Experimenting IoT in the CALIPSO Project

Ing. Pietro Gonizzi
Department of Information Engineering
University of Parma, Italy

Outline

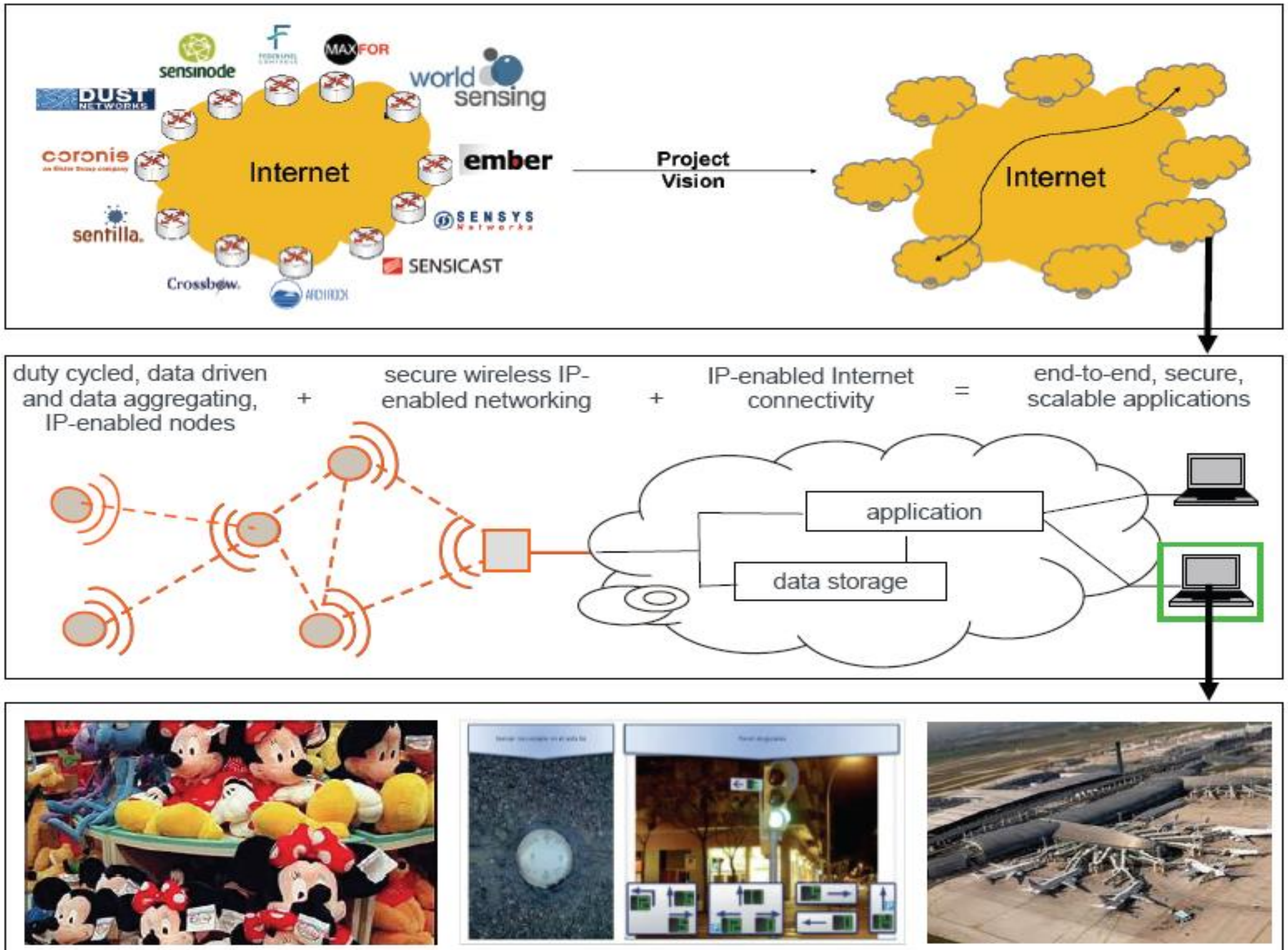
- Project overview
- IoT architecture and use cases
- Research challenges
- Dissemination and standardization
- Case studies: Distributed data storage, RAWMAC
- Calipso experimental evaluation
- Conclusions

Project overview

- FP7 Connect All IP-based Smart Objects! (3 years – Sep.2011-Aug.2014)
- Partners
  - Thales Communications and Security (J.Leguay, P.Medagliani), CNRS (A.Duda), University of Parma (G.Ferrari, P.Gonizzi, S.Cirani, L.Veltri), SICS (S.Duquennoy), Worldsensing (M.Monton, J.Pacho), Disney Research (S.Mangold), Cisco Systems (O.Dupont)
- Internet Protocol (IP) connected smart object networks, but with novel methods to attain very low power consumption
- IETF/IPv6 framework (6LoWPAN, RPL, CoAP)
- Three applications: Critical Infrastructures, Smart Parking, and Smart Toys
- Platform for developments: Contiki OS

www.ict-calipso.eu

Project vision



Calipso objectives

- 1. Design of the architecture for the Internet of Smart Objects, based on the requirements of use cases
- 2. Optimization and enhancement of IP interconnection over duty-cycled nodes and lossy links
- 3. Design and development of a support for applications and end-to-end communications
- 4. Evaluation and experimental validation on testbeds with target applications

Outline

- Project overview
- IoT architecture and use cases
- Research challenges
- Dissemination and standardization
- Case studies: Distributed data storage, RAWMAC
- Calipso experimental evaluation
- Conclusions

Smart Toys application

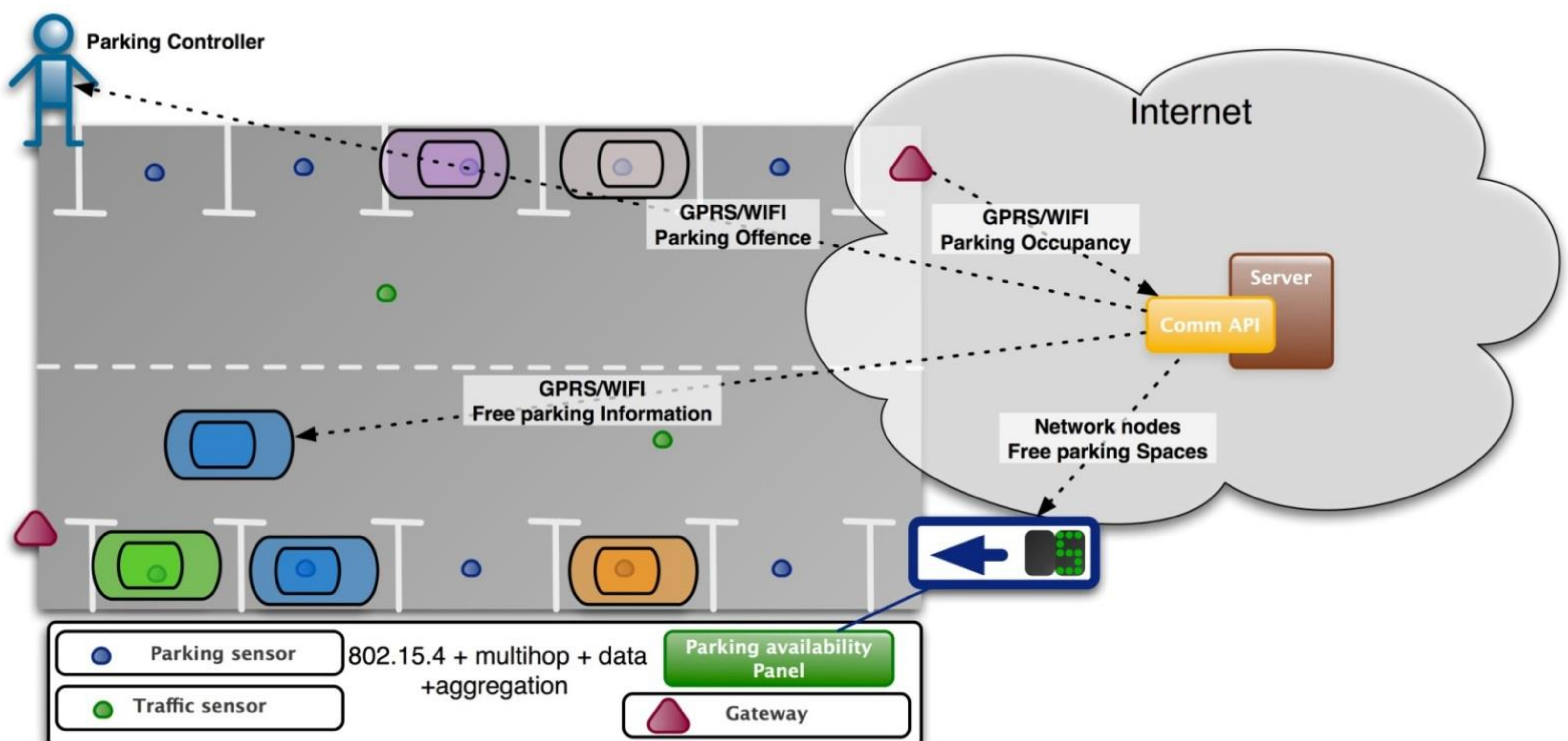
- Toy companion interacts with smart environments: home, theme park
- Equipped with power source, microcontroller and radio transceiver
- User profile and sensors on the toy provide context for personalization
- Standardized IP stack enabling 3rd parties services

REQUIREMENTS

- Physical/link layer
  - LP Wi-Fi
- Star topology, mobile nodes
- Throughput/latency
  - Audio streaming, low-latency for real-time voice
- Traffic patterns
  - One-to-one, many-to-one, pub/sub
- Neighbor and service discovery
- Security device authentication

Smart Parking application

- Parking sensor embedded in the ground detecting presence of cars
- Multihop communication between sensors until GW is reached (installed on lamposts, traffic lights)
- Car drivers are notified via mobile apps on their smartphones

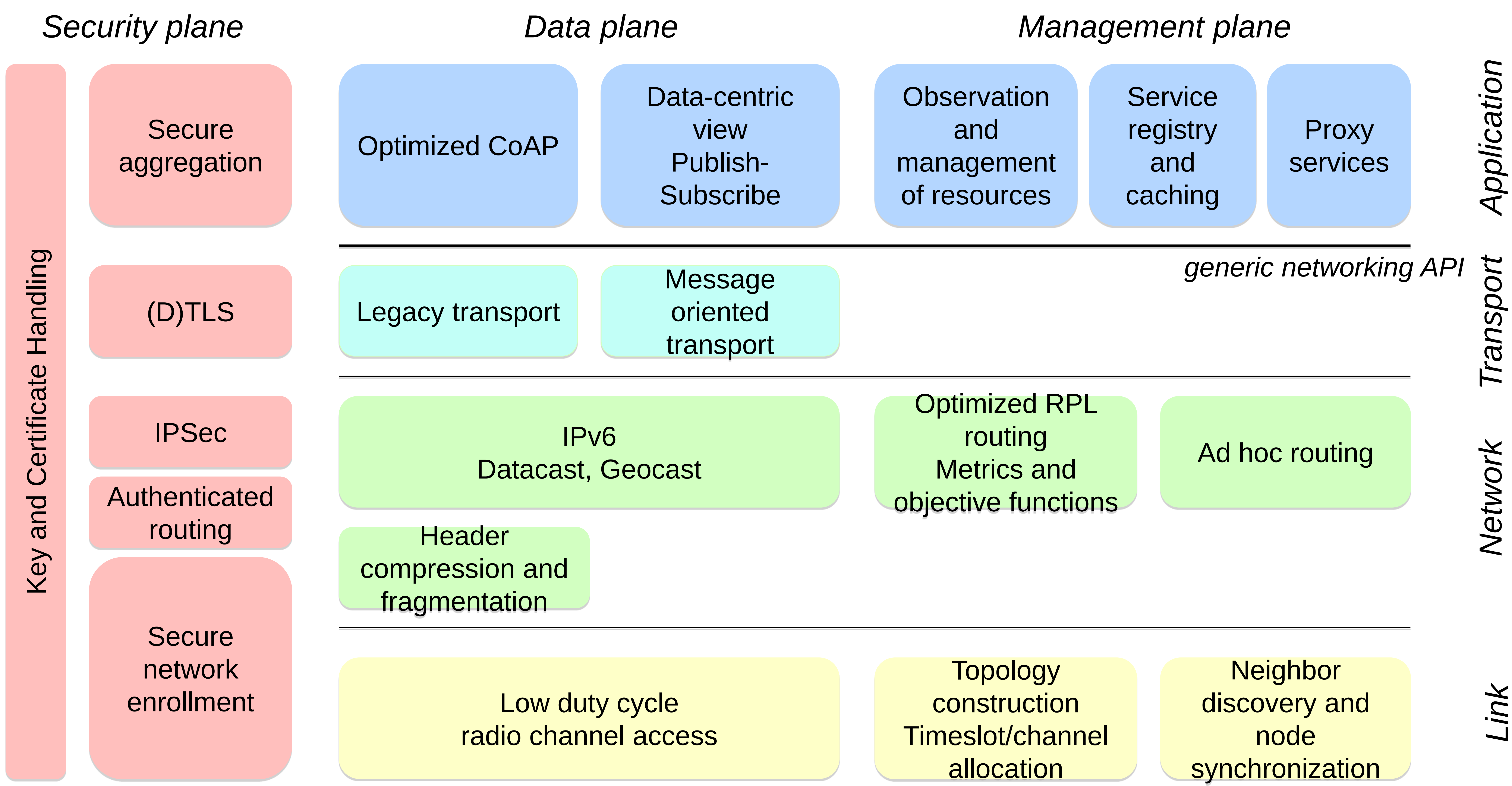


REQUIREMENTS

- Physical/link layer
  - IEEE 802.15.4, duty cycled MAC
- Mesh topology, static nodes
- Throughput/latency
  - Low throughput (parking event), medium latency but reliable
- Traffic patterns: mainly multi-hop data collection
- Routing metrics
- Secure authentication of parking nodes

CALIPSO IoT Architecture

- Many different requirements, but one unified architecture



Research contributions

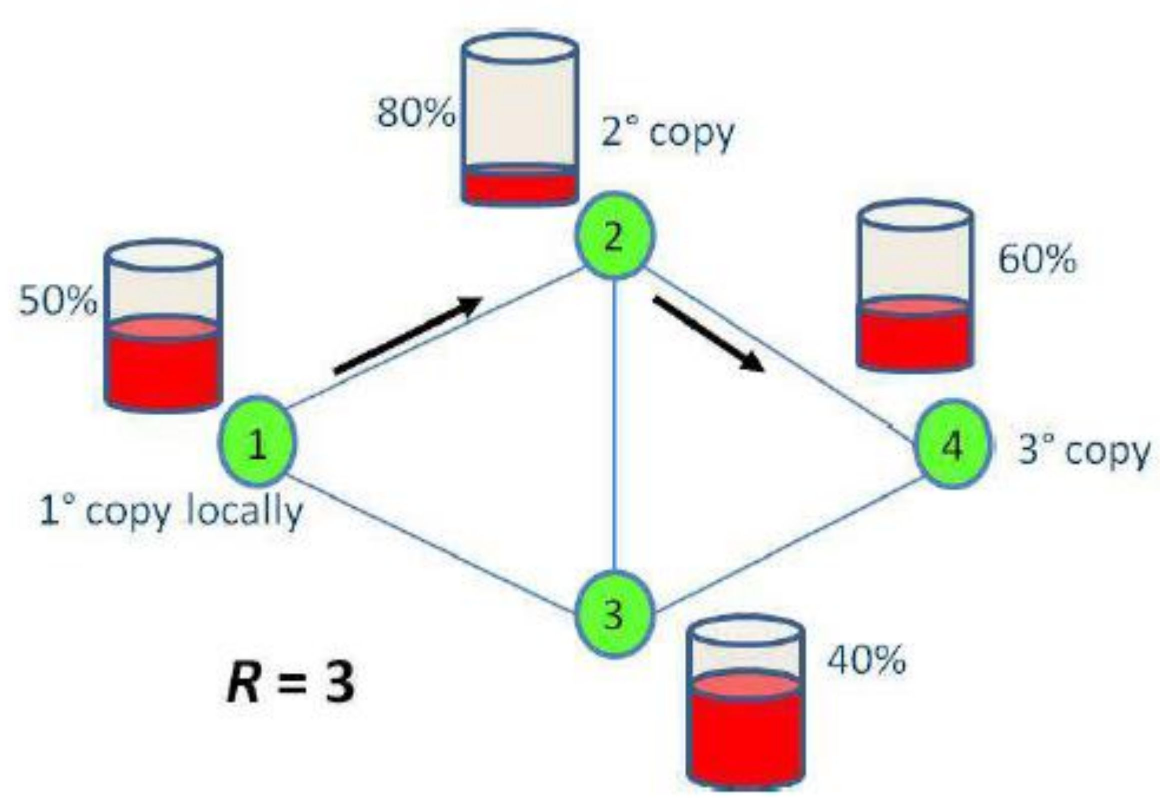
- Each functionality of the architecture addressed in specific technical contributions
- Link layer
  - Optimized 802.15.4 MAC: New neighborhood maintenance mechanism, Wake-on-Idle
  - 802.15.4e TSCH: clock drift compensation and QoS reservation mechanisms
  - Efficient multi-hop forwarding for Low Power 802.11
  - RAWMAC extension of ContikiMAC
- Network layer
  - Opportunistic extension of RPL
  - New routing metrics for low-delay and low-energy paths
  - Comparison between RPL and LoadNG
  - Evaluation of RPL under mobility

Research contributions

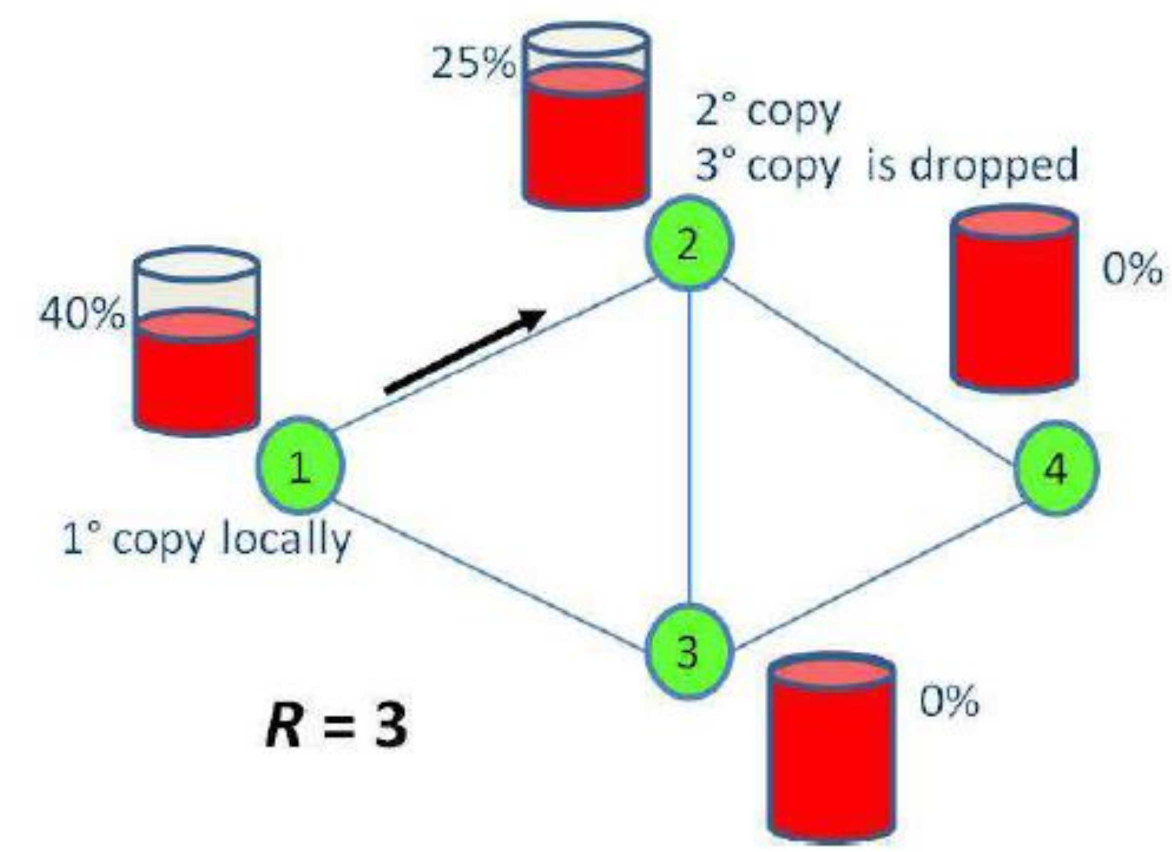
- Transport and Application layer
  - Redundant Distributed data storage and retrieval mechanism
  - HTTP/CoAP proxy and caching mechanism
  - Featurecast: strategy for efficient one-to-many communications based on type of resources
  - Service discovery mechanism for CoAP
- Security
  - Compressed IPSec
  - Compressed DTLS
  - Application of Oauth authorizarion protocol to CoAP
  - Secure distribution of shared keys

Case study 1: Distributed data storage

- Goal: increase data resilience and storage capacity of the nodes (Smart Toys, Smart Parking)
- Perform in-network storage when data cannot be sent to the Internet
- Key idea: distribute new sensing data to nodes with higher memory space
  - Donor node is chosen among 1-hop neighbors
- Hop-by-hop replication of the same data to increase robustness



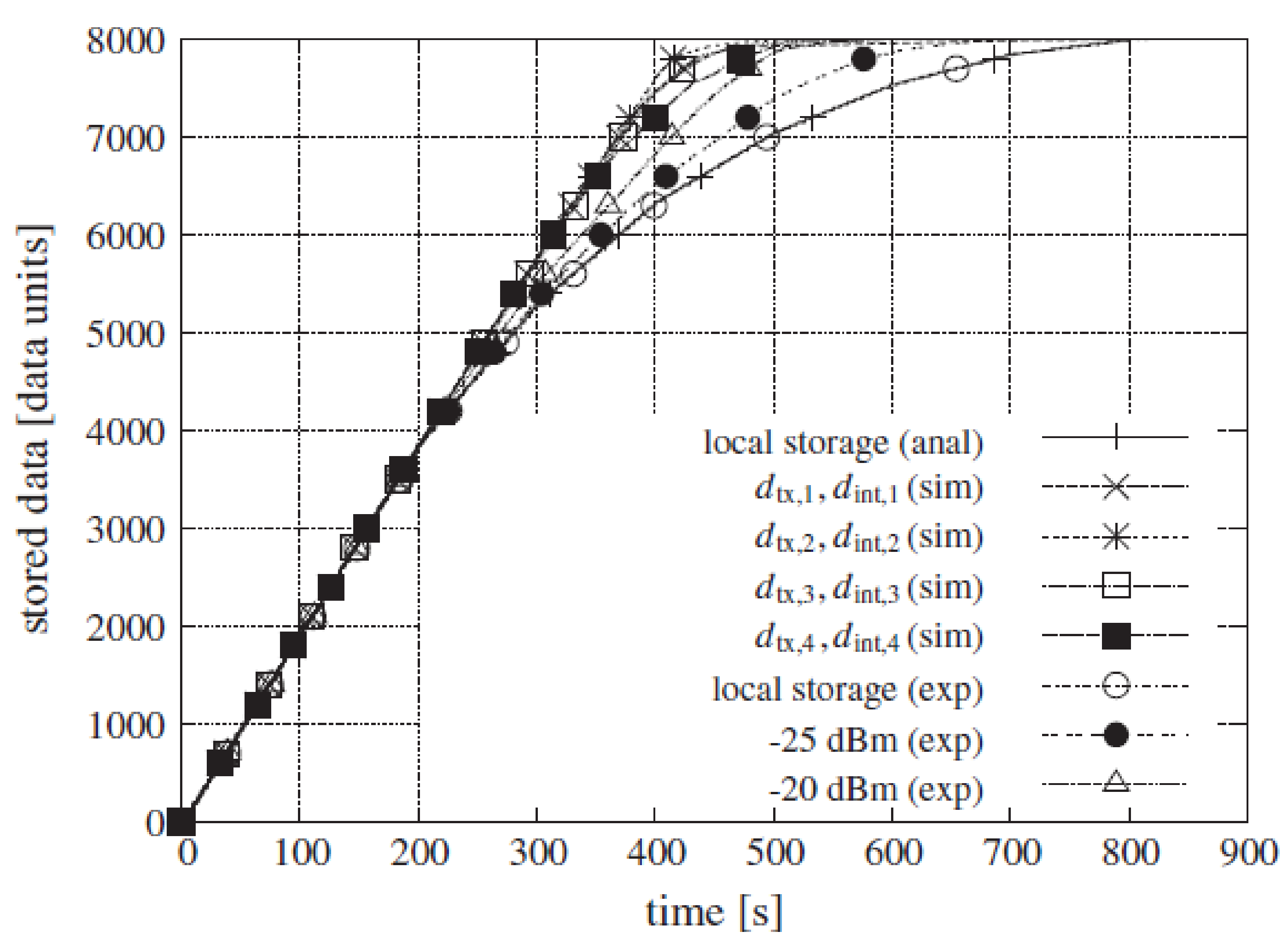
(a) 2-hop path.

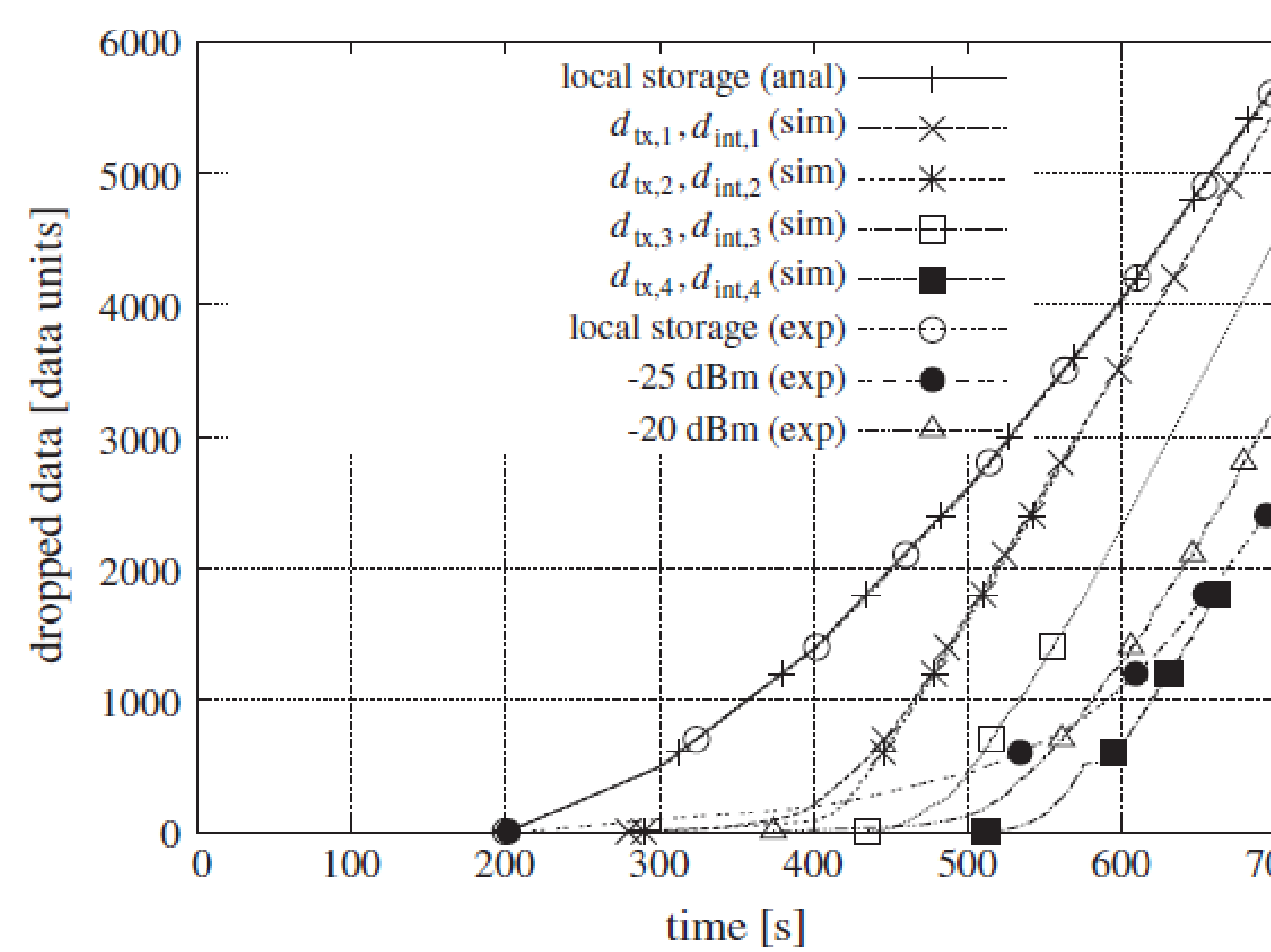


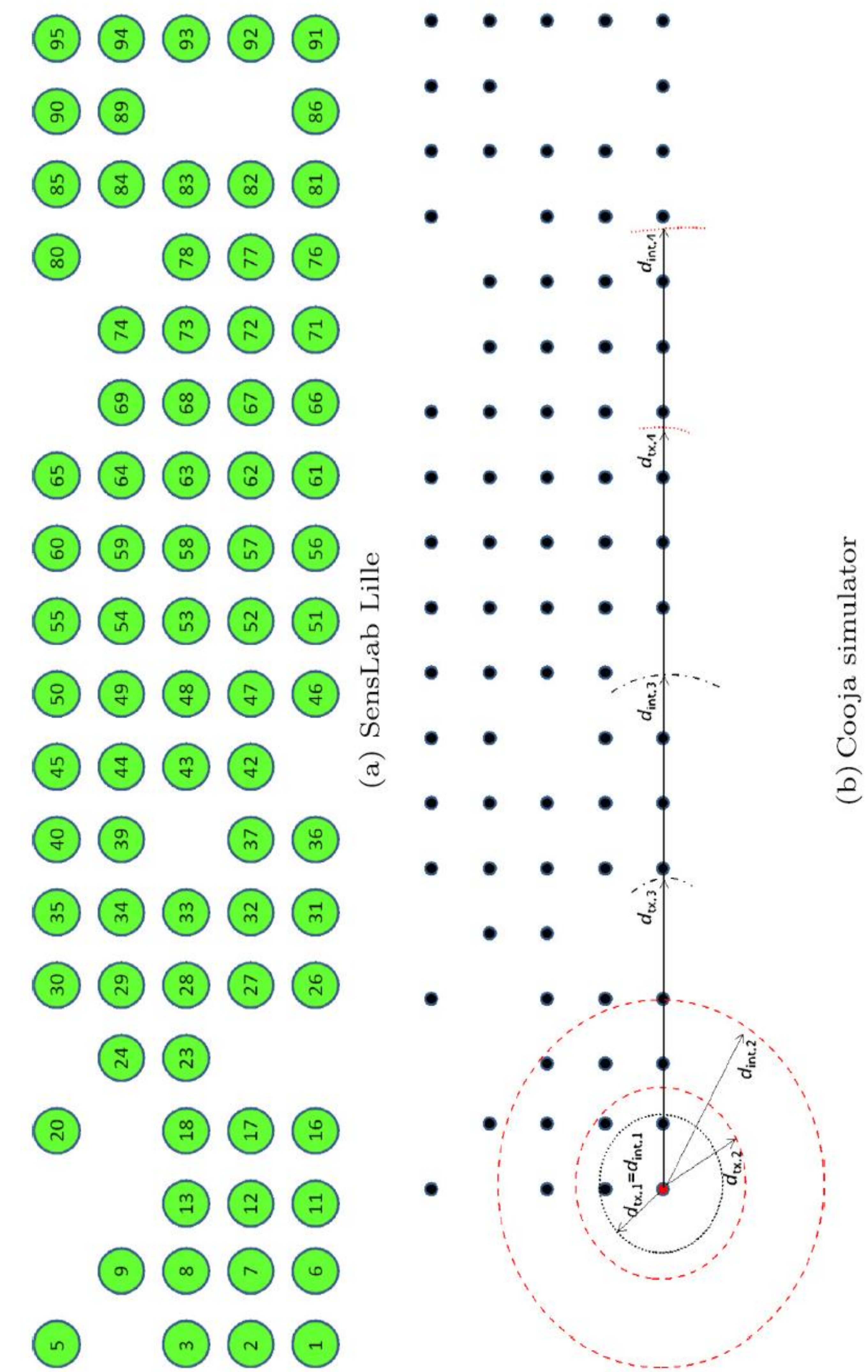
(b) 1-hop path.

Performance results

- Senslab experiments vs. Cooja simulations
  - Tests with wsn430v14 Lille platform (cc2420, MSP430 mic)
  - 80 nodes, variable tx power
  - sensing interval in [2-10] seconds for each node
- Tuning Cooja tx range to reproduce Senslab radio environment

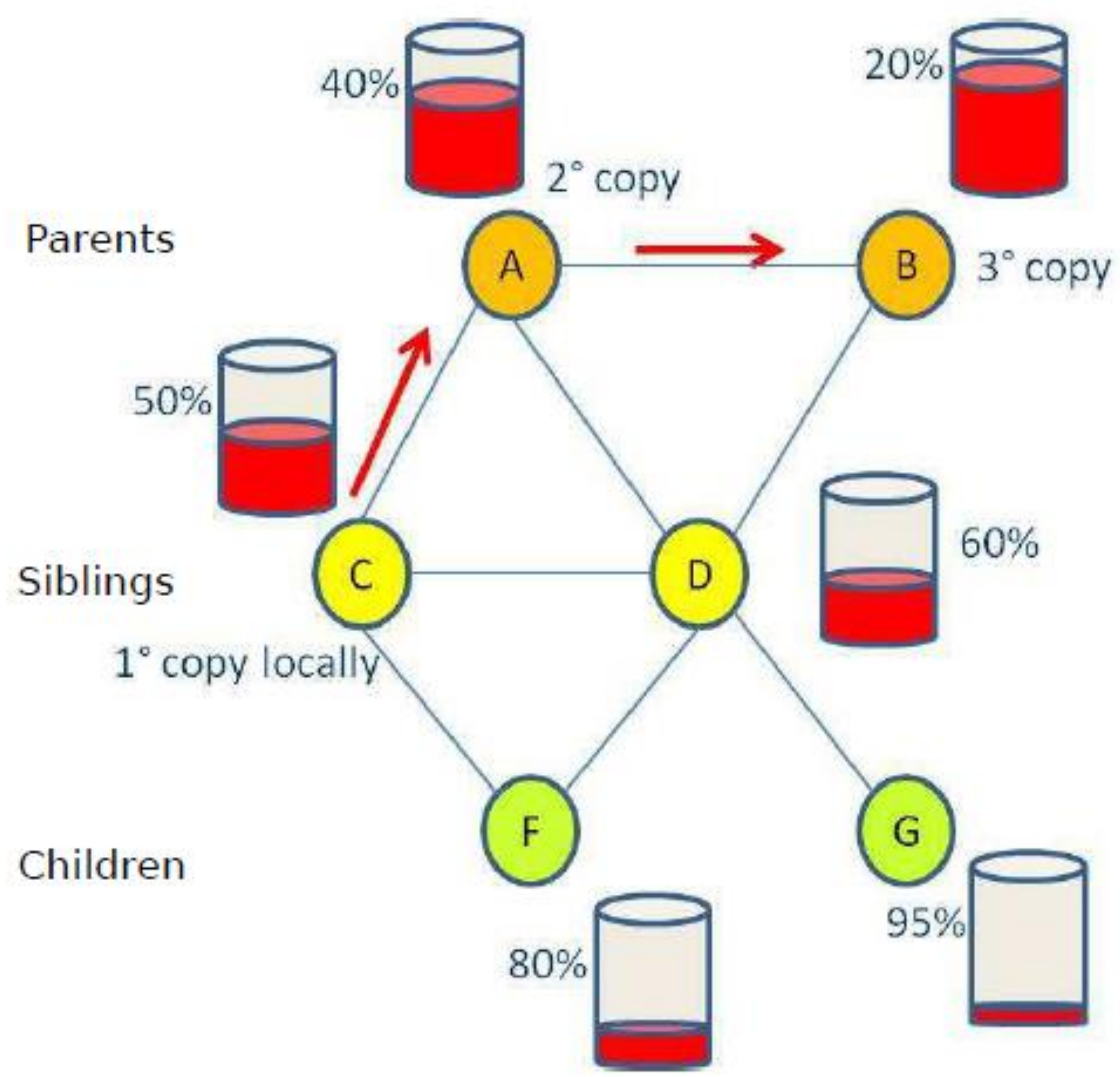


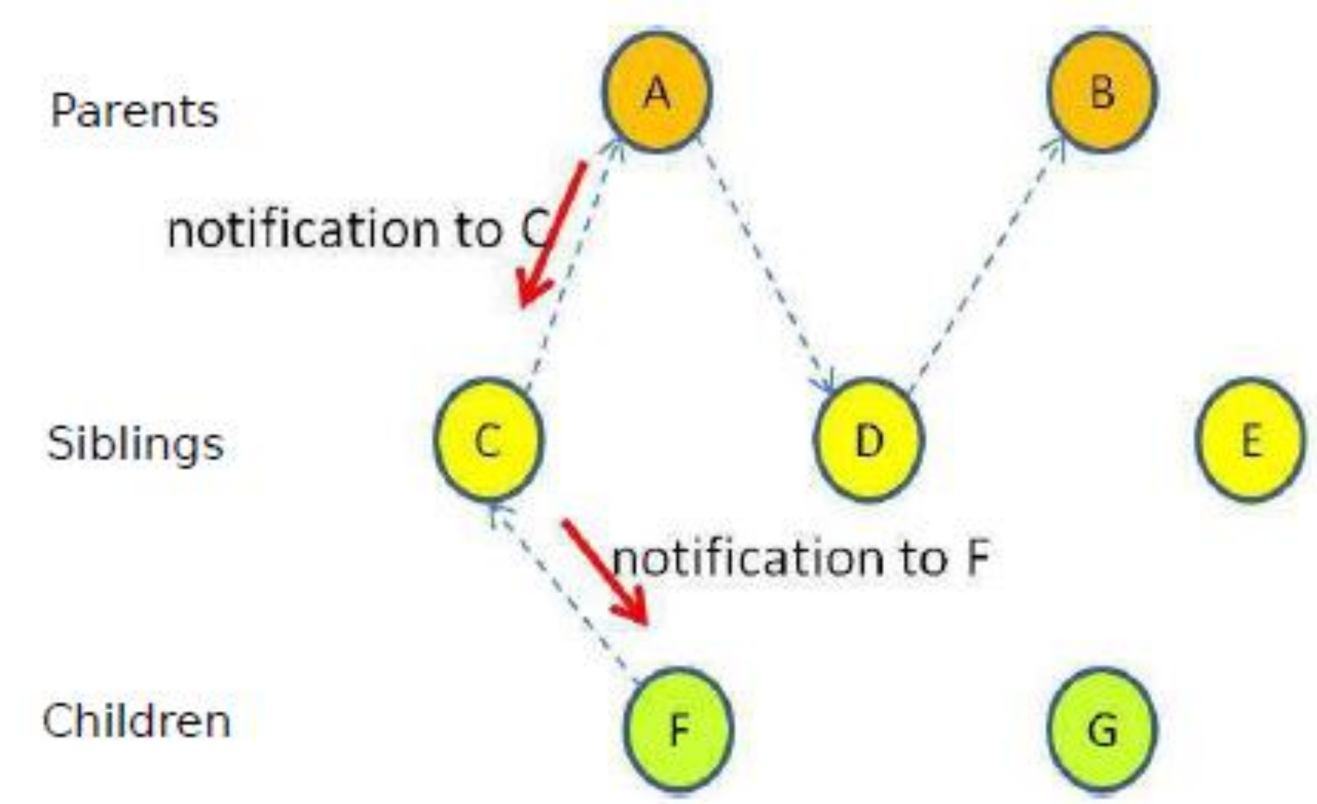




RPL support

- Use RPL routing to efficiently distribute the data in the network
  - Prioritize storage at nodes closer to the sink/DAG root and with most available space
- Periodic data retrieval of the closest replica

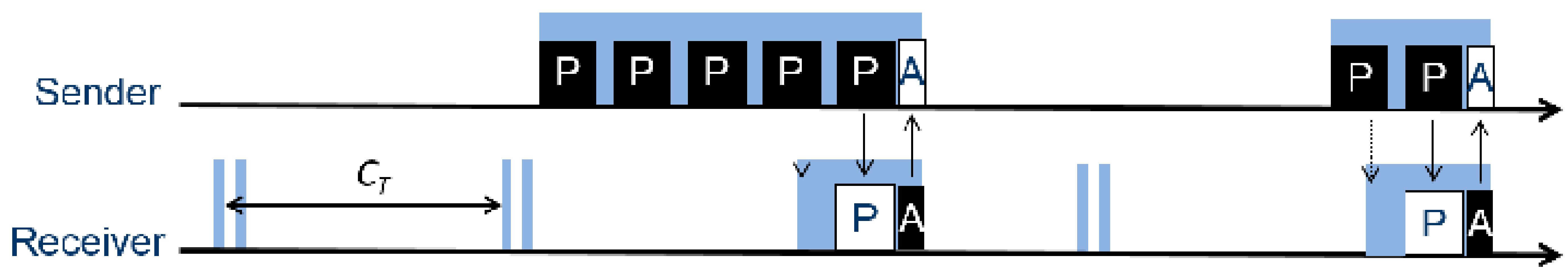




P. Gonizzi, G. Ferrari, V. Gay, and J. Leguay, "**Data dissemination scheme for distributed storage for IoT observation systems at large scale**," *Information Fusion*, Special Issue on "Collaborative Wireless Sensor Networks: Architectures, Algorithms and Applications," vol. 22, March 2015, Pages 16-25. Available online 25 April 2013. DOI: 10.1016/j.inffus.2013.04.003.

Case study 2: RAWMAC

- Goal: speed up data collection in duty-cycling networks (Smart Parking)
- Extend ContikiMAC, the default asynchronous RDC protocol of Contiki OS
- Unicast packets are repeated for a sleeping interval untik ack is received
  - Phase locking reduces strobes at subsequent transmissions

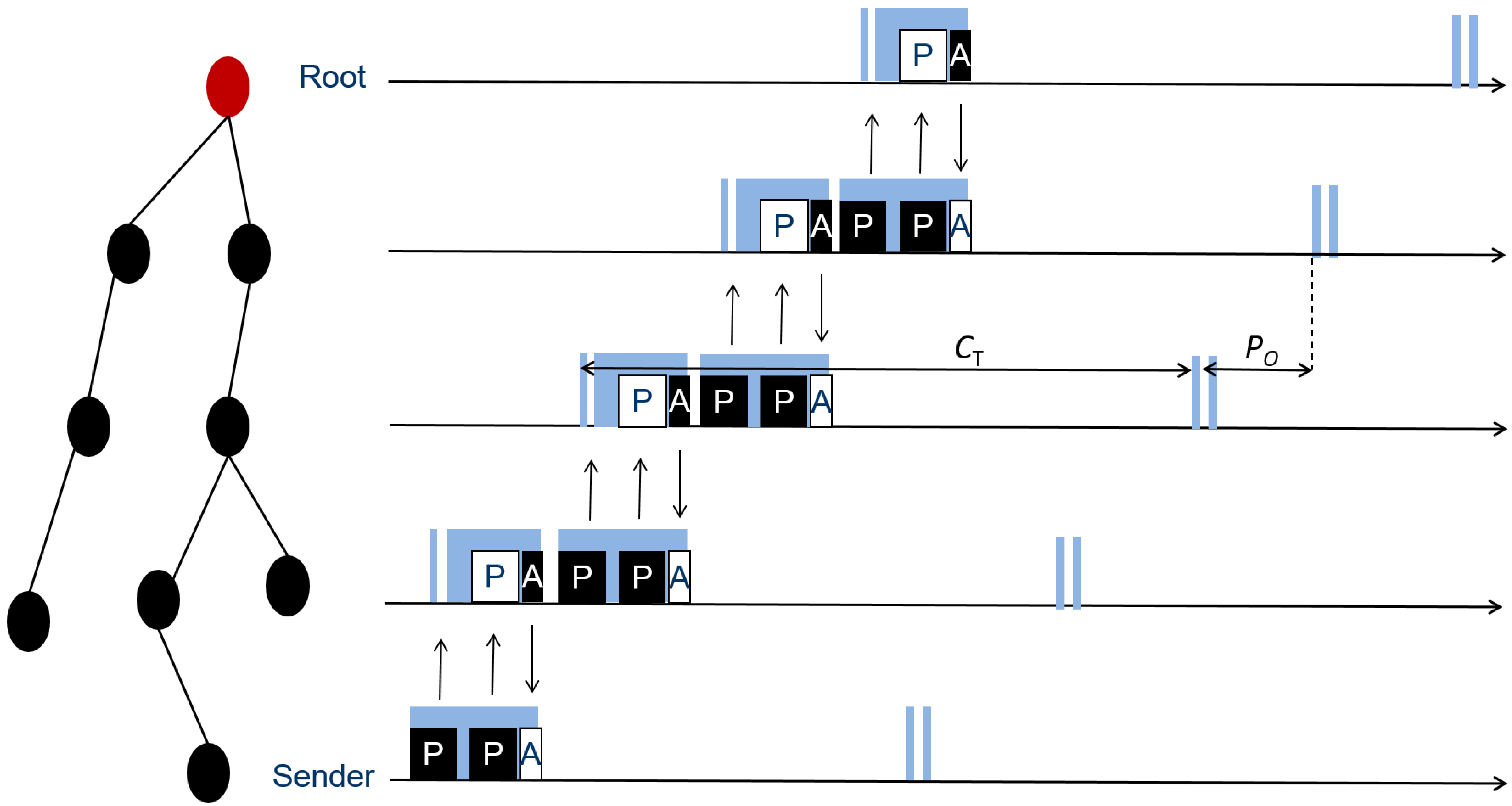


- End-to-end communication requires on average delay of $C_T/2$ at each hop to send a packet

P. Gonizzi, P. Medagliani, G. Ferrari, J. Leguay
RAWMAC: A Routing Aware Wave-based MAC protocol for WSNs
International Workshop on the GReen Optimized Wireless Networks (GROWN 2014), in conjunction with the 10th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob 2014), 2014, (Larnaca, Cyprus, 8-10 October 2014

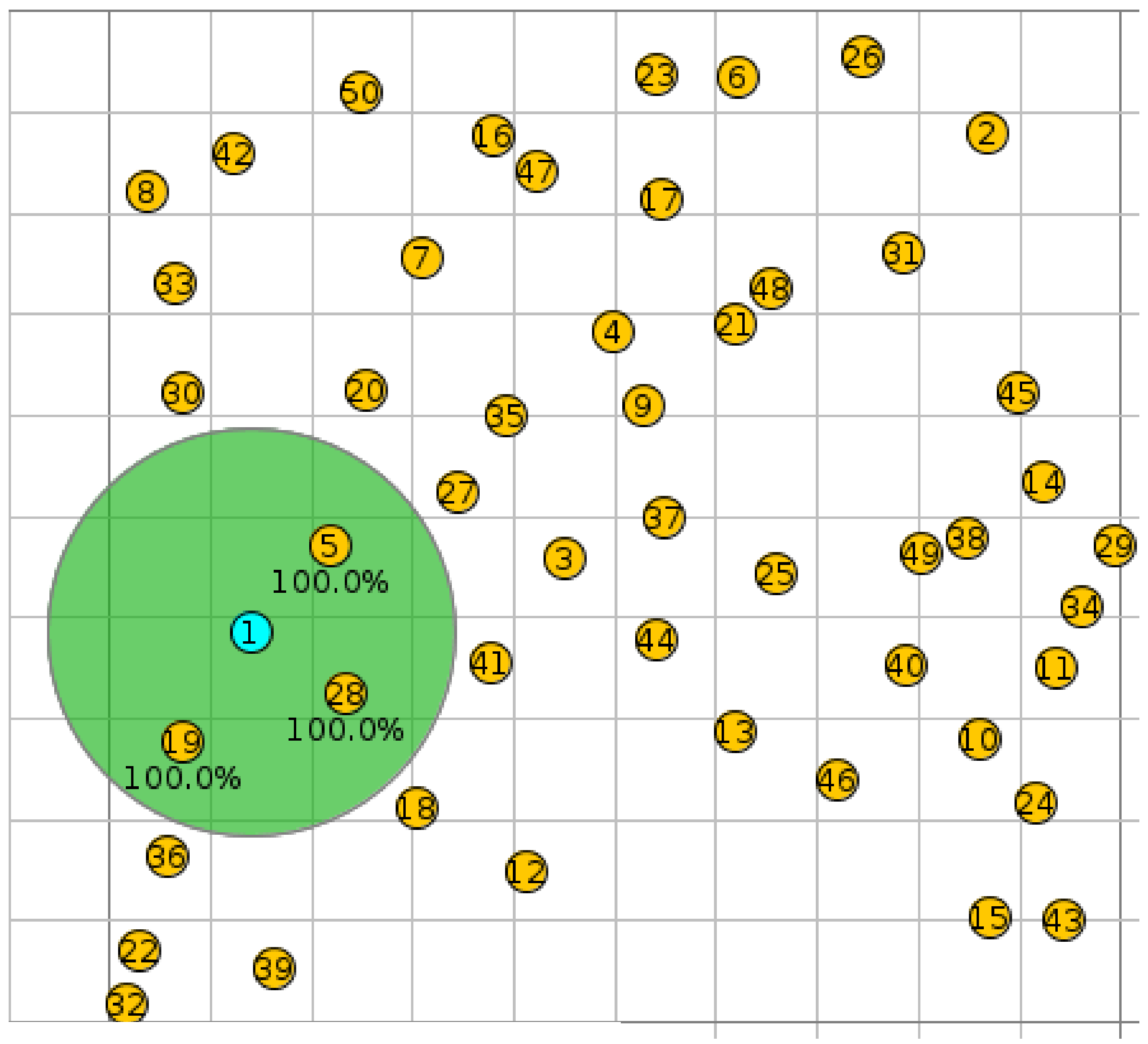
Design of RAWMAC

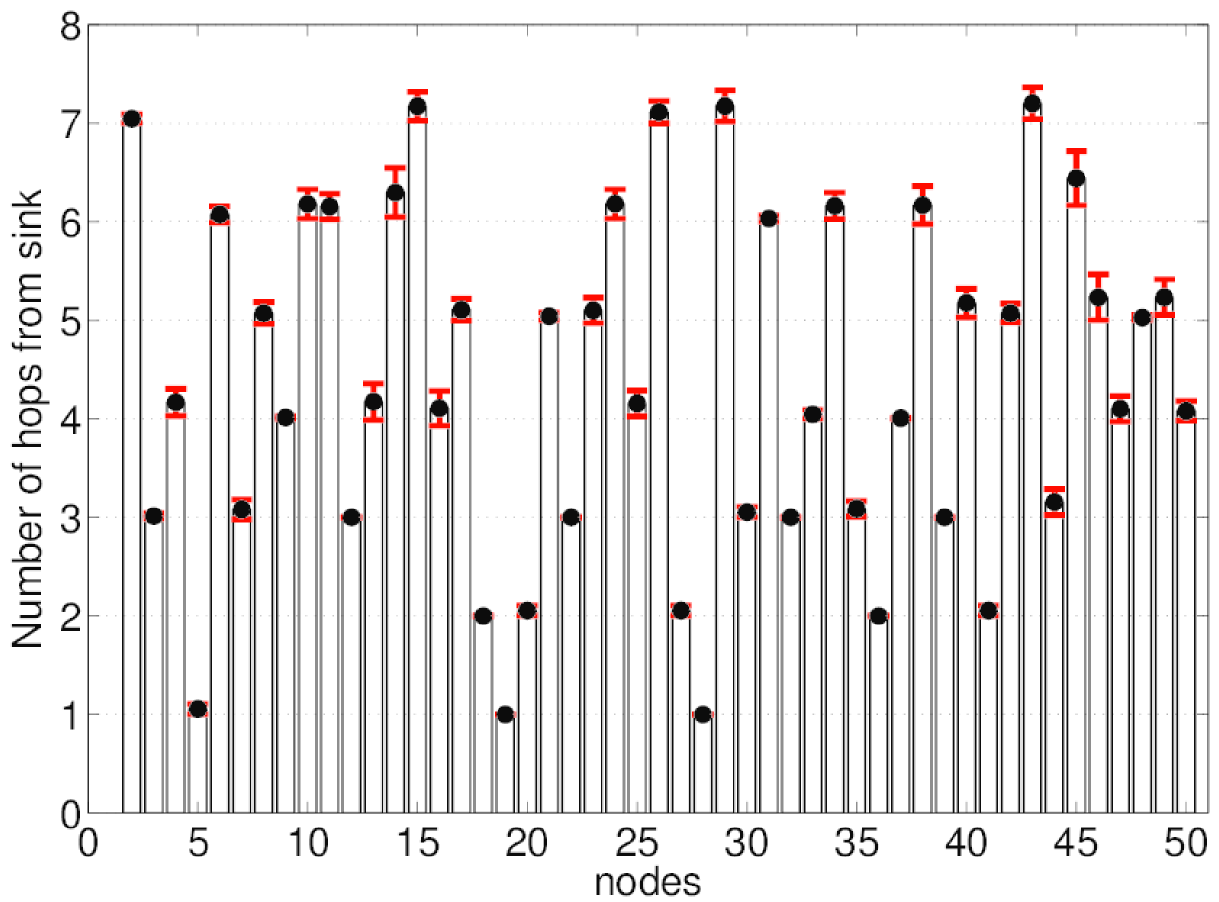
- Key idea: extend ContikiMAC by aligning wake-up phase to the parent in RPL tree
  - Phase offset is configured with P_o parameter (optimal value should be chosen carefully)
  - Including mechanism to mitigate clock drifts

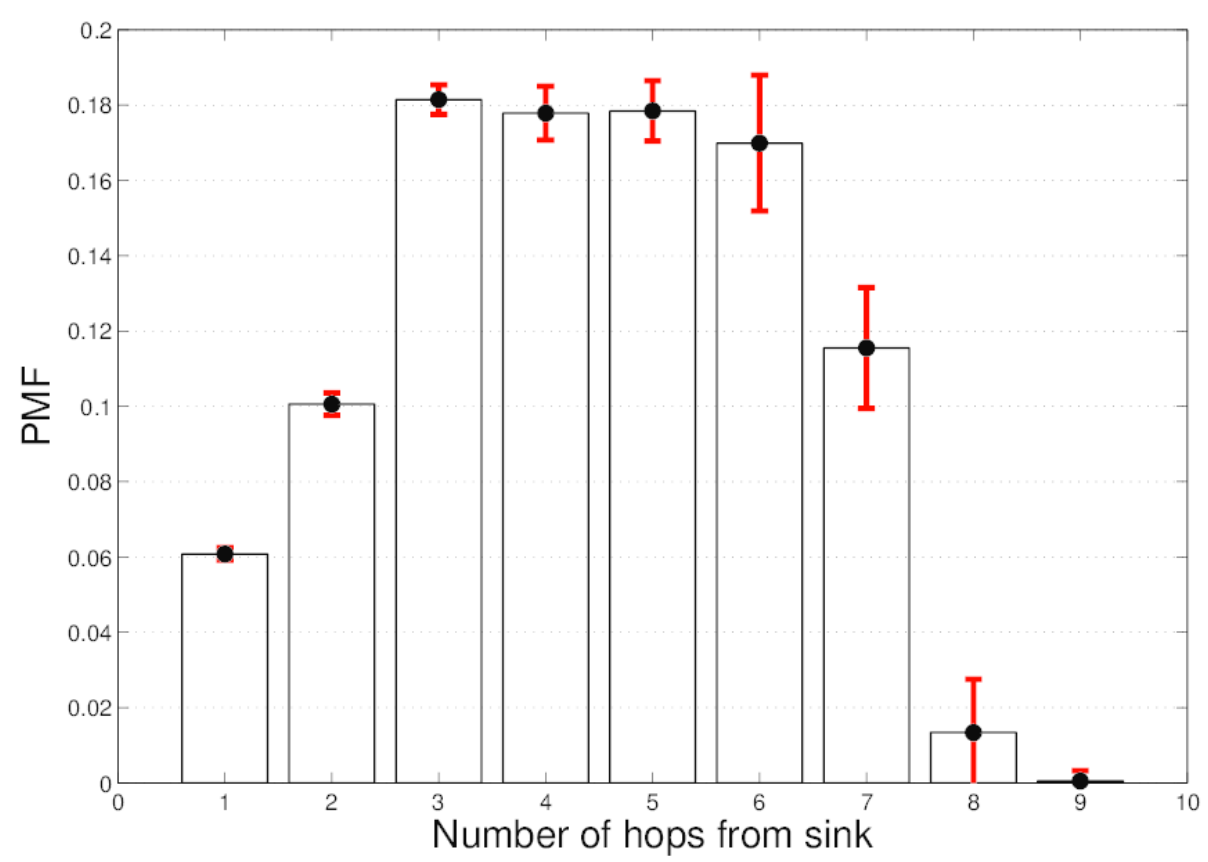


Scenario

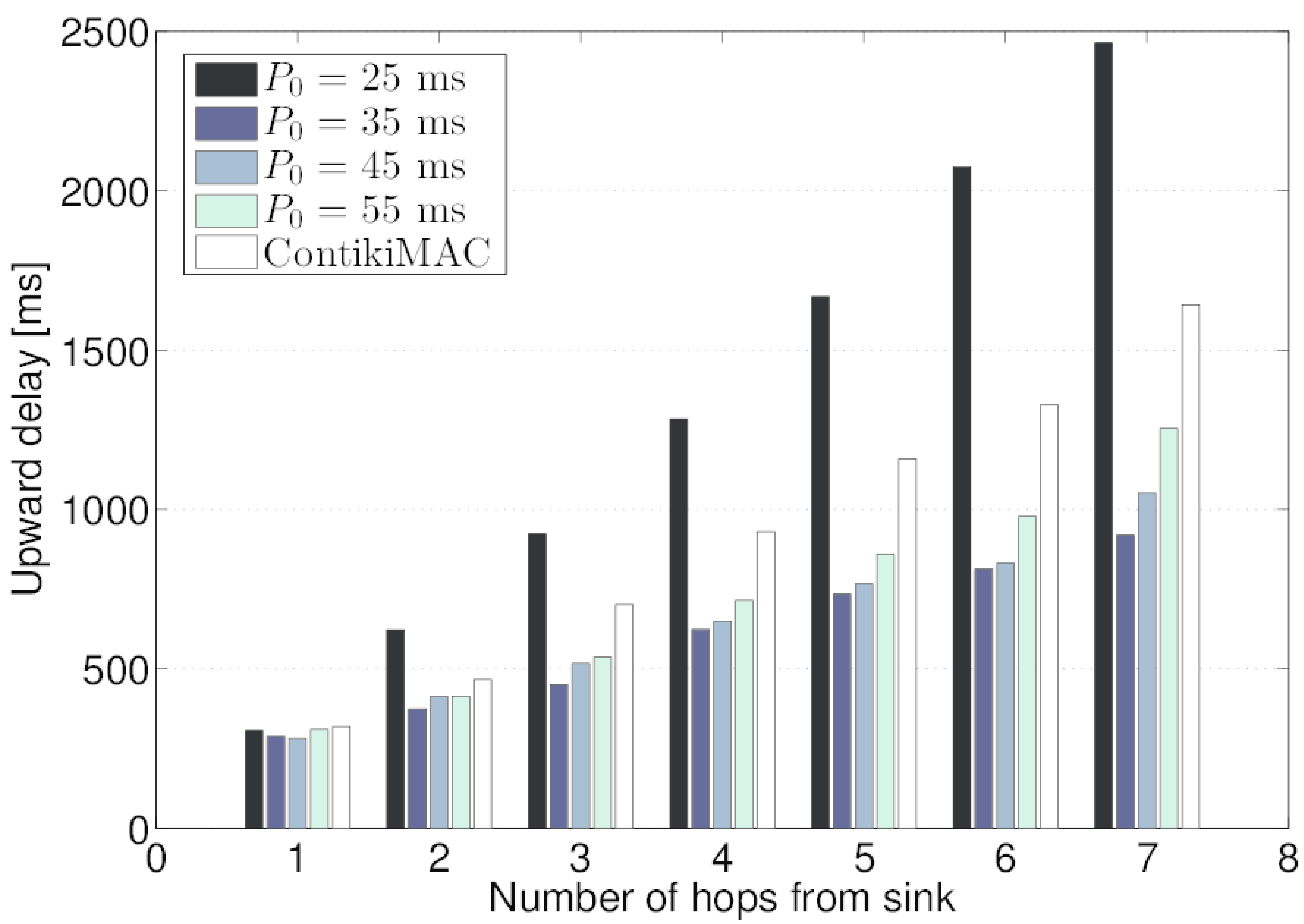
- RAWMAC implementation in Contiki OS
- Tests in Cooja network simulator
  - Emulated Tmote Sky platform
- Random topology with N=50 nodes
- Multihop RPL tree
- Low traffic rate (1 pkt every *T*=120 s)
  - Each node is source of information
  - Data is sent to the sink
- Maximum depth of 7 hops
- Comparison against ContikiMAC

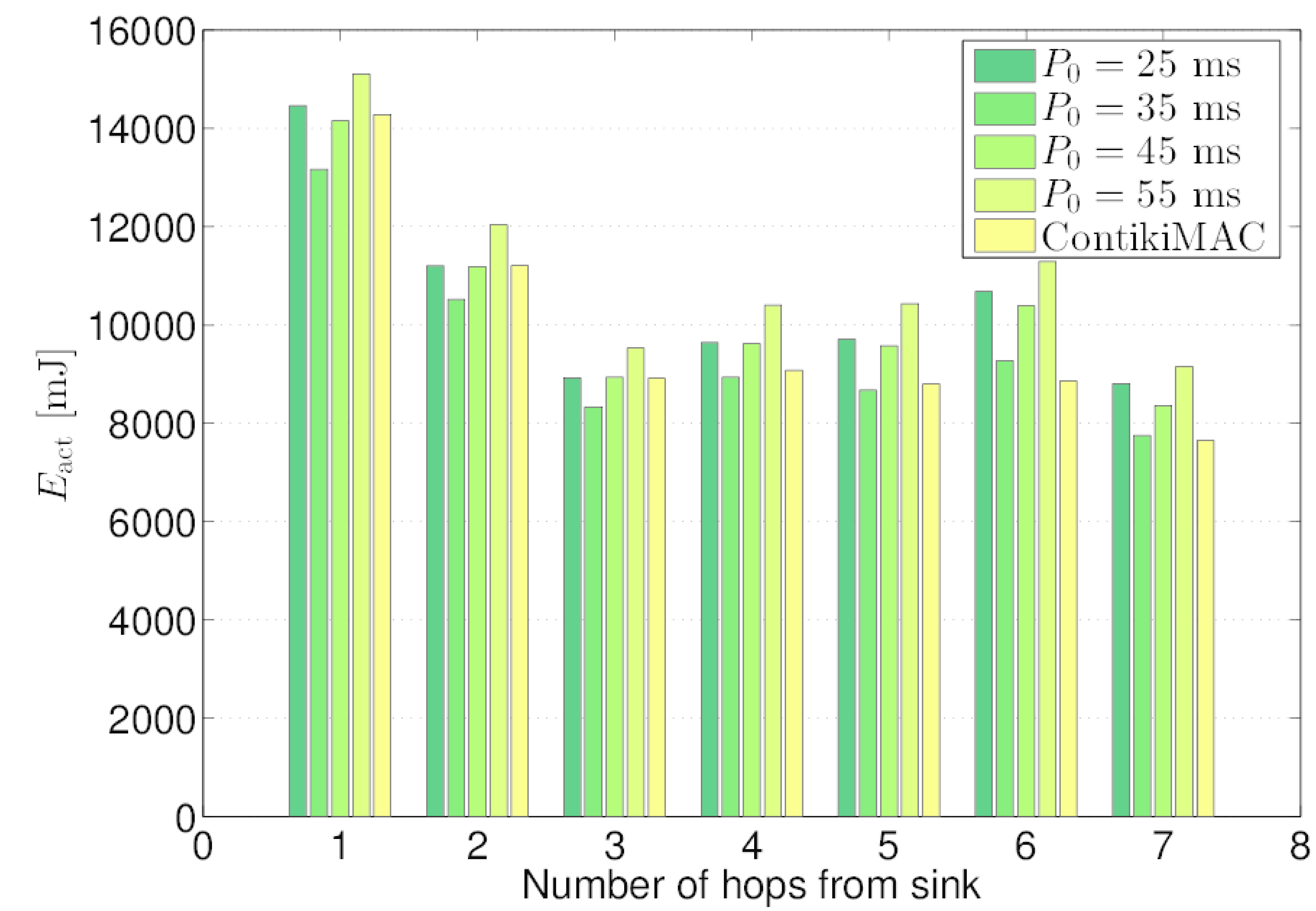






RAWMAC results





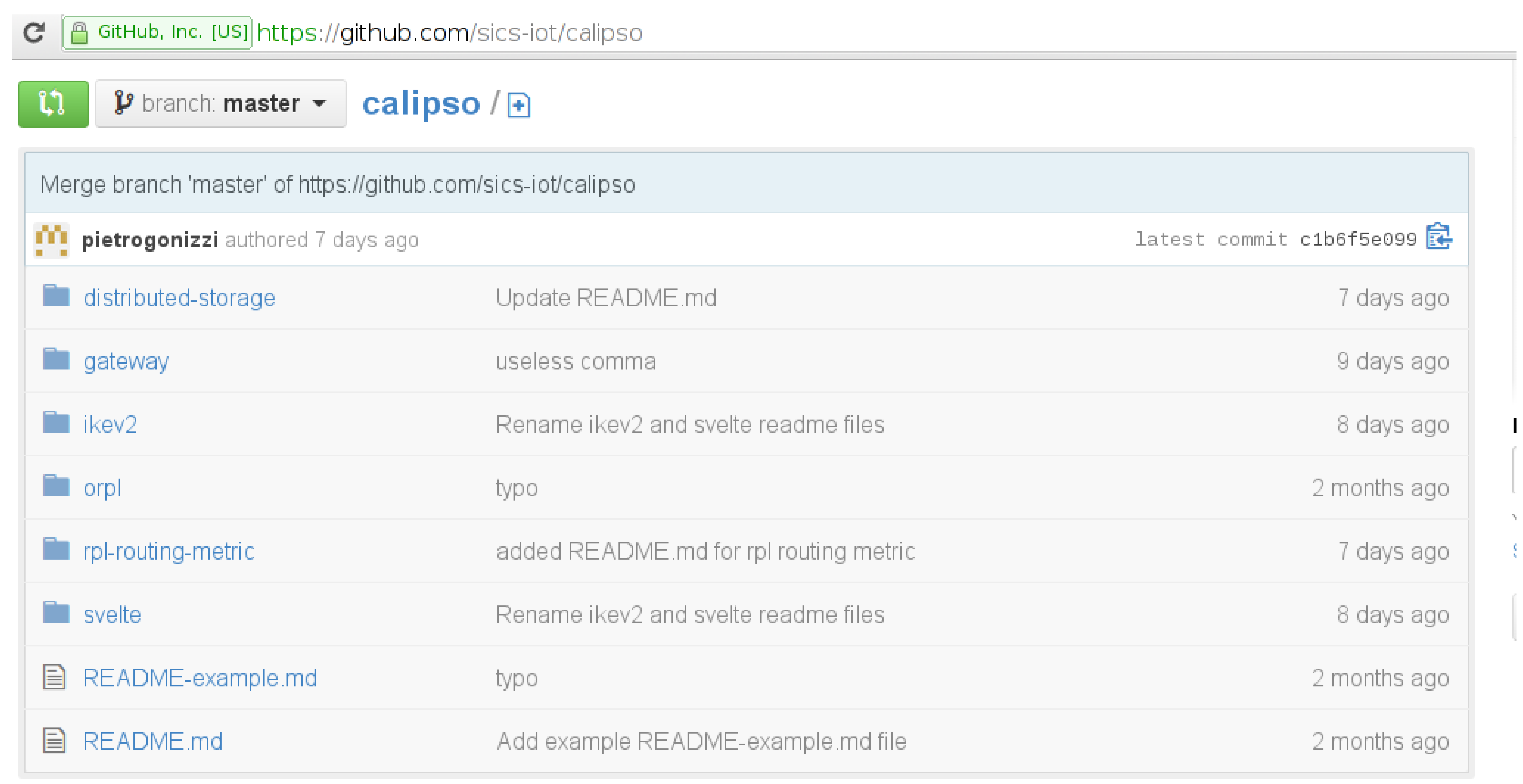
- Up to 40% upward delay reduction compared to ContikiMAC
- Optimal P_o≈35ms
- Energy consumption is comparable

Outline

- Project overview
- IoT architecture and use cases
- Research challenges
- Dissemination and standardization
- Case studies: Distributed data storage, RAWMAC
- Calipso experimental evaluation
- Conclusions

Open source software

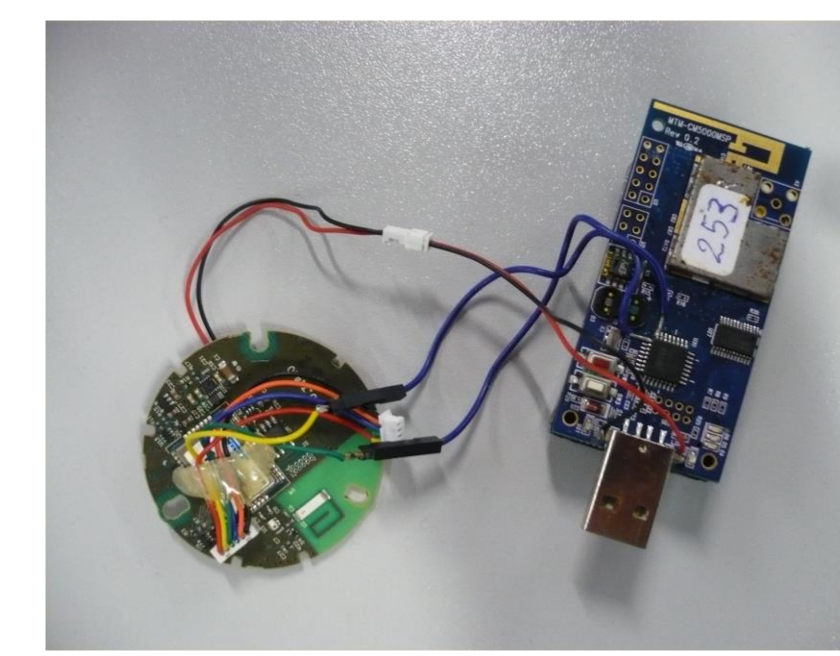
- All CALIPSO contributions are hosted on public github: https://github.com/sics-iot/calipso
  - It contains code of each single module (not integrated)
- Calipso integrated repository: https://github.com/sics-iot/calipso-integrated
  - forked from official Contiki
  - Validated in field trials/use cases

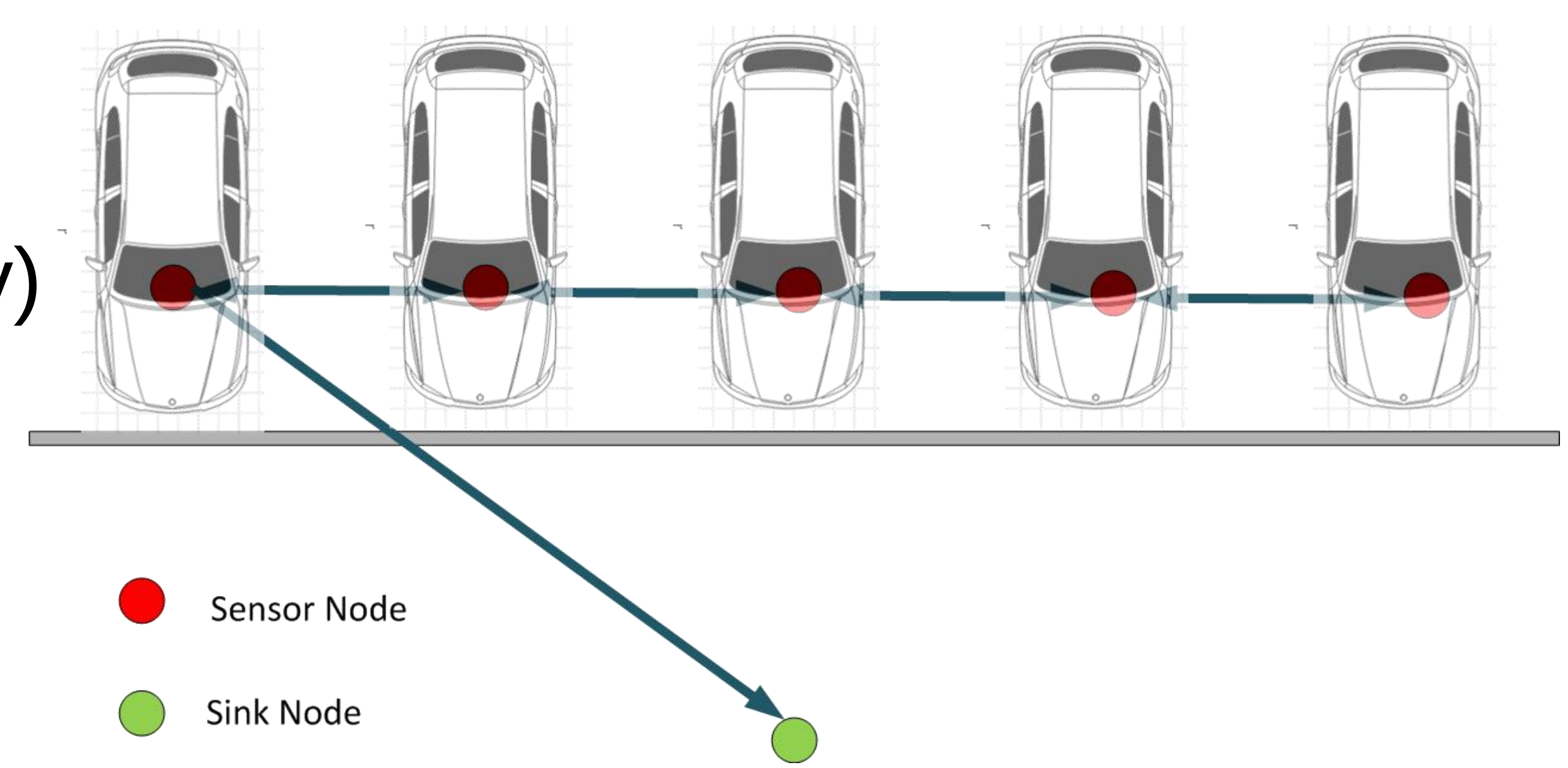


Calipso stack validation with Smart Parking

- Installation site in Barcelona
  - Parking sensor inserted in tarmac

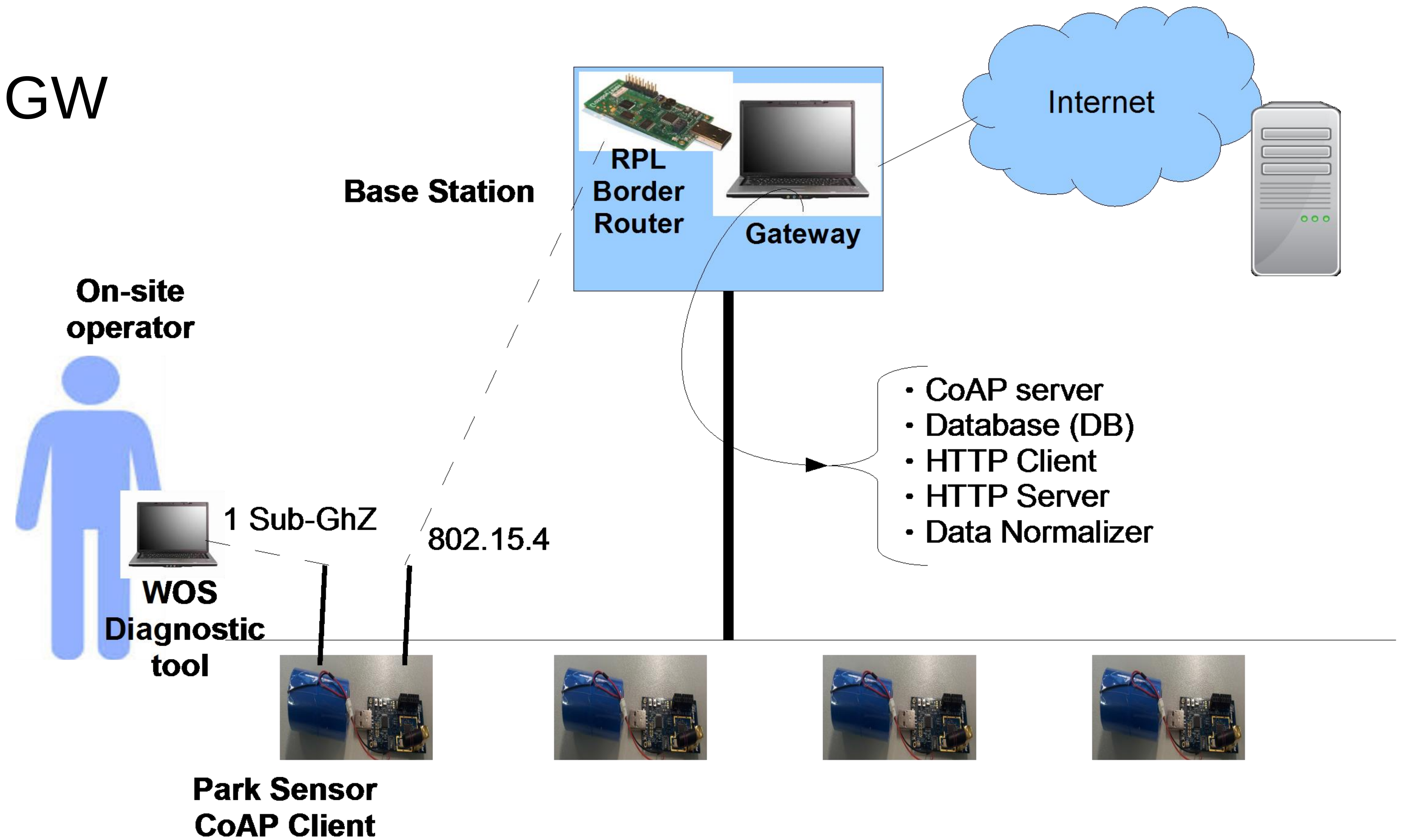
- Prototype
  - Parking sensor: Tmote Sky (with radio stack) + magnetic sensor (for car detection)
    - Connected via serial interface
  - Base station
    - RPL border router (Tmote Sky)
    - Gateway (Linux PC)





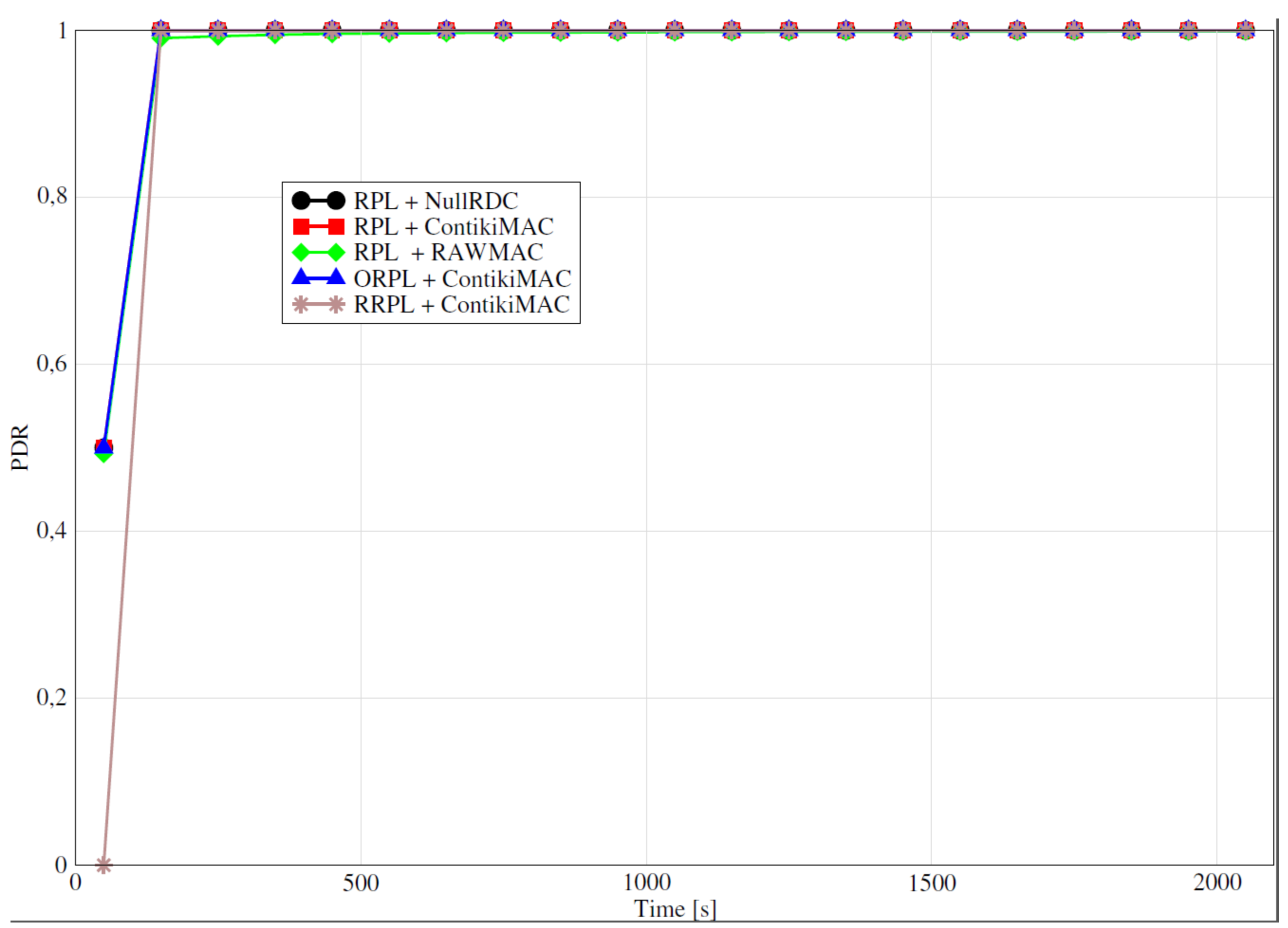
Software architecture

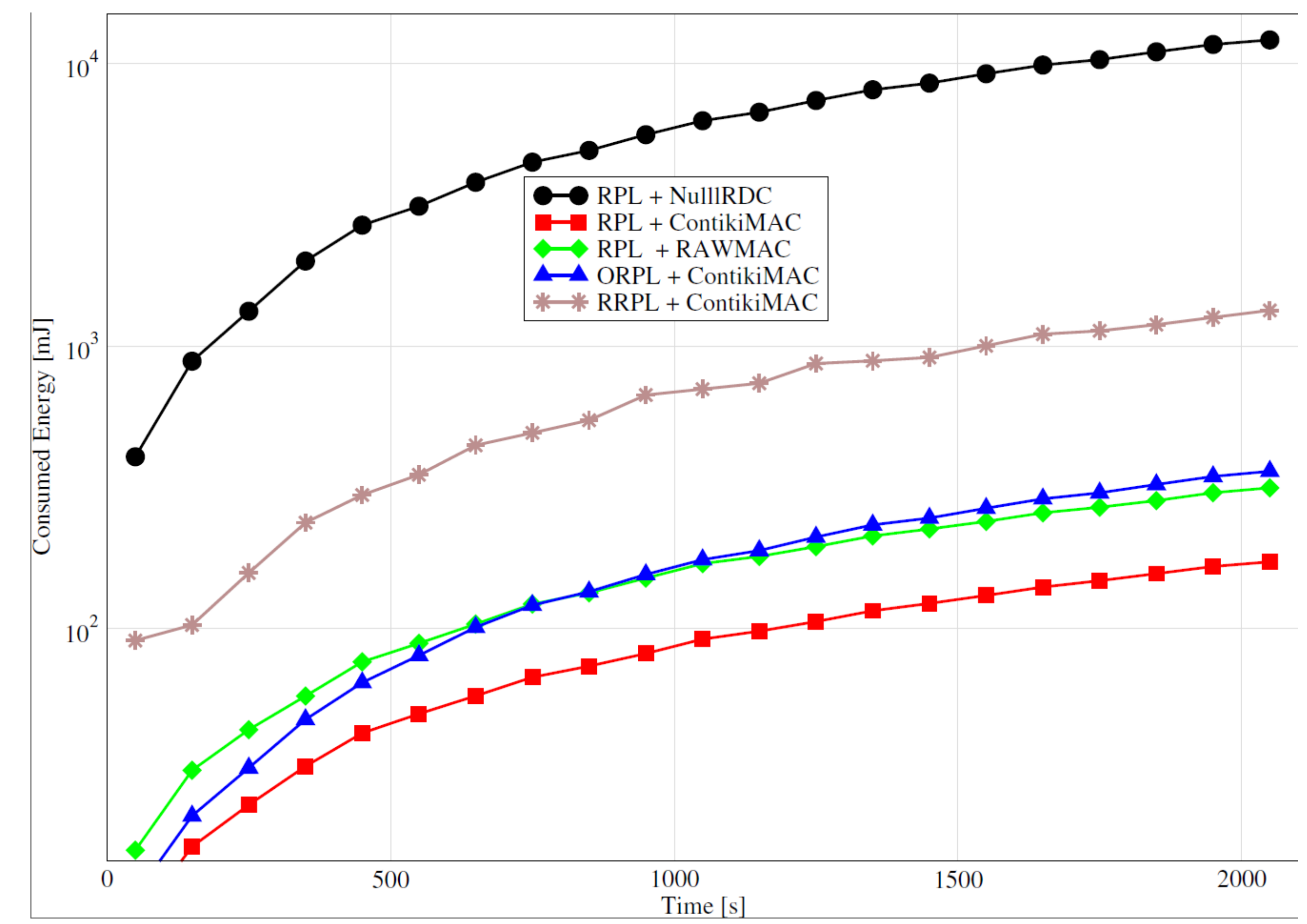
- Each parking sensor is a CoAP client
  - Push parking info and network statistics to GW
- GW stores the statistics and displays results
  - CoAP Server
  - Database
  - HTTP Server
- Performance metrics
  - Consumed energy
  - RPL information and hop count
  - Packet delivery ratio
  - Latency
  - Overhead
  - Presence of a car

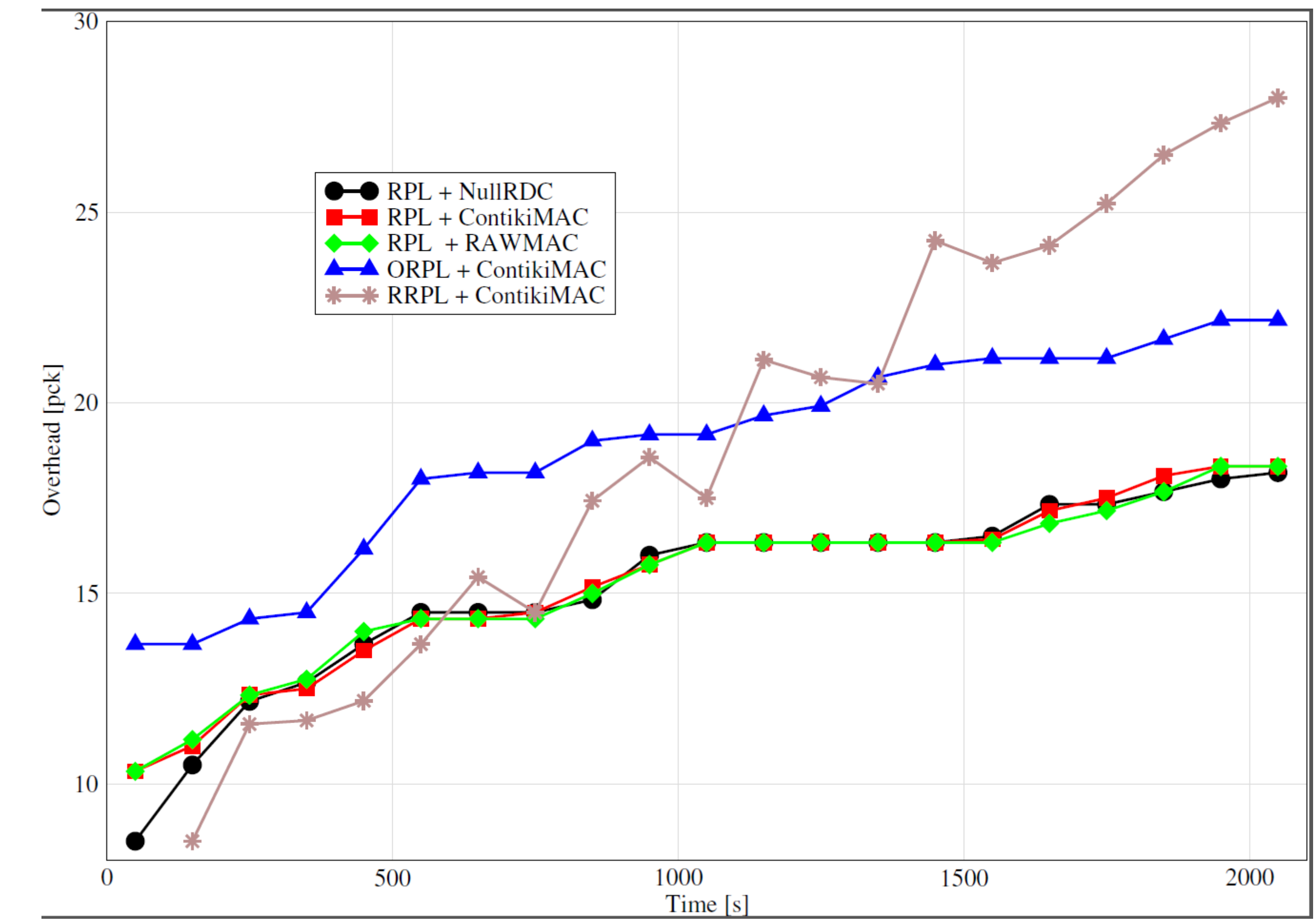


Results

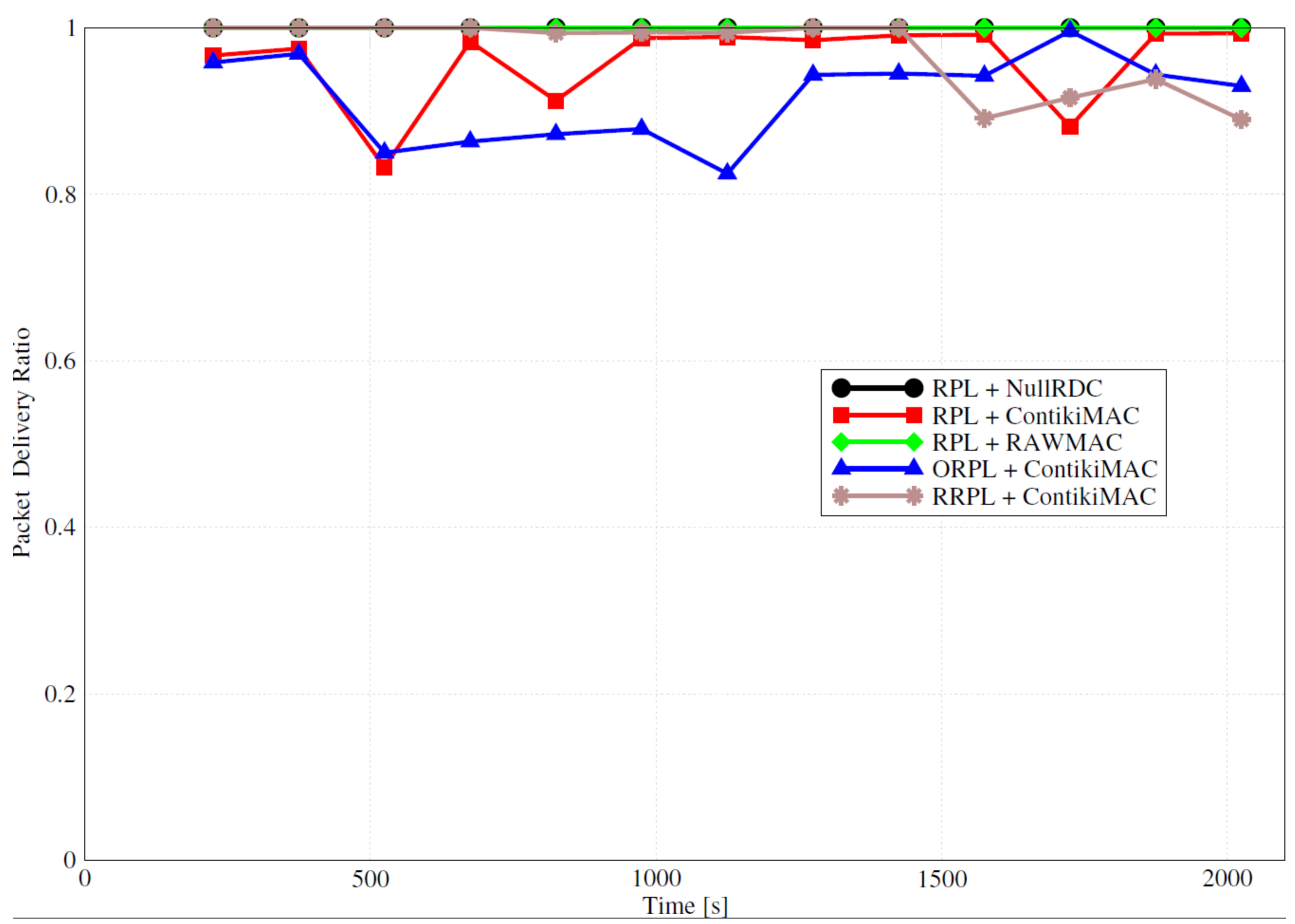
Sim

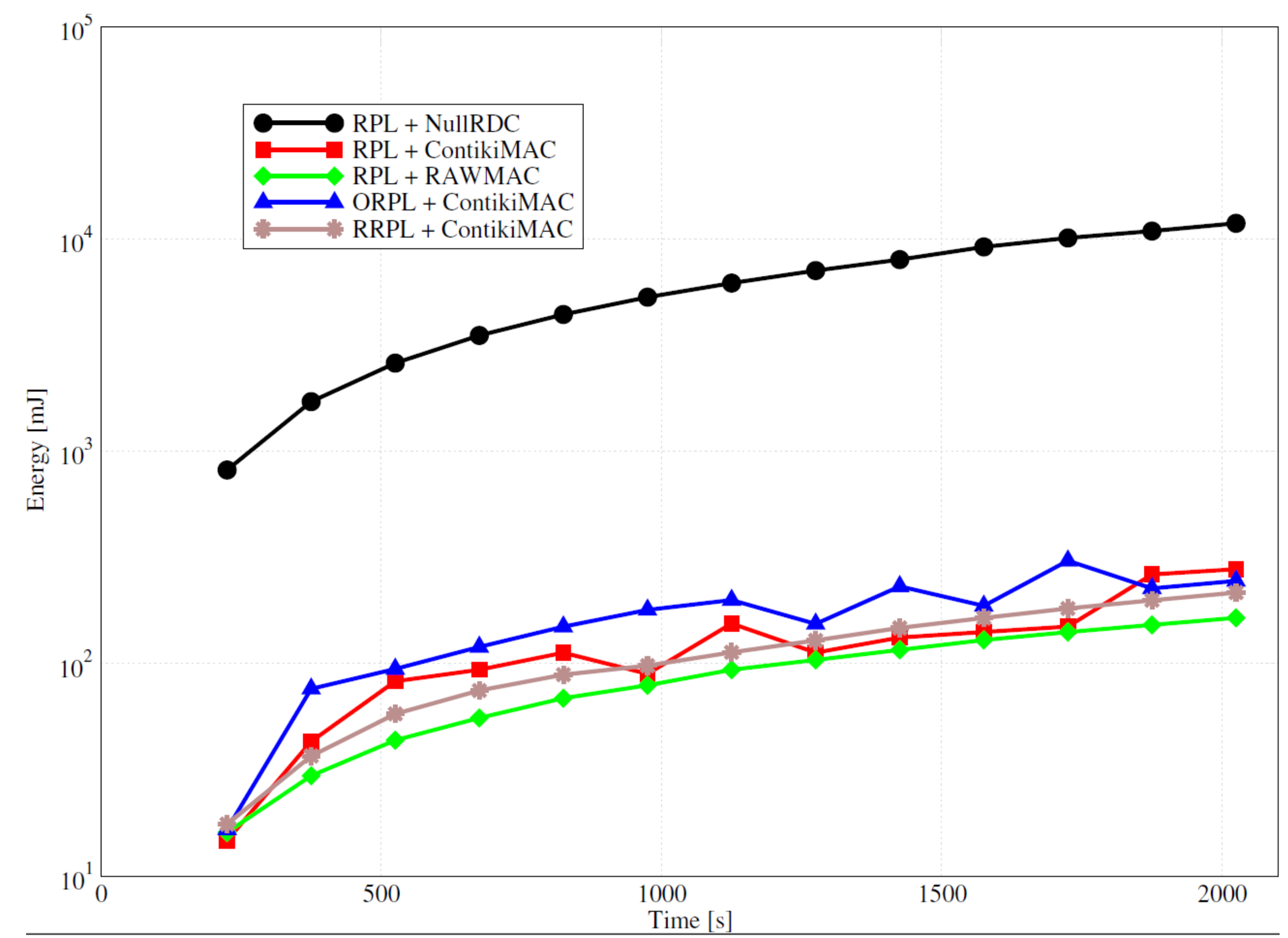


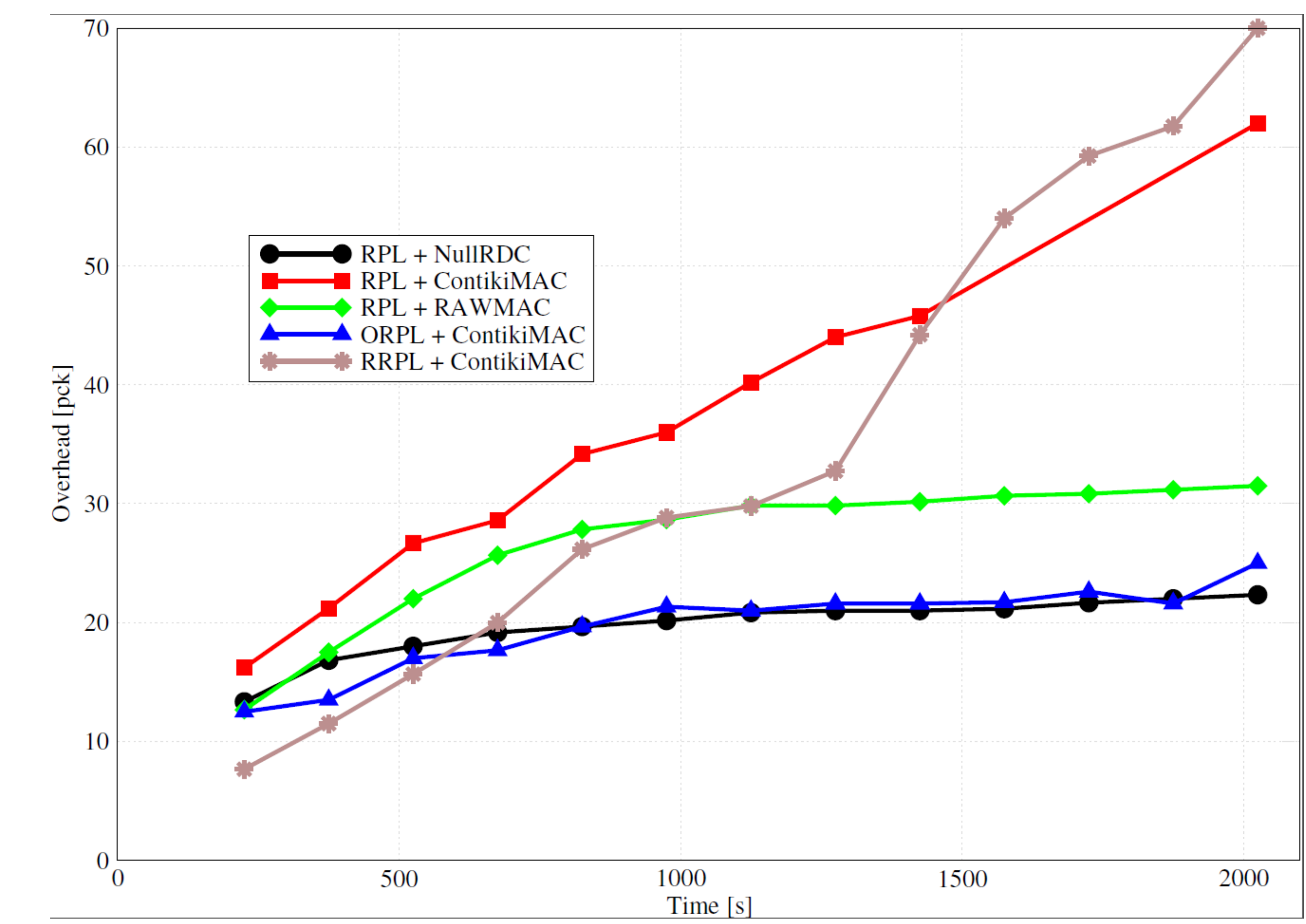




Exp







Conclusions

- CALIPSO has investigated IoT research challenges driven by real applications
  - Smart Parking, Smart Toys, Critical Infrastructure
- Experimentation and real deployment as major outcomes
- Huge dissemination activities
  - A lot of publications
  - Open source software
  - Organized events: IoT & Smart Cities PhD School, seminars
- All the material is available on the official web site
- CALIPSO could really help IoT-LAB users to deploy and make IoT a reality

IoT & Smart Cities PhD School

- International speakers from academia and industry gave lectures tailoring their research field for an interdisciplinary audience. A dedicated discussion panel focused on the interaction and the collaboration between academia and industry in order to depict the future vision of Smart Cities and IoT.
- Location: Castle of Lerici, La Spezia
- 2 successful editions in 2013, 2014…. and 2015 too!
- ~40 participants from Europe
- Project hackathon for the students
- Would be great to have a speaker from INRIA/IoT-LAB in 2015!

http://phdschool.tlc.unipr.it

Thank you

pietro.gonizzi@studenti.unipr.it