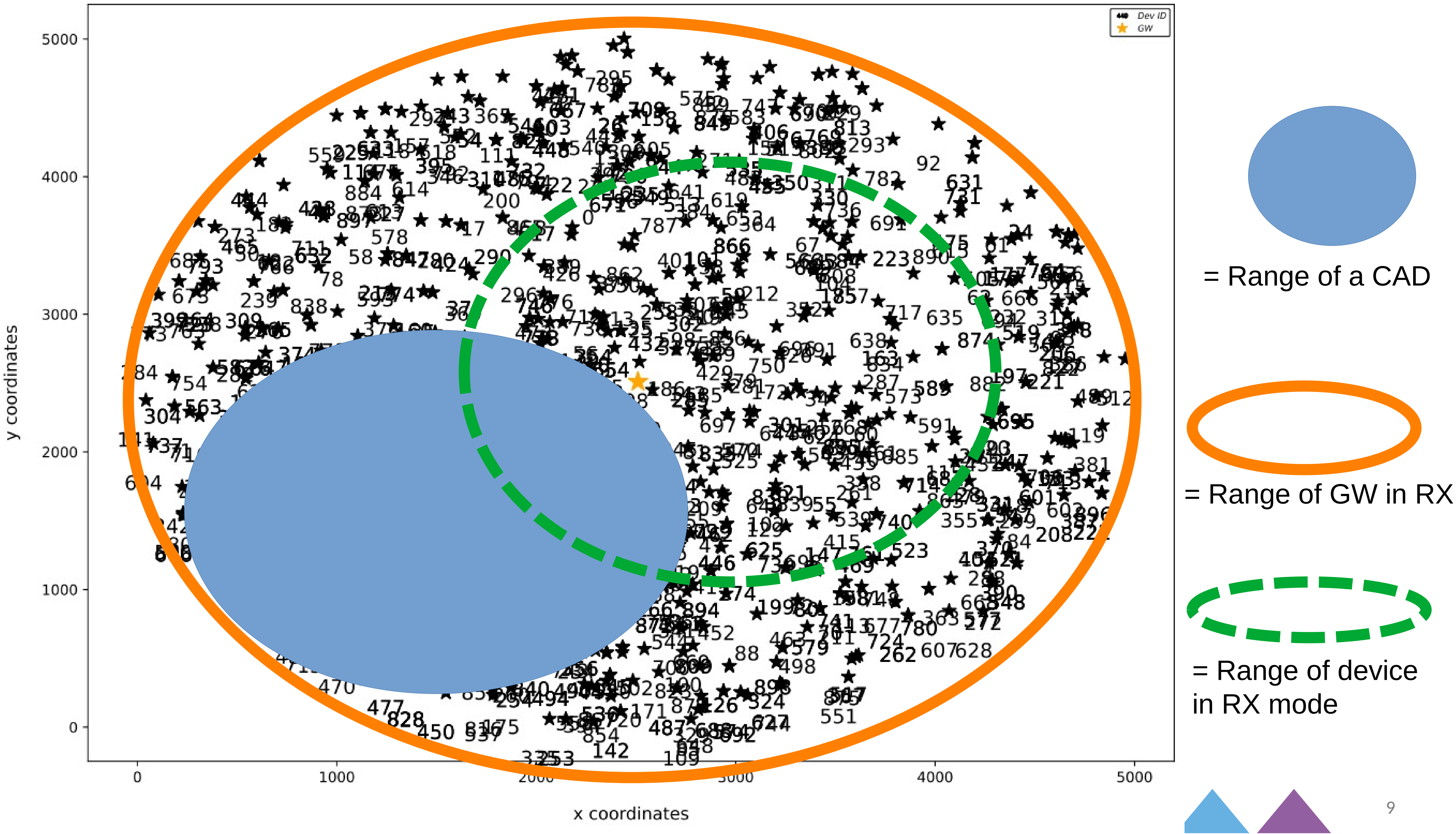
Slope-specific, asynchronous, not limited to preamble

Overlaps, down chirps, degraded signals?

Single Channel Collision Avoidance in LoRa: CAD and Backoff



CAD = *Channel Activity Detection*:
Low-energy and short channel sampling



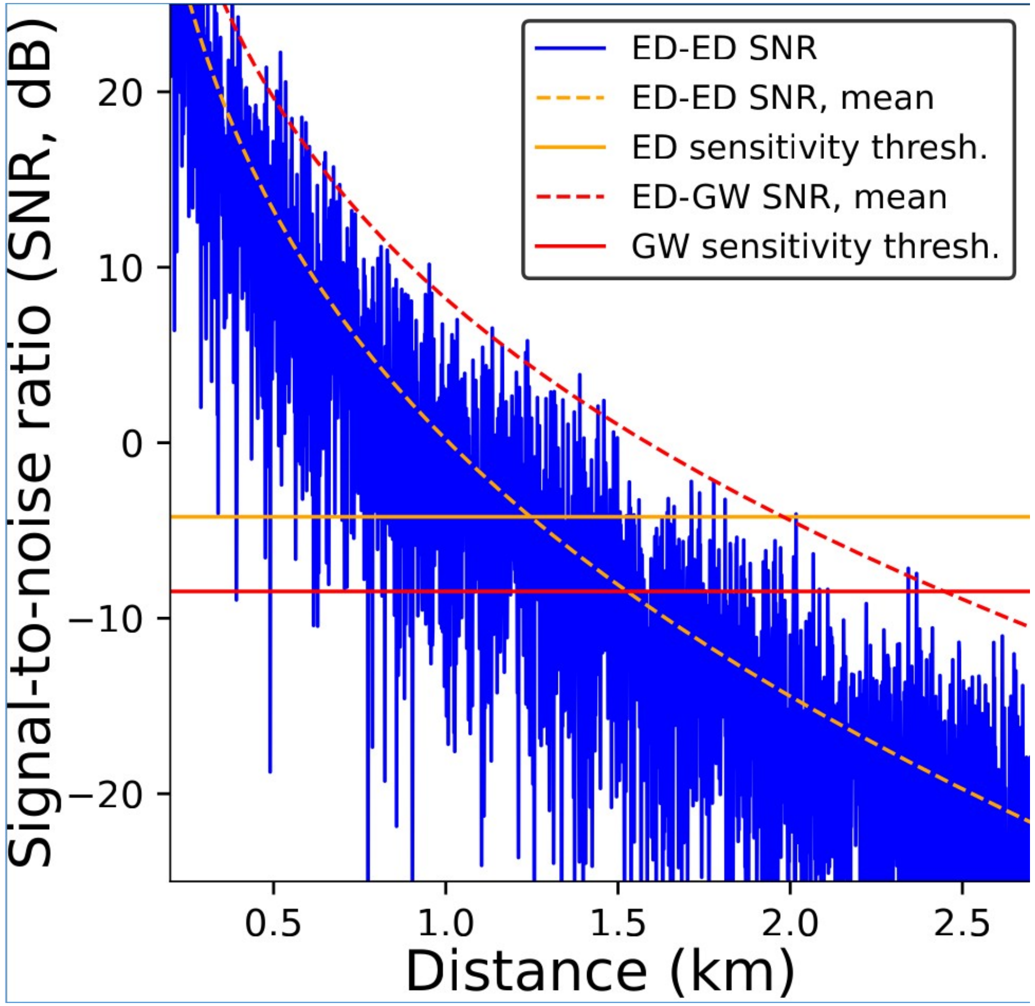
= Range of a CAD

= Range of GW in RX

= Range of device in RX mode

Path loss in simulation

SNR (dB) $= P_{TX}^{dB} + G - Lpld0 - \gamma 10 log_{10}(\frac{d}{d0}) - n - r$



- Gateways have better SNR than devices in RX mode
- Interference also impacts
- Devices fail RX after 1.3 km
- CAD follows a similar error rate

Simulation: LoRa-CSMA-Sim

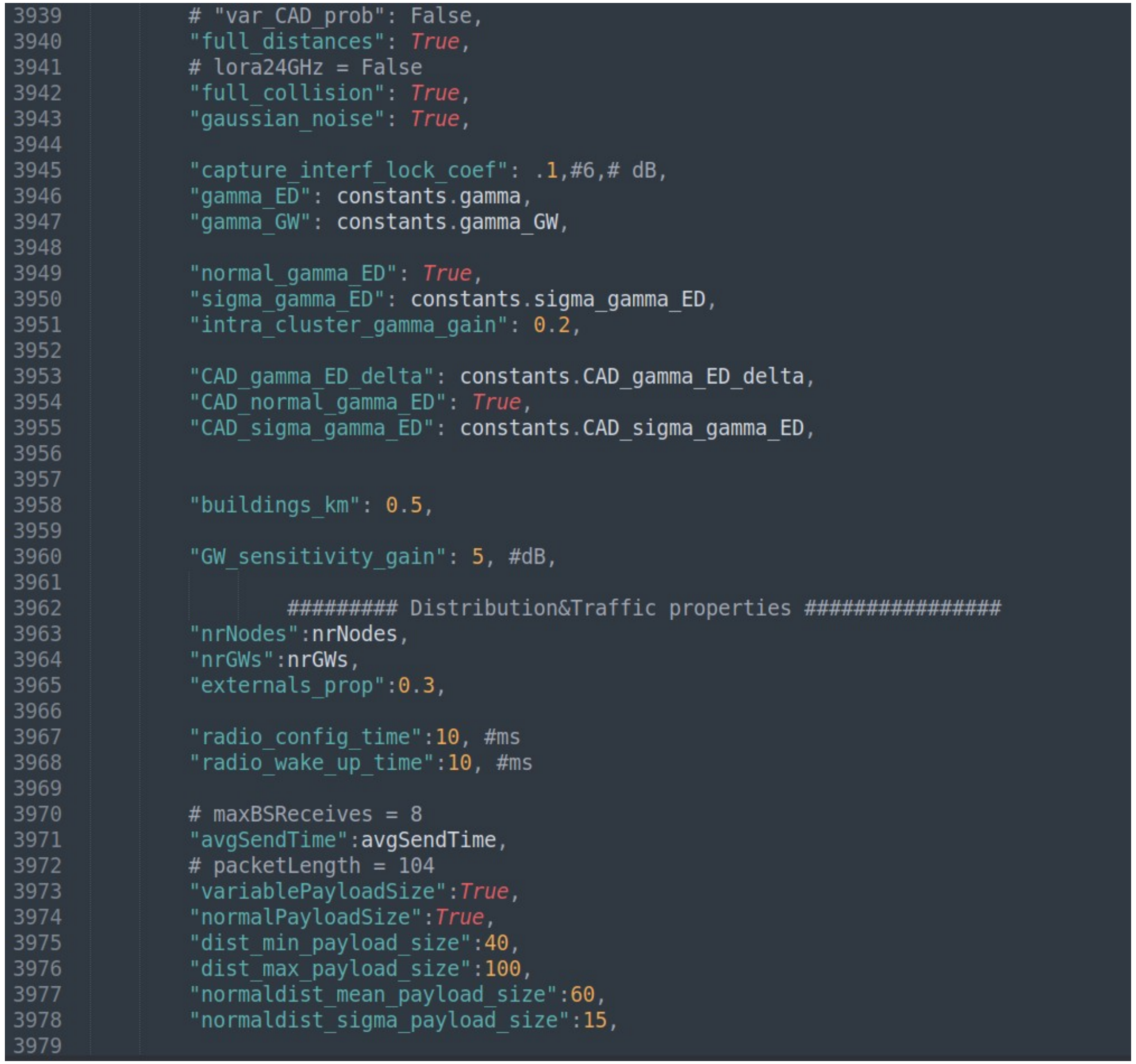
- Based on LoRaSim (Python) 2016 : Voigt, T., Bor, M. LoRaSim
- We improved with many advanced features & debug
  - Implements CAD, Backoff, RX, multi GW, multi CH → CSMA
  - Higher reproducibility (topology is generated separately)
  - Capture Effect with more than 2 transmitters
  - End-Device to End-Device communications (sensitivity, path-loss and collision models)
  - More accurate energy and channel model (noise, fast fading, interference)
  - Use of NumPy high performance numerical vectorized libraries → Large scale

Github: Guillaumegaillard/LoRa-CSMA-Sim

Simulation: LoRa-CSMA-Sim (2)

A fully parametrizable scenario

- Topology
- Traffic
- Physical layer parameters
- Device characteristics
- Protocols
- Duration, etc.

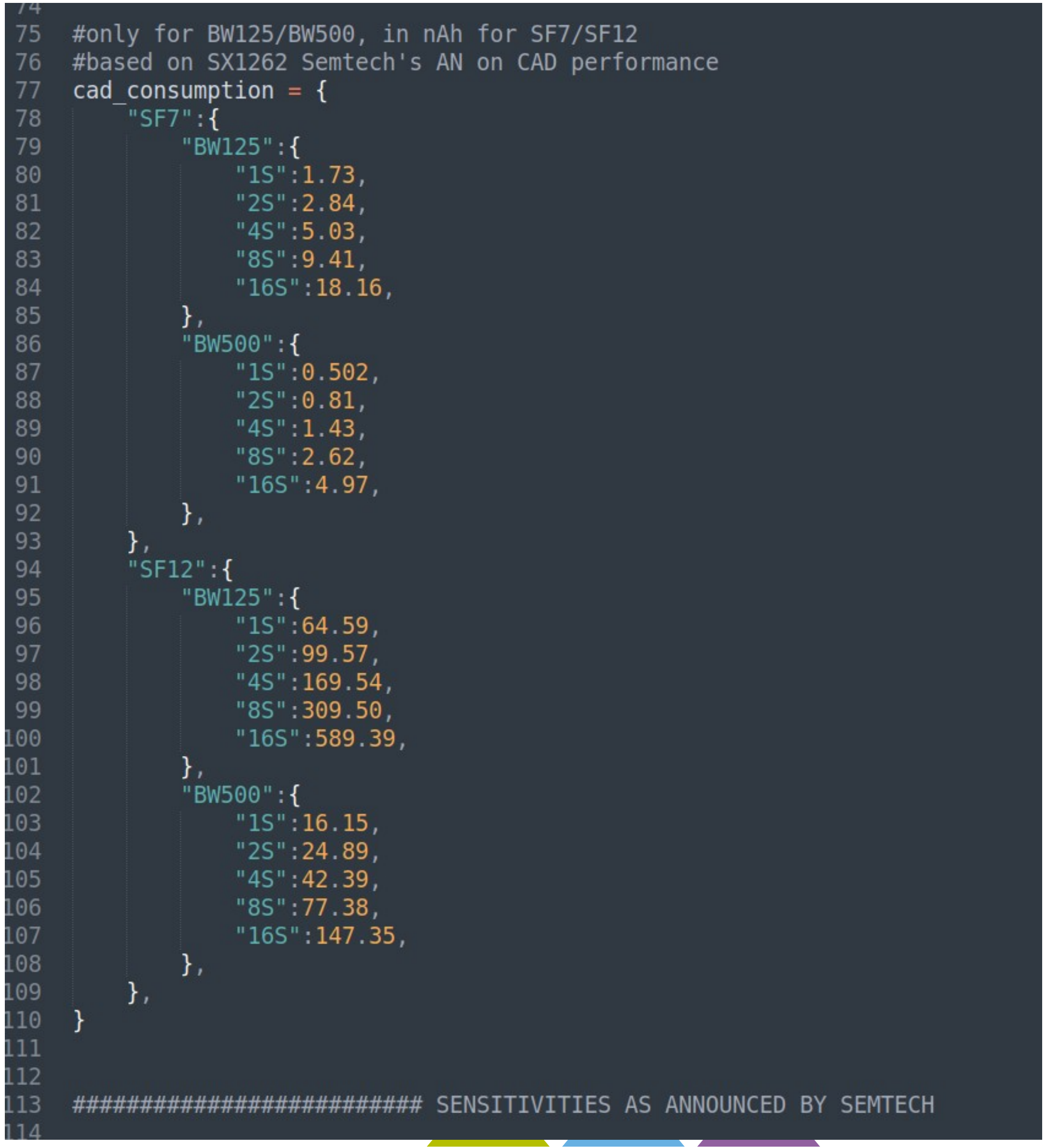


```
        # "var_CAD_prob": False,
        "full_distances": True,
        # lora24GHz = False
        "full_collision": True,
        "gaussian_noise": True,

        "capture_interf_lock_coef": .1,#6,# dB,
        "gamma_ED": constants.gamma,
        "gamma_GW": constants.gamma_GW,

        "normal_gamma_ED": True,
        "sigma_gamma_ED": constants.sigma_gamma_ED,
        "intra_cluster_gamma_gain": 0.2,

        "CAD_gamma_ED_delta": constants.CAD_gamma_ED_delta,
        "CAD_normal_gamma_ED": True,
        "CAD_sigma_gamma_ED": constants.CAD_sigma_gamma_ED,


        "buildings_km": 0.5,

        "GW_sensitivity_gain": 5, #dB,

                ######### Distribution&Traffic properties ################
        "nrNodes":nrNodes,
        "nrGWs":nrGWs,
        "externals_prop":0.3,

        "radio_config_time":10, #ms
        "radio_wake_up_time":10, #ms

        # maxBSReceives = 8
        "avgSendTime":avgSendTime,
        # packetLength = 104
        "variablePayloadSize":True,
        "normalPayloadSize":True,
        "dist_min_payload_size":40,
        "dist_max_payload_size":100,
        "normaldist_mean_payload_size":60,
        "normaldist_sigma_payload_size":15,

```

Simulation: LoRa-CSMA-Sim (2)

A fully parametrizable scenario

- Topology
- Traffic
- Physical layer parameters
- Device characteristics
- Protocols
- Duration, etc.

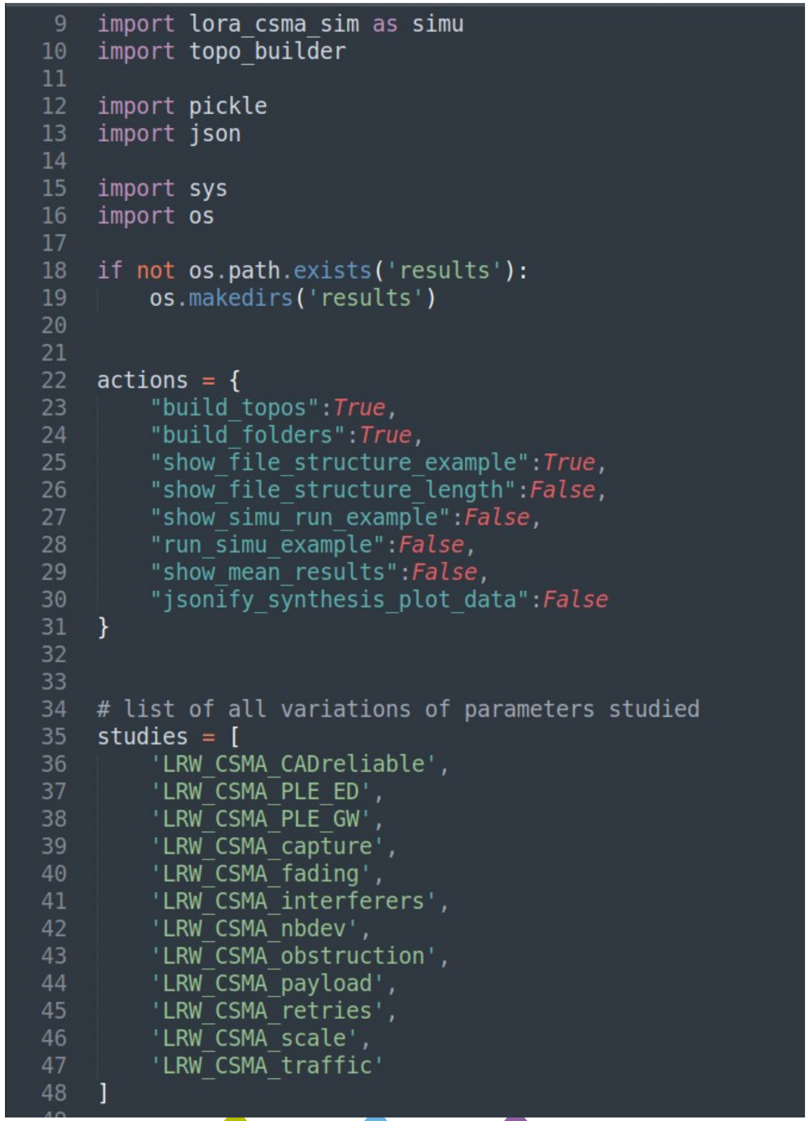


```
#only for BW125/BW500, in nAh for SF7/SF12
#based on SX1262 Semtech's AN on CAD performance
cad_consumption = {
    "SF7":{
        "BW125":{
            "1S":1.73,
            "2S":2.84,
            "4S":5.03,
            "8S":9.41,
            "16S":18.16,
        },
        "BW500":{
            "1S":0.502,
            "2S":0.81,
            "4S":1.43,
            "8S":2.62,
            "16S":4.97,
        },
    },
    "SF12":{
        "BW125":{
            "1S":64.59,
            "2S":99.57,
            "4S":169.54,
            "8S":309.50,
            "16S":589.39,
        },
        "BW500":{
            "1S":16.15,
            "2S":24.89,
            "4S":42.39,
            "8S":77.38,
            "16S":147.35,
        },
    },
}


######################### SENSITIVITIES AS ANNOUNCED BY SEMTECH
```

Simulation: LoRa-CSMA-Sim (3)

Independent protocol comparison studies varying a parameter, repeated over multiple topology instances

- Traffic
- Scale
- Payload sizes
- Path loss
- etc.

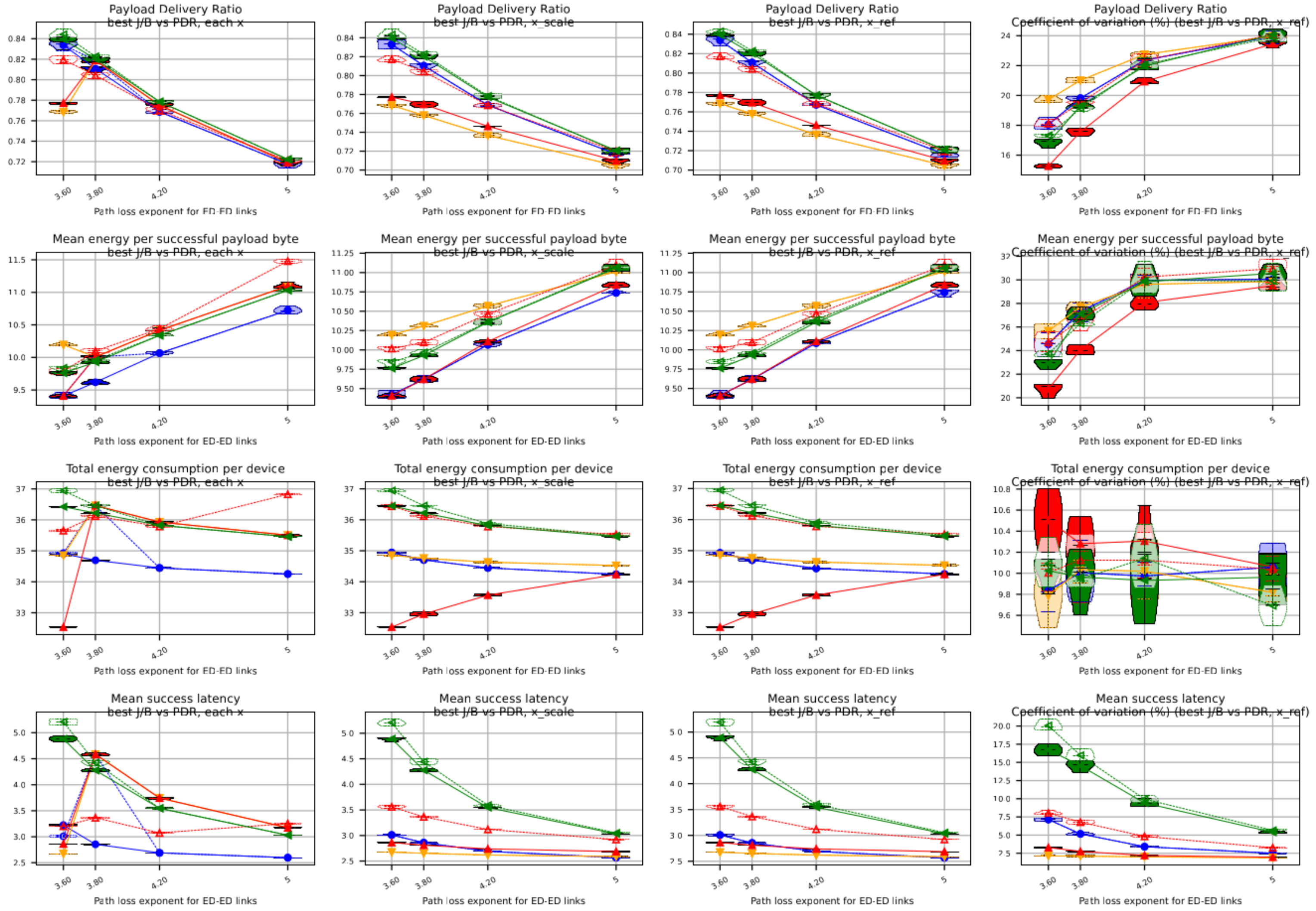


```python
import lora_csma_sim as simu
import topo_builder

import pickle
import json

import sys
import os

if not os.path.exists('results'):
    os.makedirs('results')


actions = {
    "build_topos":True,
    "build_folders":True,
    "show_file_structure_example":True,
    "show_file_structure_length":False,
    "show_simu_run_example":False,
    "run_simu_example":False,
    "show_mean_results":False,
    "jsonify_synthesis_plot_data":False
}


# list of all variations of parameters studied
studies = [
    'LRW_CSMA_CADreliable',
    'LRW_CSMA_PLE_ED',
    'LRW_CSMA_PLE_GW',
    'LRW_CSMA_capture',
    'LRW_CSMA_fading',
    'LRW_CSMA_interferers',
    'LRW_CSMA_nbdev',
    'LRW_CSMA_obstruction',
    'LRW_CSMA_payload',
    'LRW_CSMA_retries',
    'LRW_CSMA_scale',
    'LRW_CSMA_traffic'
]
```

Simulation: LoRa-CSMA-Sim (4)

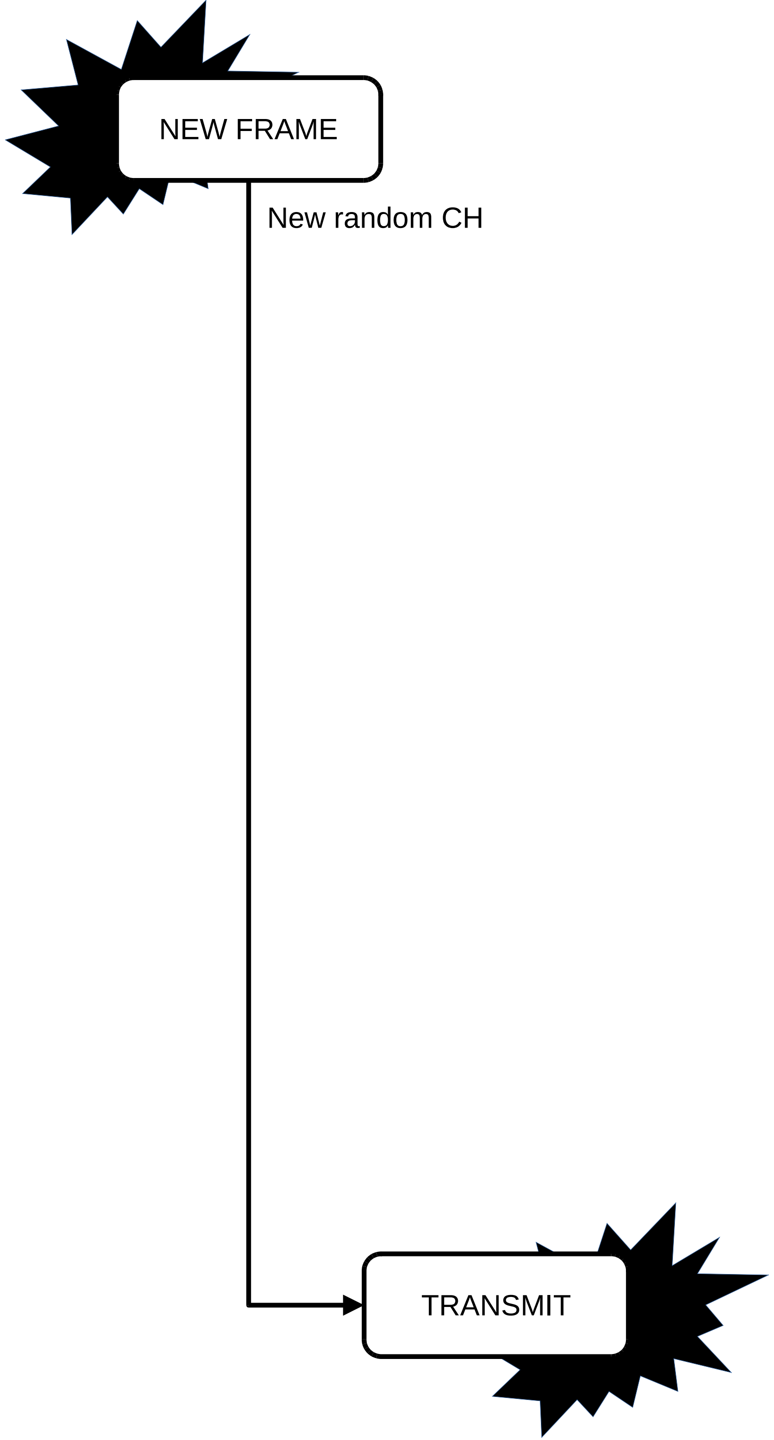
Outputs ~40 metrics

- PDR
- Collisions
- End-to-end Delays
- Energy consumption
- Channel occupation
- Etc.



Let’s compare existing/naive CSMA approaches

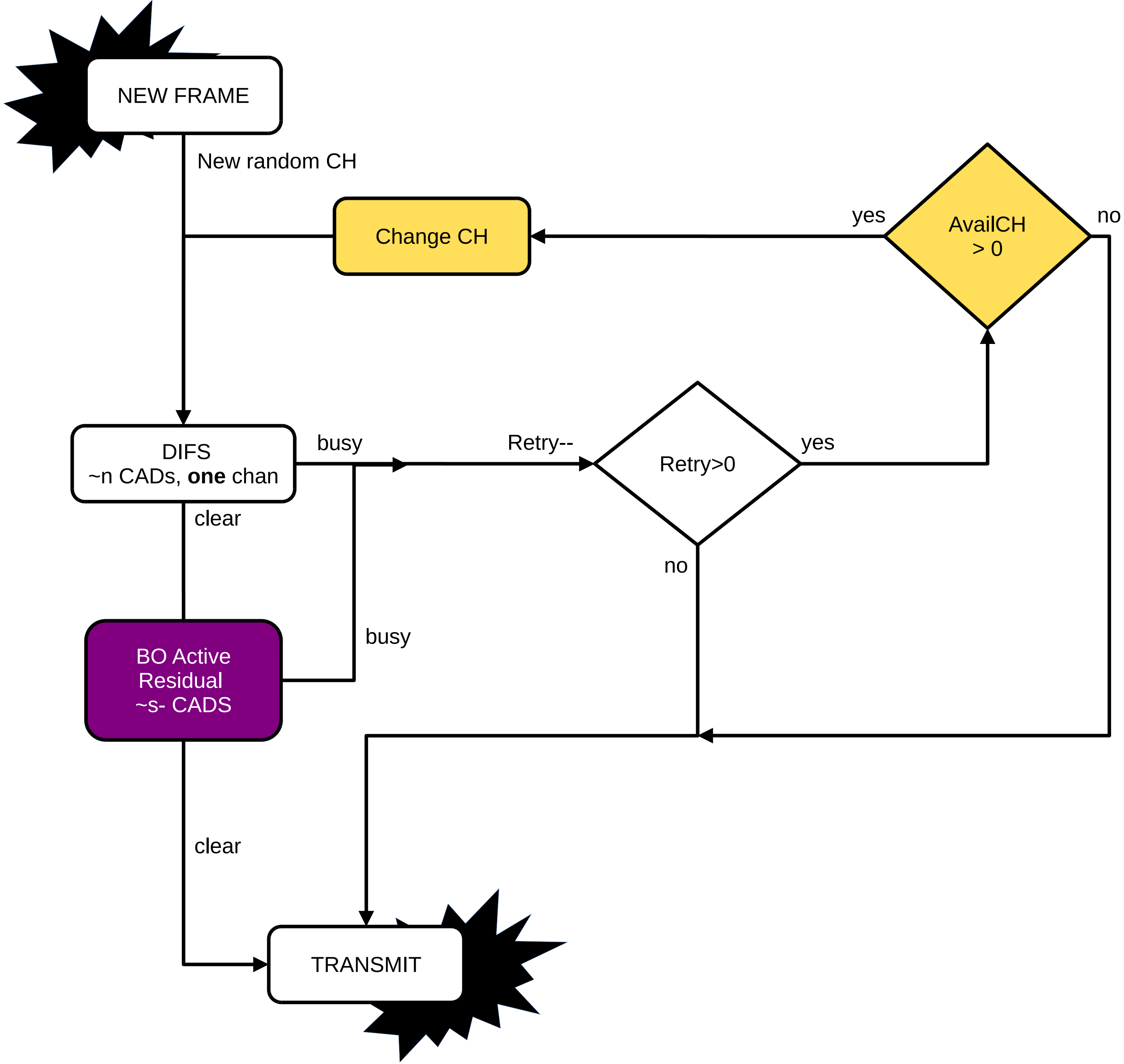
- LoRa
  - ALOHA
- LoRaWAN CSMA (LoRa Alliance)
  - Change Channel + residual backoff
- Basic LoRa CSMA
  - naive multi-CH LBT + exponential backoff



ALOHA

LoRaWAN CSMA

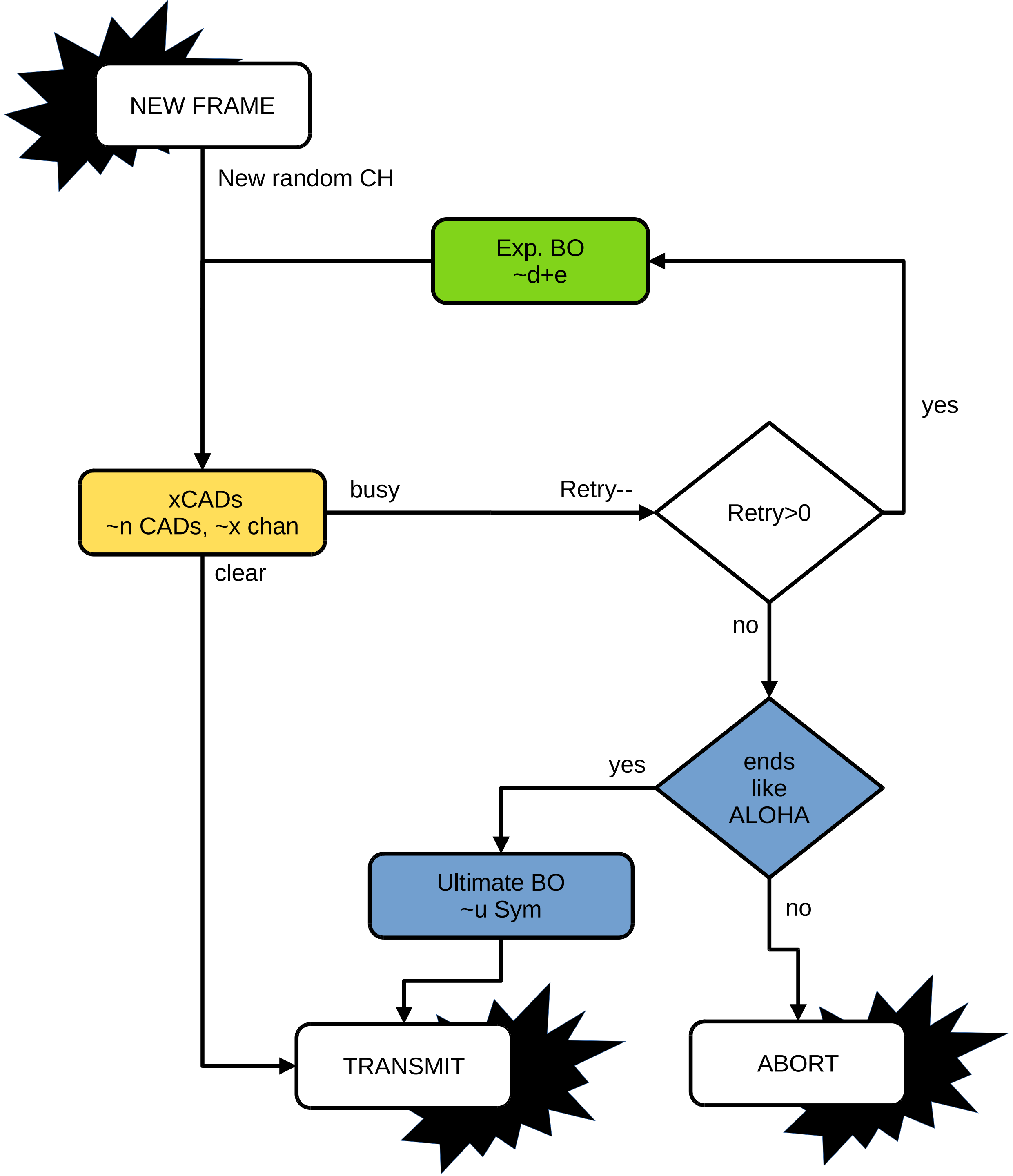
- Introduces a DIFS
  - Sequence of n CADs on the same channel
- Introduces an active residual backoff when channel is CLEAR
  - Reduced duration at each retry
  - Sequence of s CADs
- Change channel when BUSY



LoRaWAN CSMA

Basic LoRa CSMA

- Inspired from CSMA in other wireless (e.g. IEEE 802.15.4)
- Introduces a xCADs as LBT
  - Sequence of n CADs on a channel
  - Repeated over x channels until a CLEAR one is found
- If all (8) channels BUSY
  - Binary Exponential Backoff
    - Increased duration at each retry
- A softer termination



Basic LoRa CSMA

Simulation: LoRa-CSMA-Sim (5)

- Independent simulation runs
- Protocols compared for 64 variants each:
  - BO sizes
  - CAD sequences
  - inter CAD intervals

For this whole simulation:

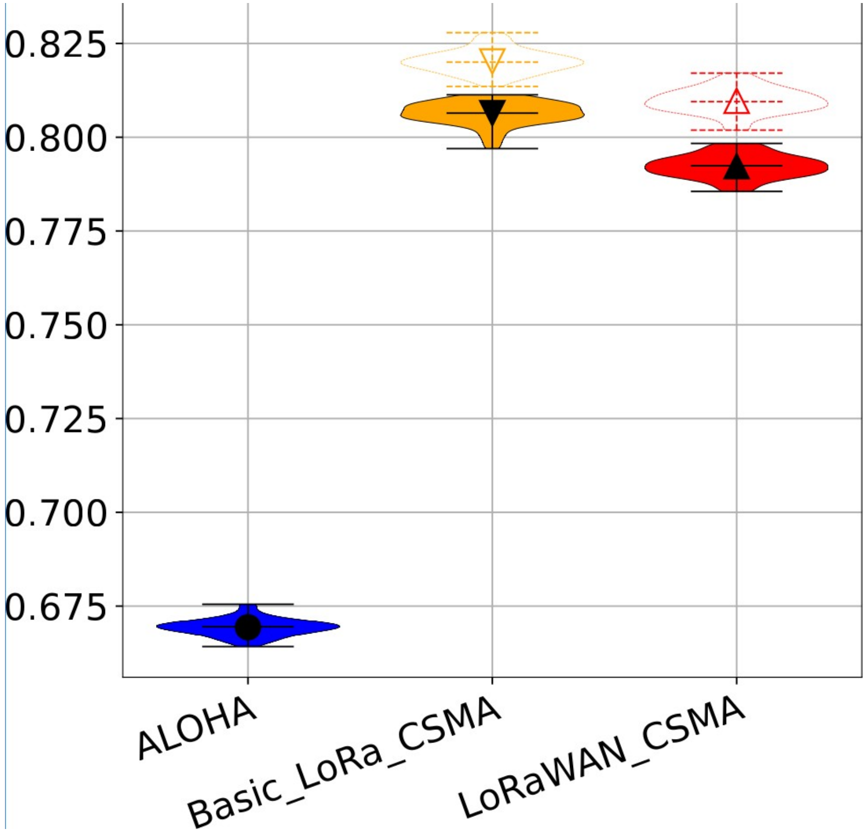
61632 simulation runs
6.29E9 frame generations.

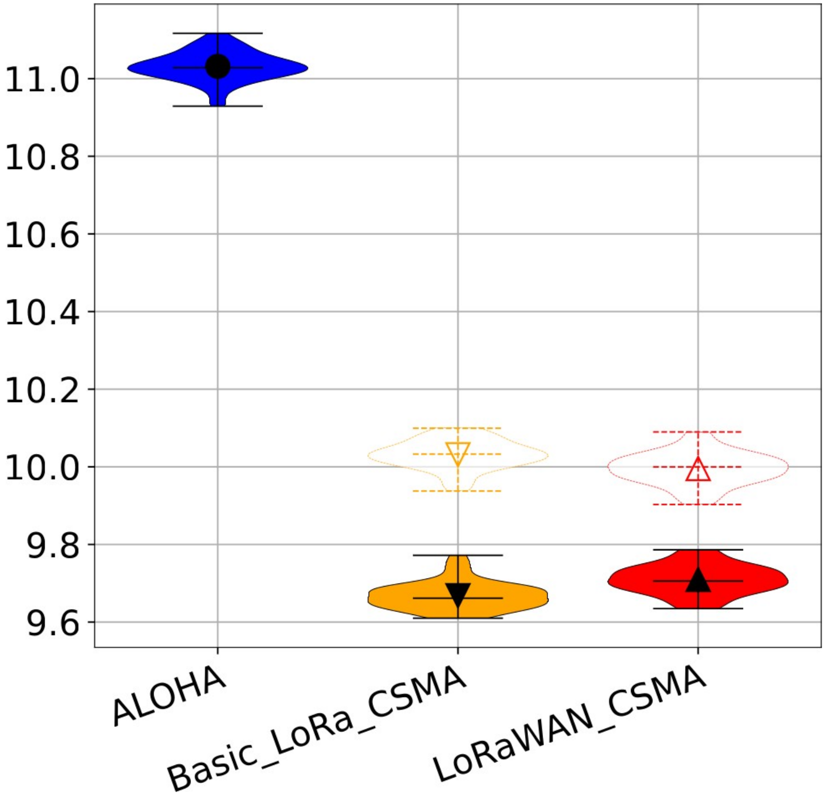
6h30 on HPC cluster Pyrene
(UPPA-E2S, Pau, France)
~43 days of CPU-time

SimPy event based
- frame components

NumPy C-based vectorization

Results (1): ALOHA is outperformed



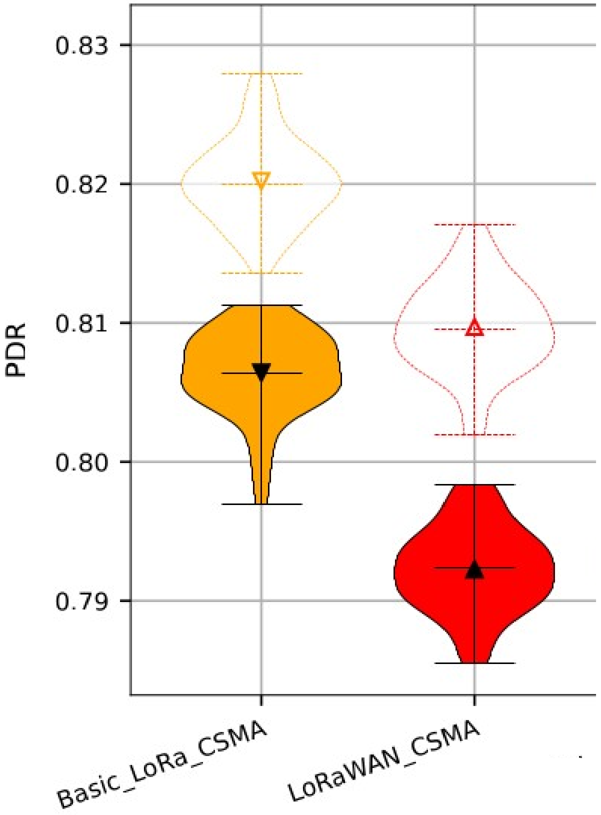


PDR

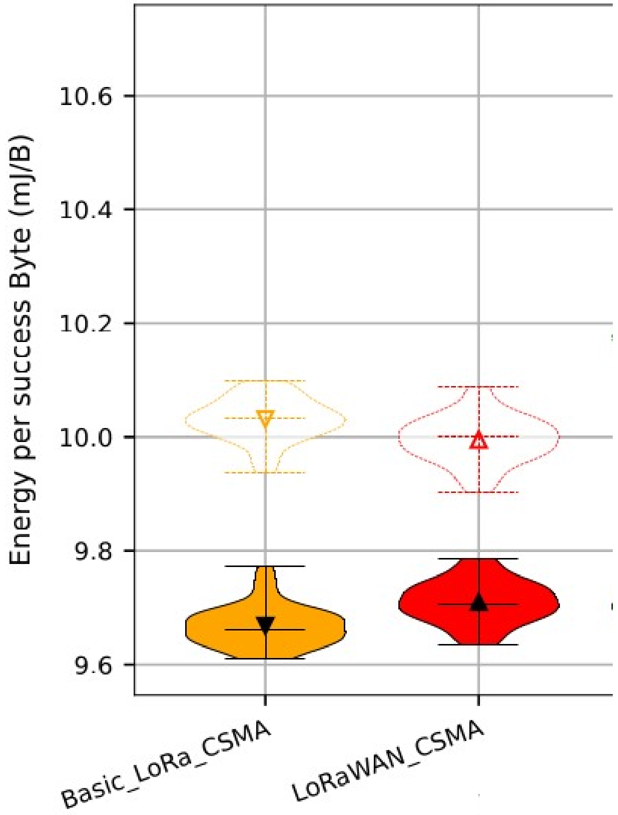
Energy consumed per successful payload byte

mJ/B

Results (2): Basic better than LoRa alliance!!



PDR



Energy consumed per successful payload byte

mJ/B

Three improved proposals with:

1

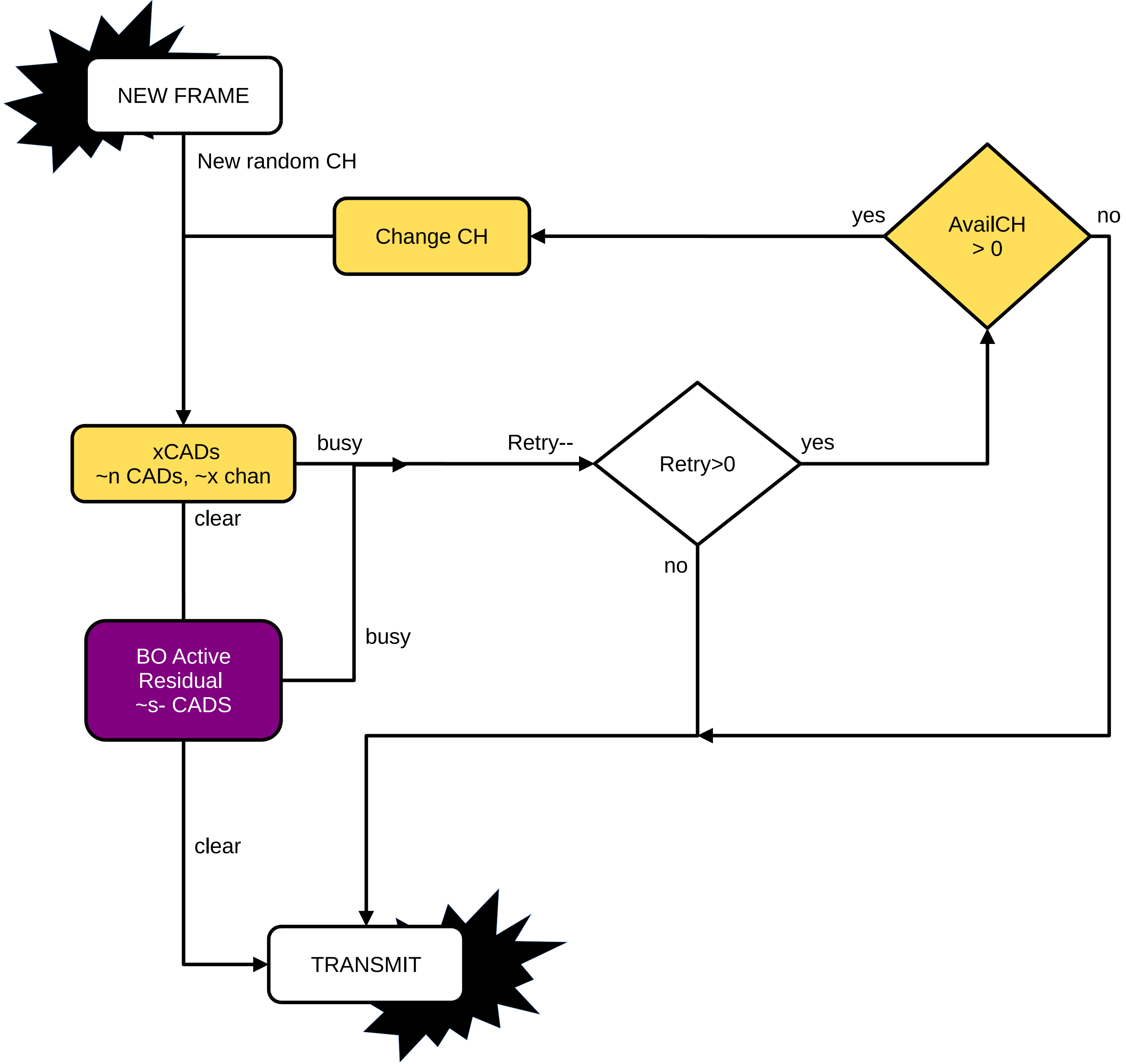
- Add the multi-channel CAD-based LBT as in “Basic”

2

- A passive Backoff (sleep) when channel is CLEAR
- A softer termination as in “Basic”

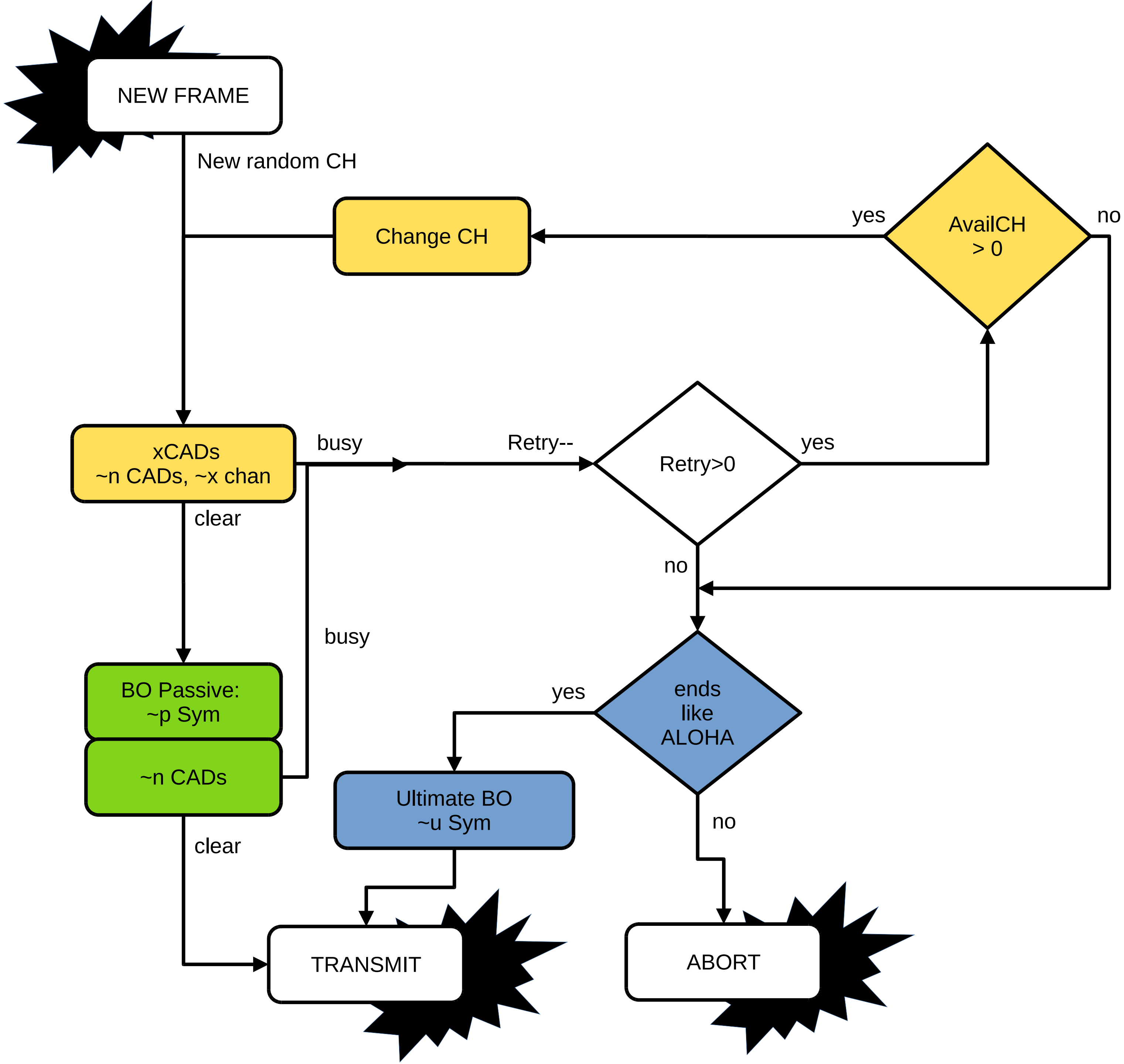
3

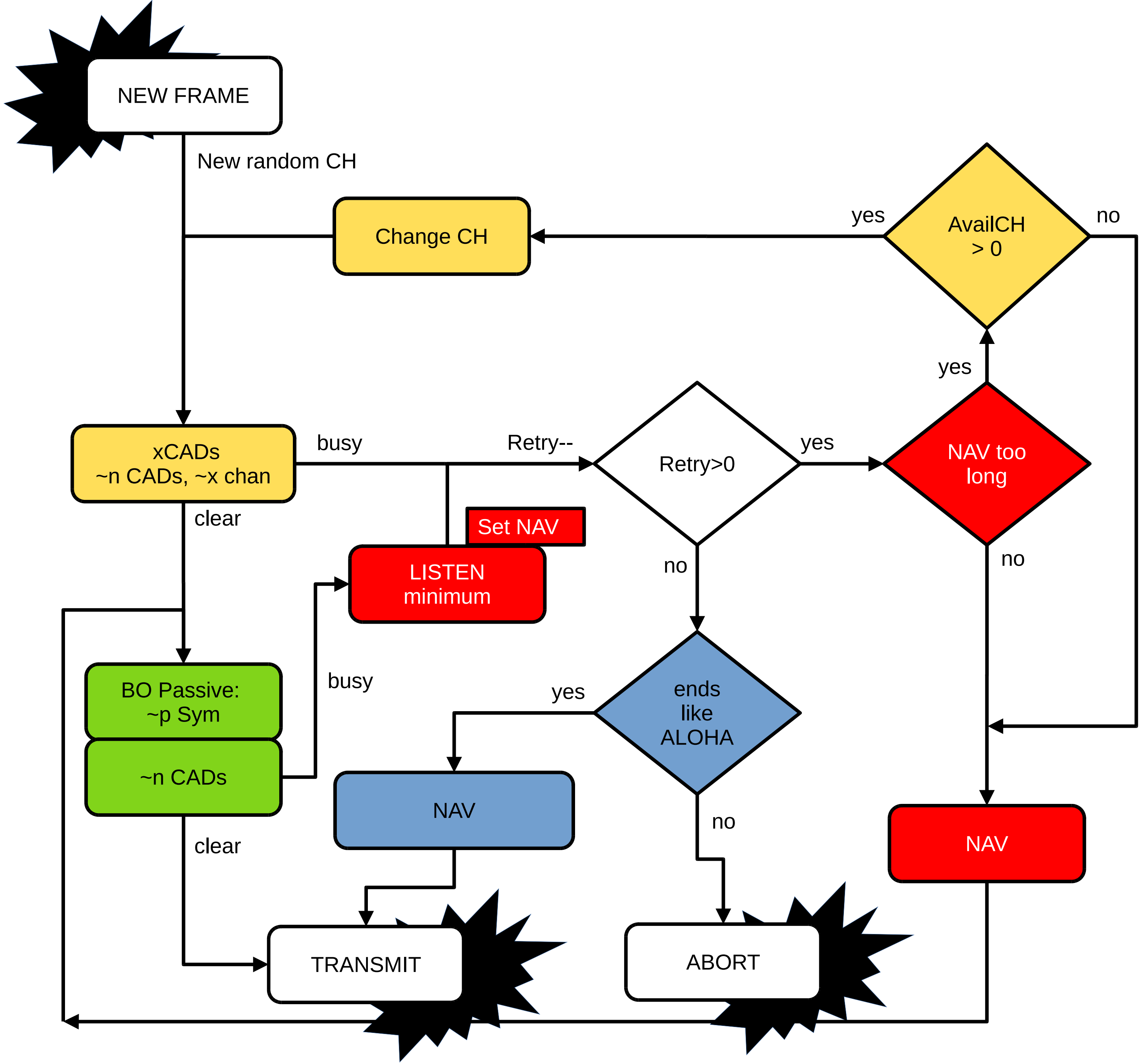
- Obtain frame’s Time on Air
  - using RX mode on preambles
- Add threshold for “wait / change CH” decision when Busy



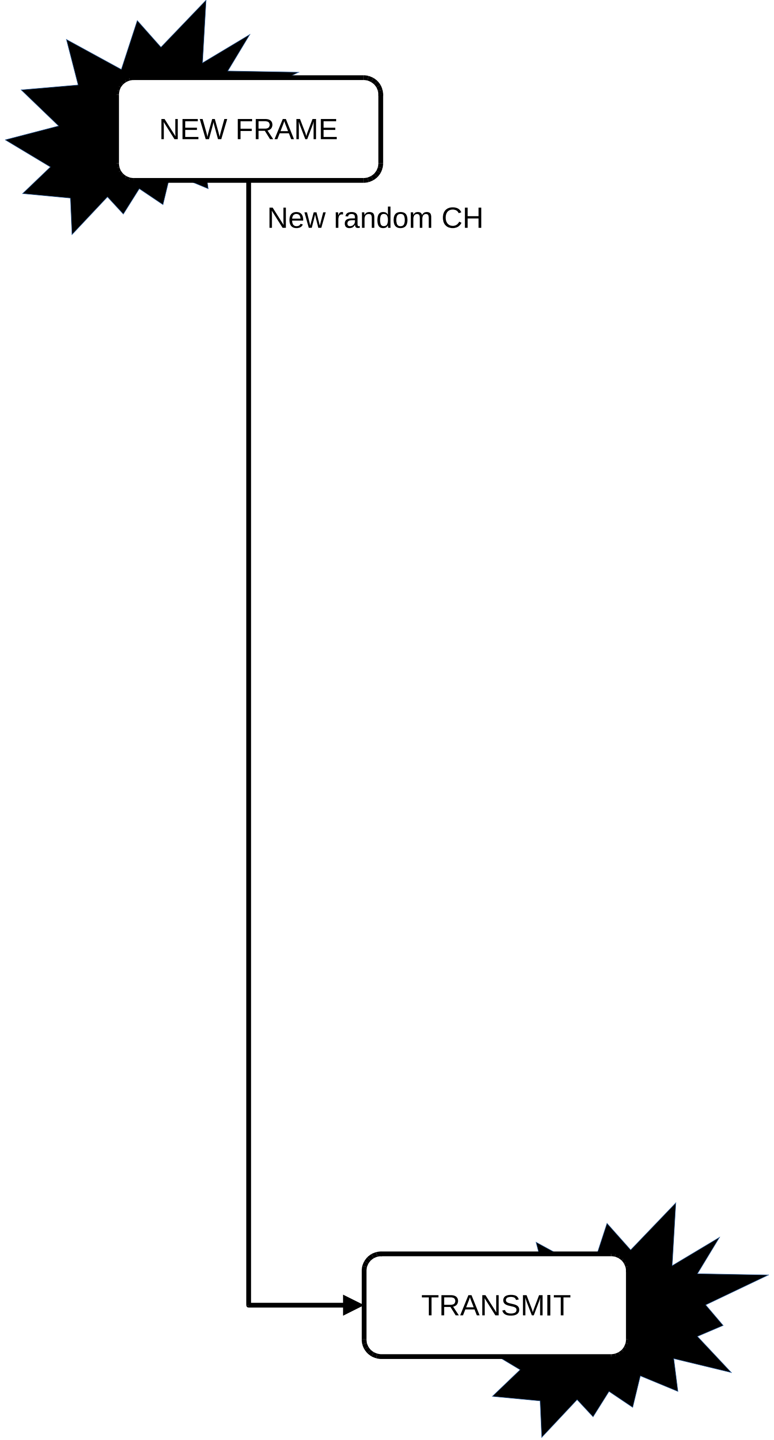
LoRaWAN CSMA 1

LoRaWAN CSMA 2



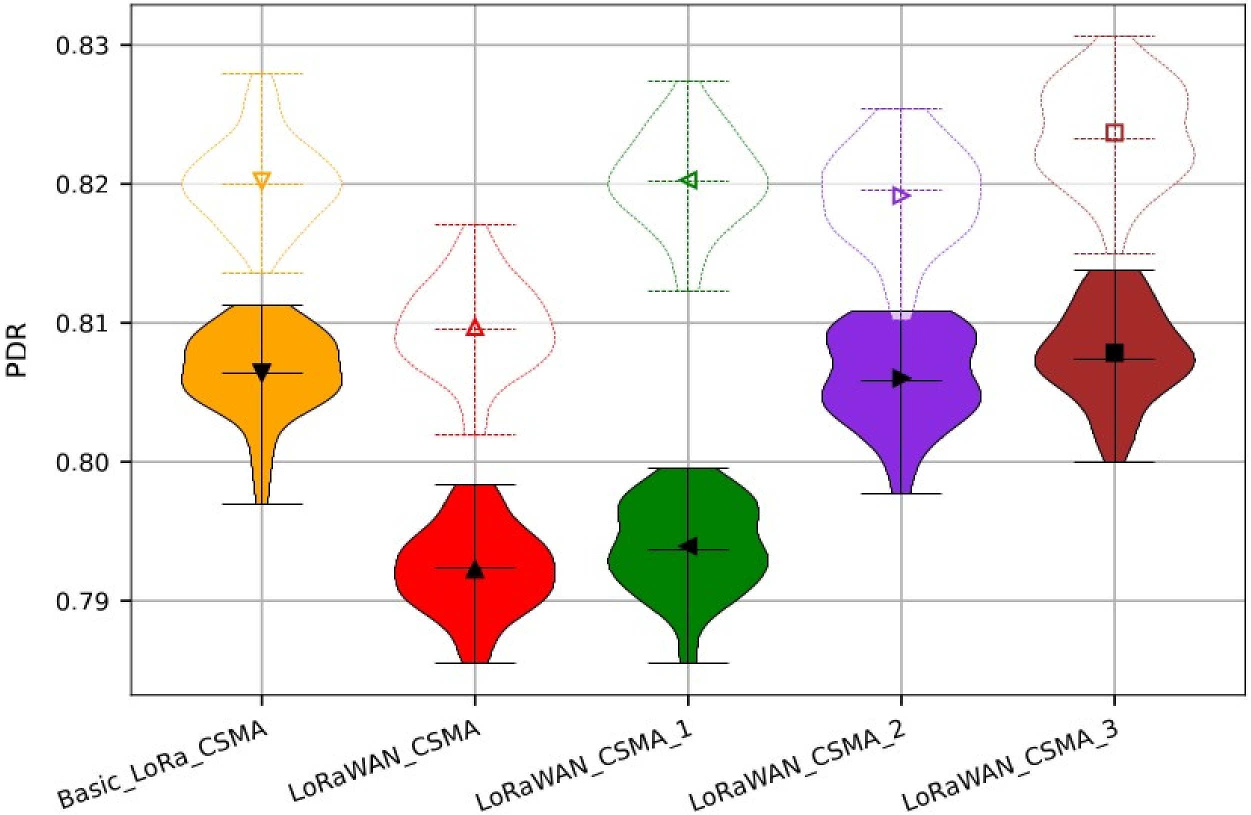


LoRaWAN CSMA 3

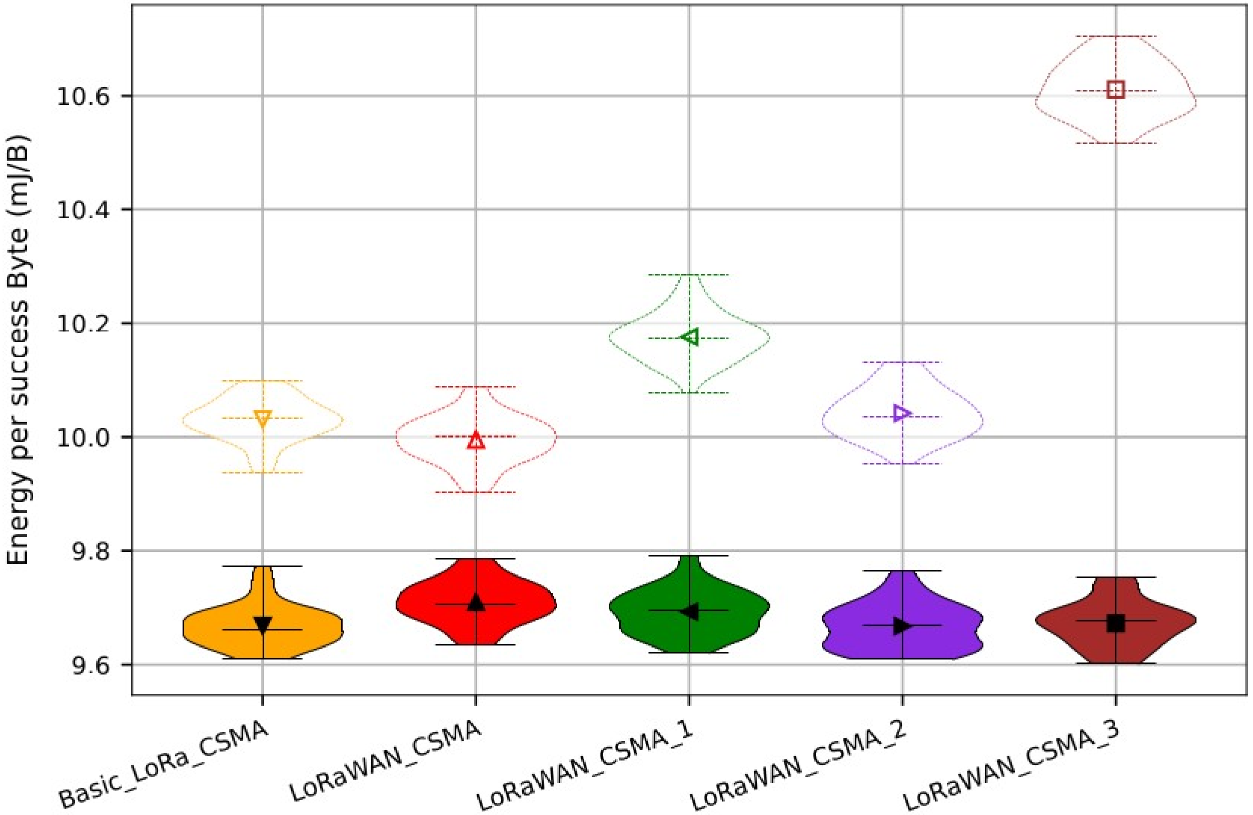


ALOHA

Results (3): Improvement 3 better than Basic



PDR



Energy consumed per successful payload byte

mJ/B

Conclusions, Perspective

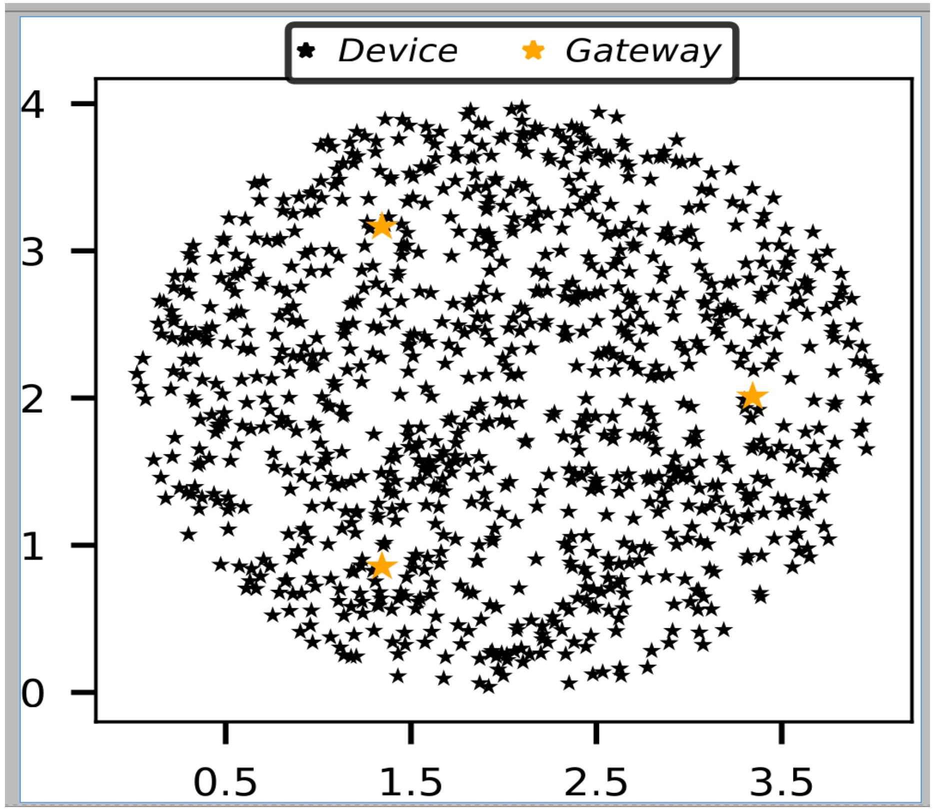
- A “basic” multi-channel LBT outperforms LoRa alliance CSMA (2023) in terms of PDR and energy efficiency in dense LoRa networks.
- The improvements we propose permit to reach the best PDRs with increased energy consumption, but also to improve PDRs while keeping a reasonable energy efficiency.
- Protocol choice is scenario-dependent (“it depends”!)
- LoRa-CSMA-Sim (github) allows fine-grained protocol simulations and results and is available as an open source modular code

Perspective

- Large-scale experimental validation
- Analytical model (markov)
- Machine-learn the best protocol variant
- Compare (in coexistence) with other LPWAN
  - LR-FHSS, Mioty, etc.

Questions ?

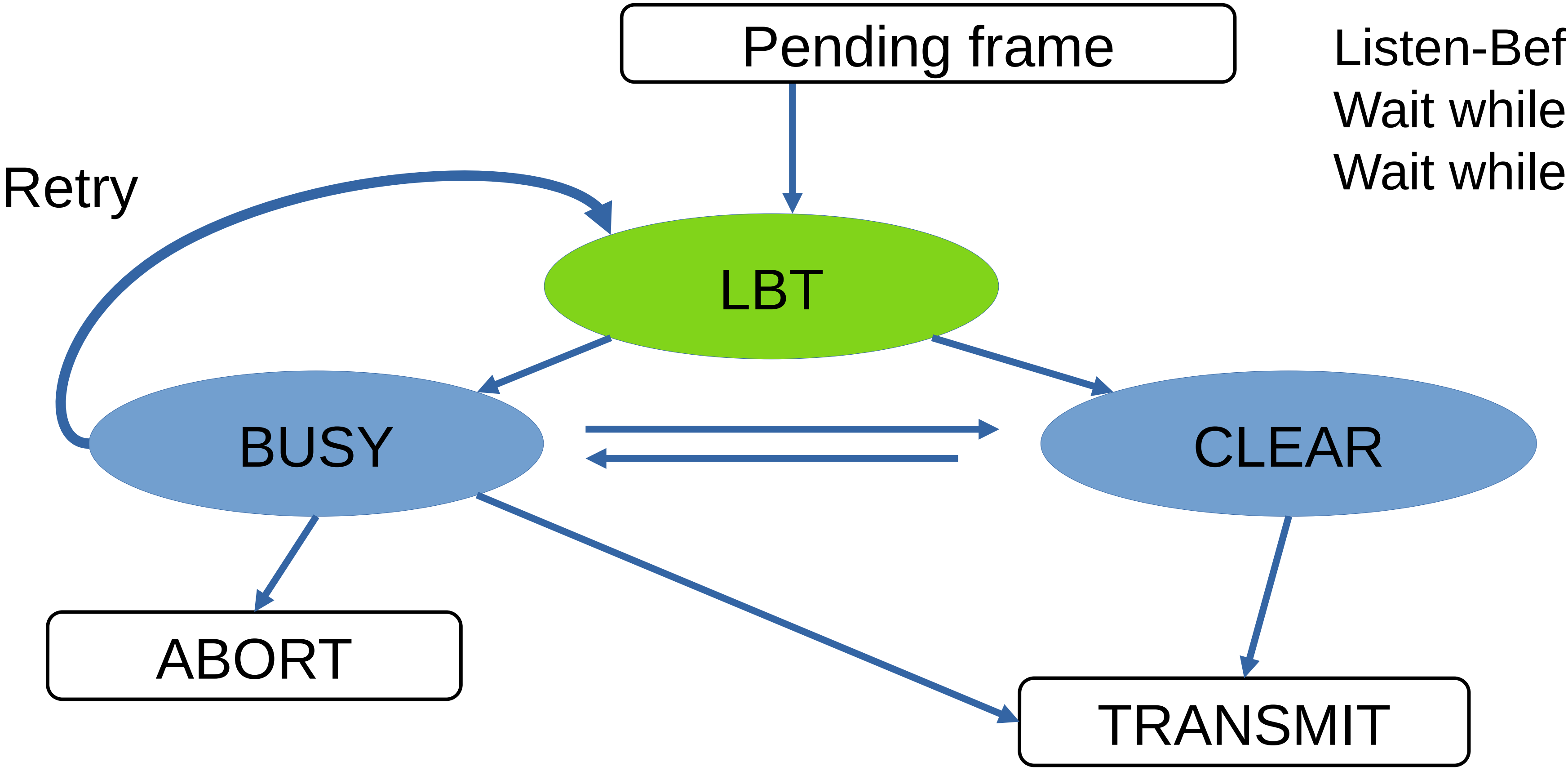
- Thanks :)



Guillaume Gaillard, Congduc Pham

@univ-pau.fr

State machine of CSMA



Listen-Before-Talk
Wait while Busy
Wait while Clear

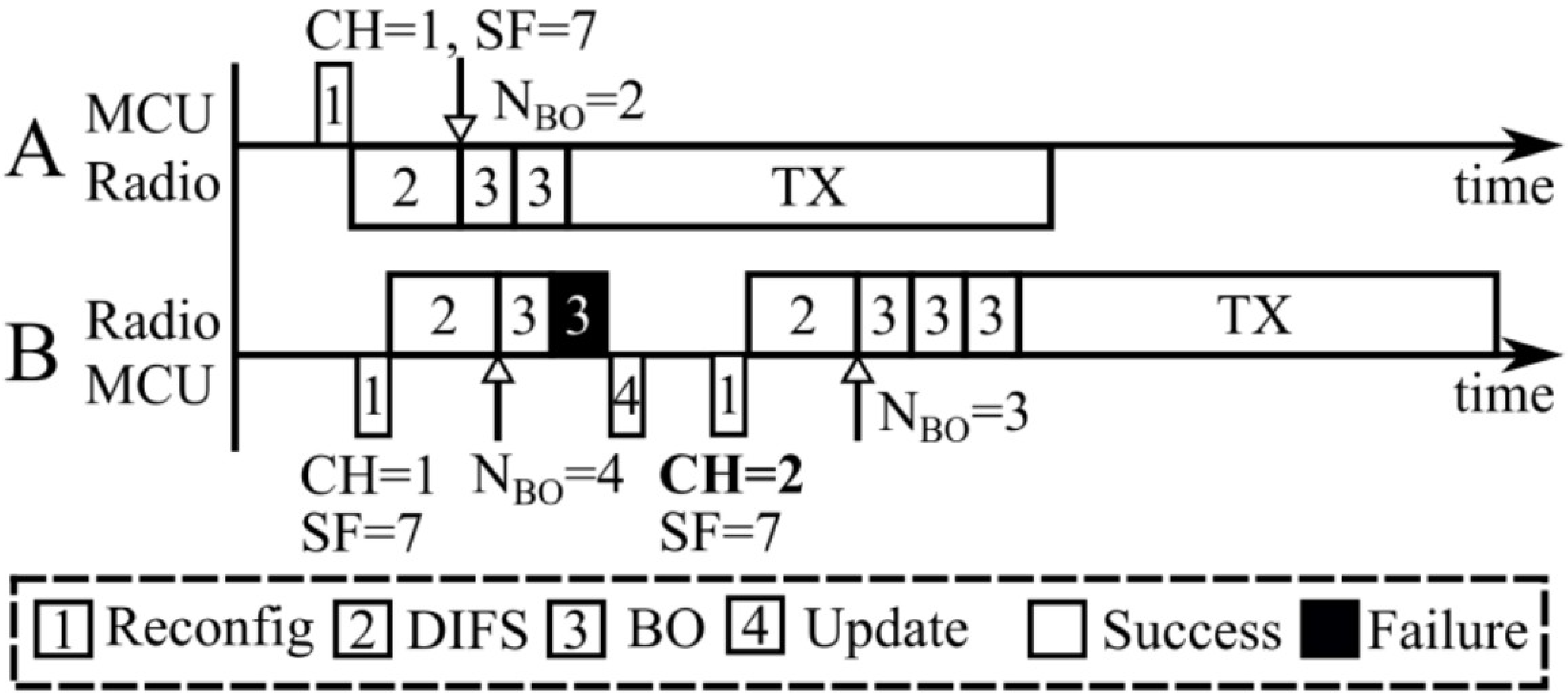
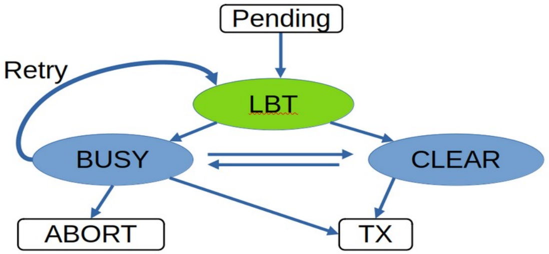


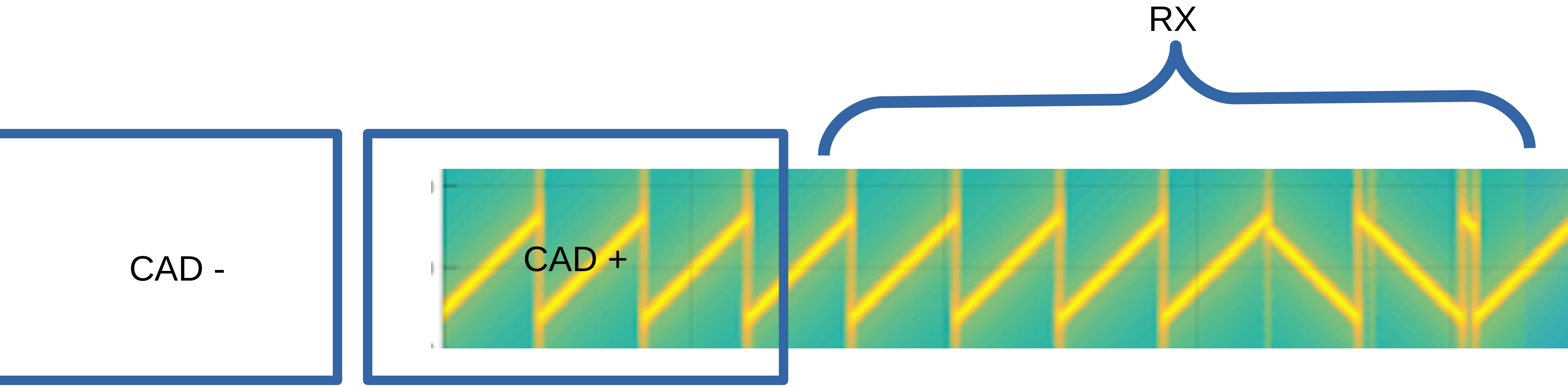
Figure 2: Two nodes contending for the same CH/SF combination

LoRaWAN CSMA

Proposal: CAD + RX


Pending
Retry
LBT
BUSY
CLEAR
ABORT
TX

- RX only when a CAD switches Busy
  - Target preamble only
- Get NAV length


RX

CAD -

CAD +