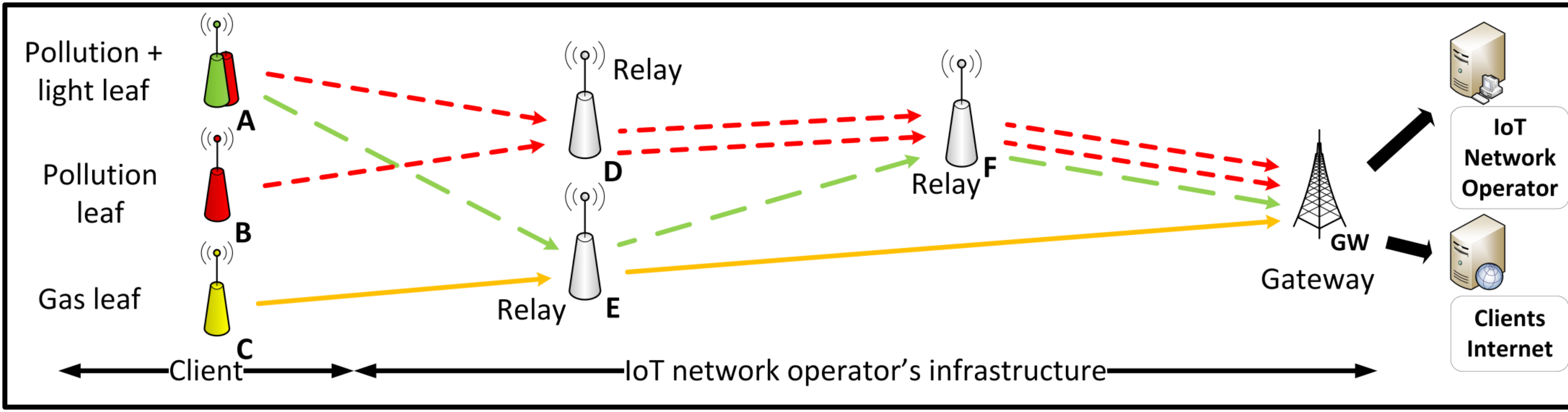
KAUSA: KPI-aware Scheduling Algorithm for Multi-flow in Multi-hop IoT Networks

Guillaume Gaillard, PhD Student, Univ Lyon, INSA Lyon, Inria, CITI, F-69621 Villeurbanne, France
Dominique Barthel, Orange Labs, Meylan, France
Fabrice Theoleyre, CNRS, University of Strasbourg, France
Fabrice Valois, Univ Lyon, INSA Lyon, Inria, CITI, F-69621 Villeurbanne, France

Ad Hoc Now 2016 July 04-06

Context: operating a multi-client multi-hop network

One network for several applications



- Multi-hop shared infrastructure
  - Applications & Flows
- Deployment costs:
  - capacity
  - resource allocation
- Guarantying Quality of Service (QoS)
- Specific Service Level Agreements (SLAs):
  - traffic patterns
  - Key Performance Indicators (KPIs)

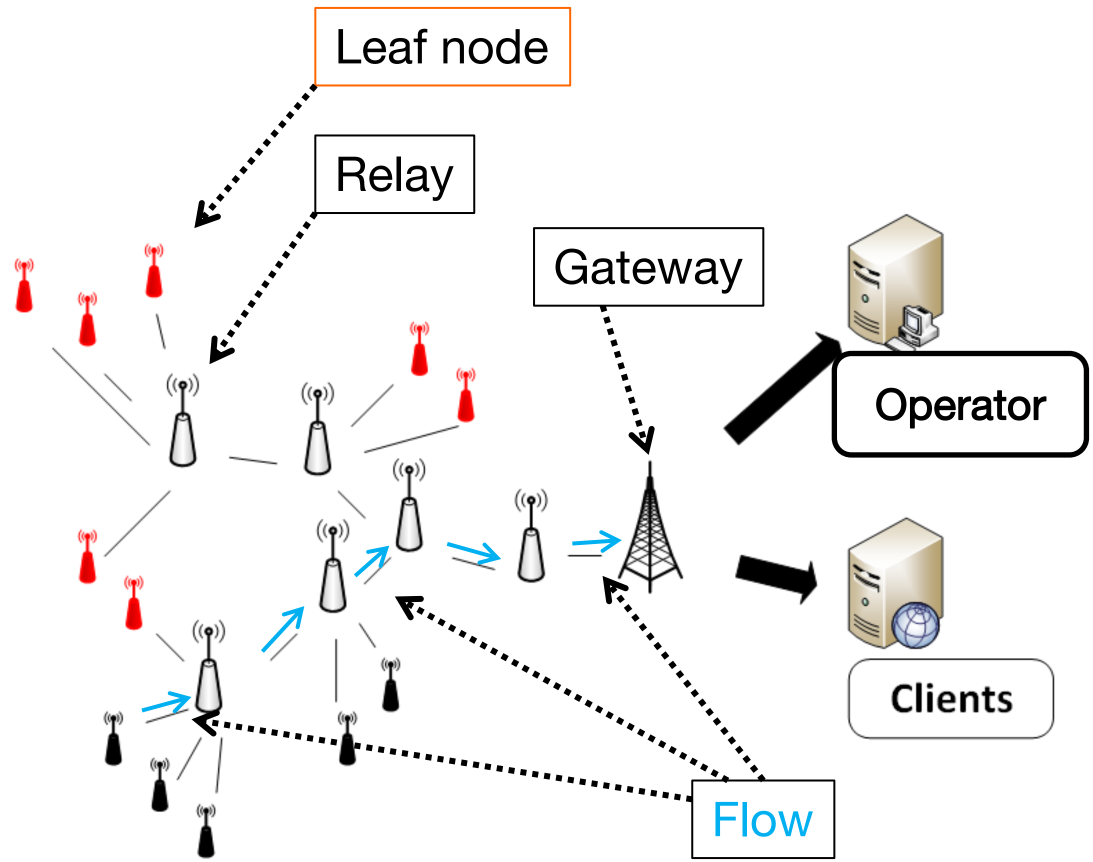
From the SLAs to the need of resource allocation

- Two Key Performance Indicators (KPIs) :
  - **End-to-end Packet Delivery Ratio (PDR)**
  - **End-to-end Delay**

=> Traffic Generation:
- for each leaf node
- for each application
- on a given period

=> QoS Requirements:
- app. specific
- sources - gateway(s)

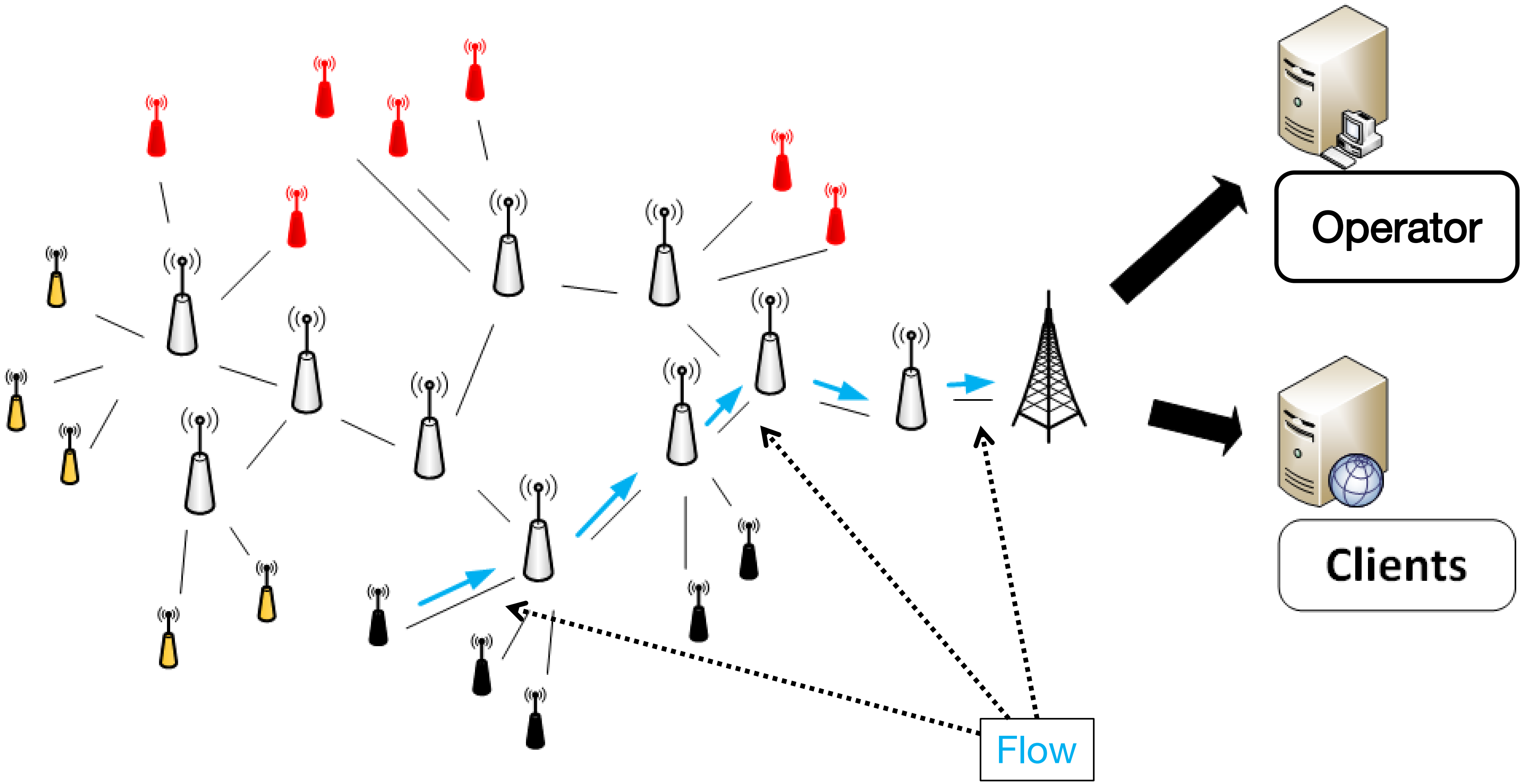
  => flows



"Spectrum Requirements for Short Range Device, Metropolitan Mesh Machine Networks (M3N) and Smart Metering (SM) applications," ETSI Tech. Report, Sep. 2011.

Challenge of Multi-fow Network Operation

- Flow prioritization ?
- Route selection ?
- Link resource allocation ?
- Load balancing

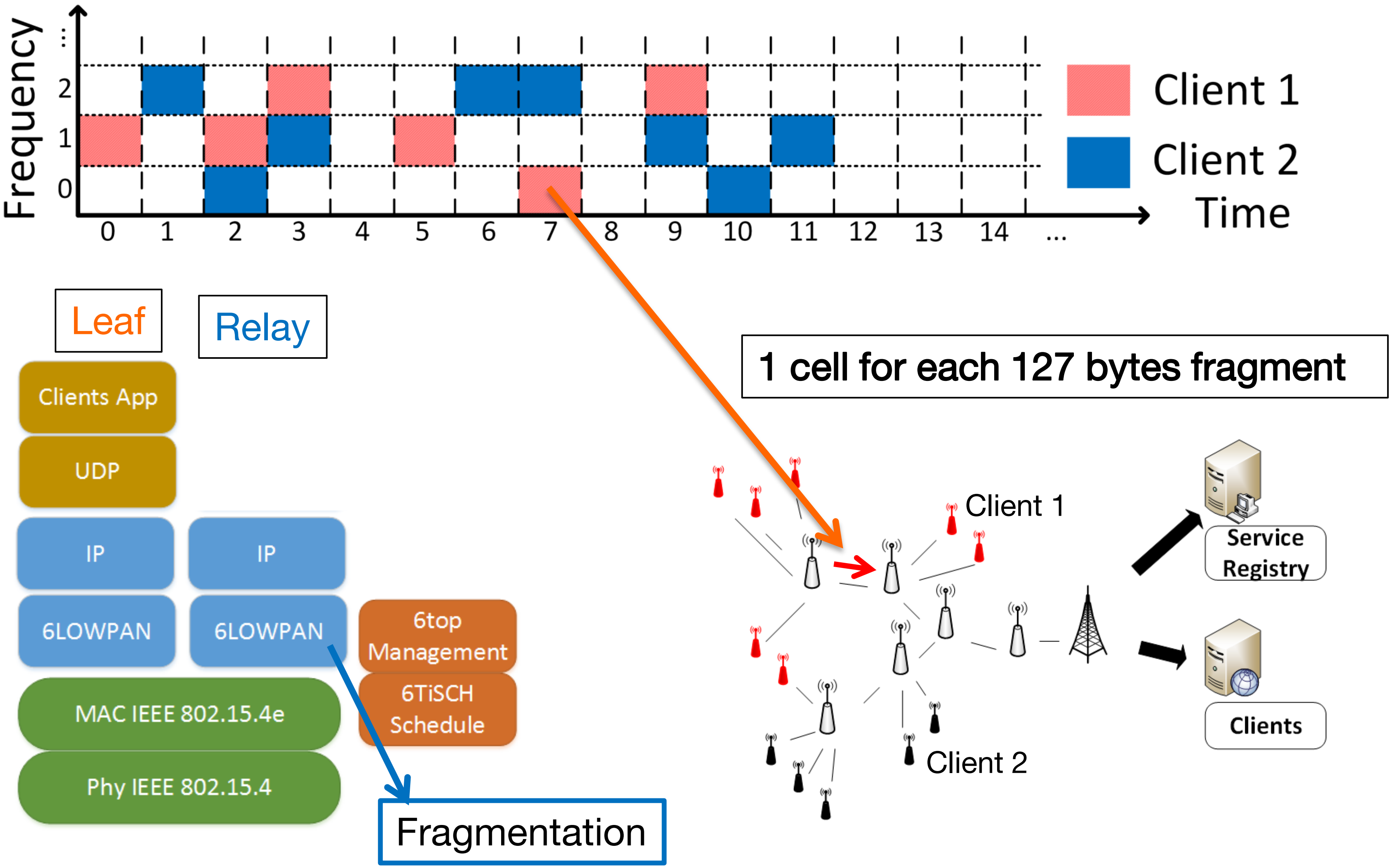


preserve network capacity
serving multi-flow & multi-KPI applications

Contributions

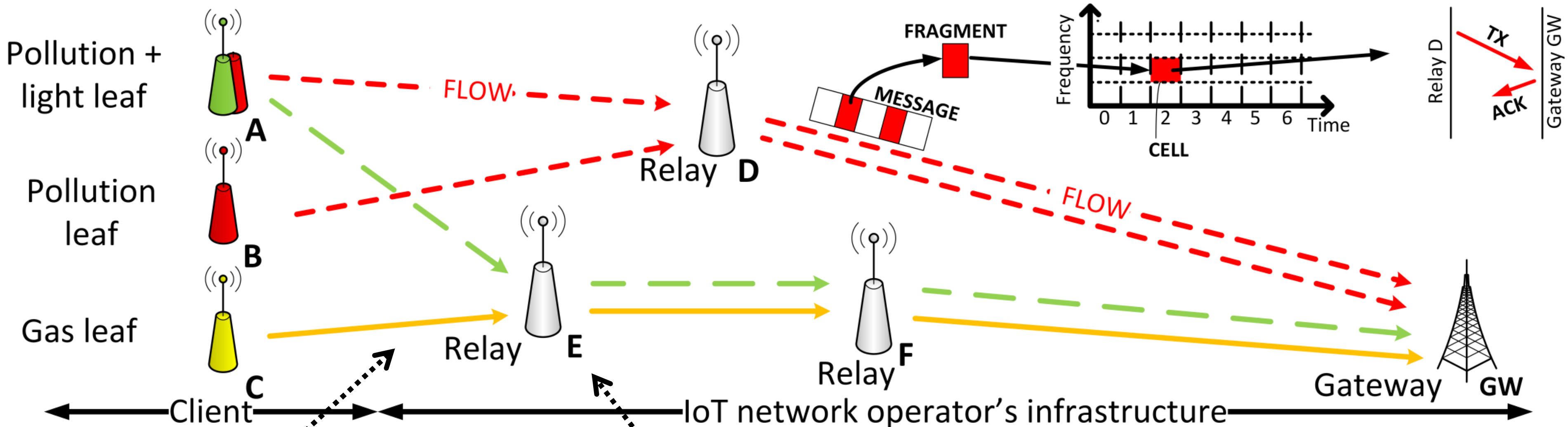
- Provide Load-balanced and Reliable Routing :
  - One path for each flow
- Propose KAUSA Scheduling Algorithm:
  - Provision resource for fragments & retransmissions
  - Satisfy flow PDR and delay
  - Makes use of Skip, Shift, Backtracking techniques
- Our results : SLA Satisfaction, Allocation Effectiveness
  - Verification by simulation

Background: the 6TiSCH protocol stack



From end-to-end data payload to single-hop fragments

Flow-level allocation constraints



Link
- Cells ?
- Allocation load ?
- Packet Error Rate ?
- Conflict graph ?

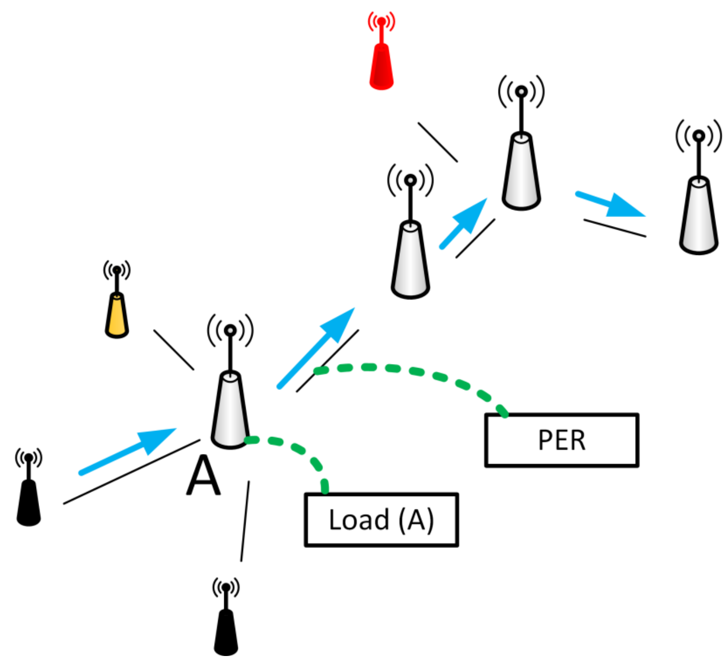
Node
- Forwarding plane ?
- Buffer occupation ?

- Interferences
- Half-duplex
- Application-specific traffic period
- Message versus Fragments

Contrib. 1: Load balanced and Reliable Routing

- Assumptions:
  - Neighborhood
  - Packet Error Rate (PER) of each link
  - Already allocated load

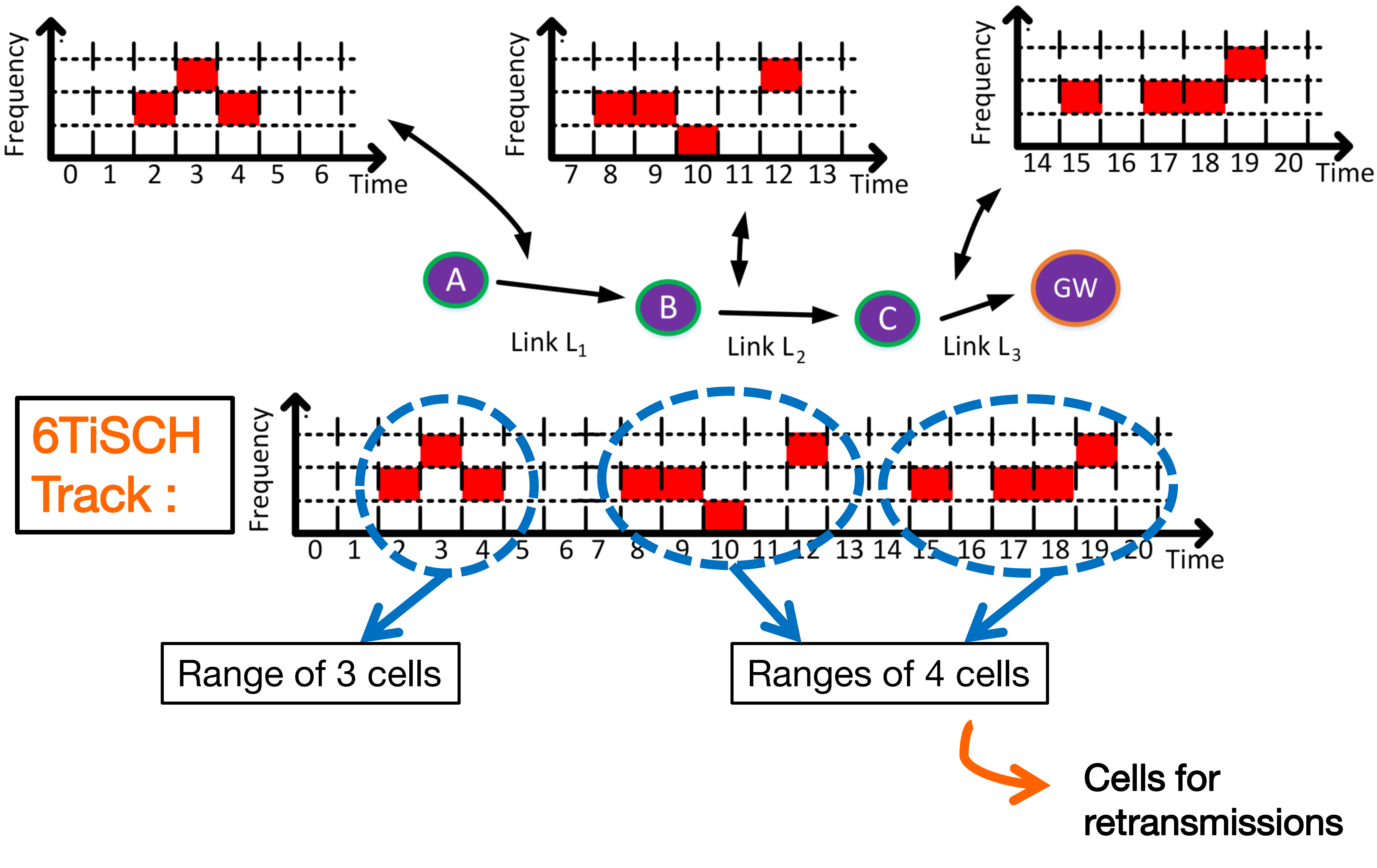
- Construction of **route** metric:
  - 1° Maximum node load
  - 2° Sum of load
  - 3° Sum of Expected Transmission Count (ETX)



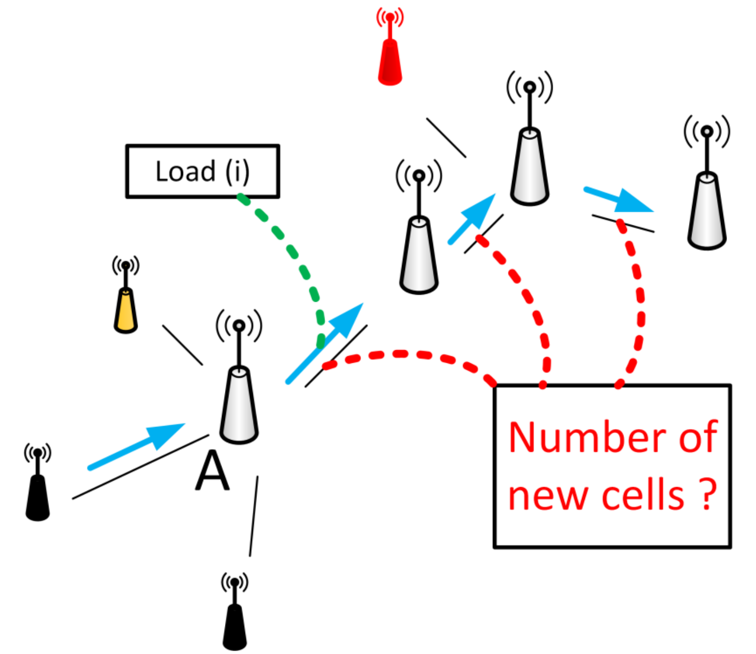
=> load-balancing & reliability

Forming tracks and per-flow allocation

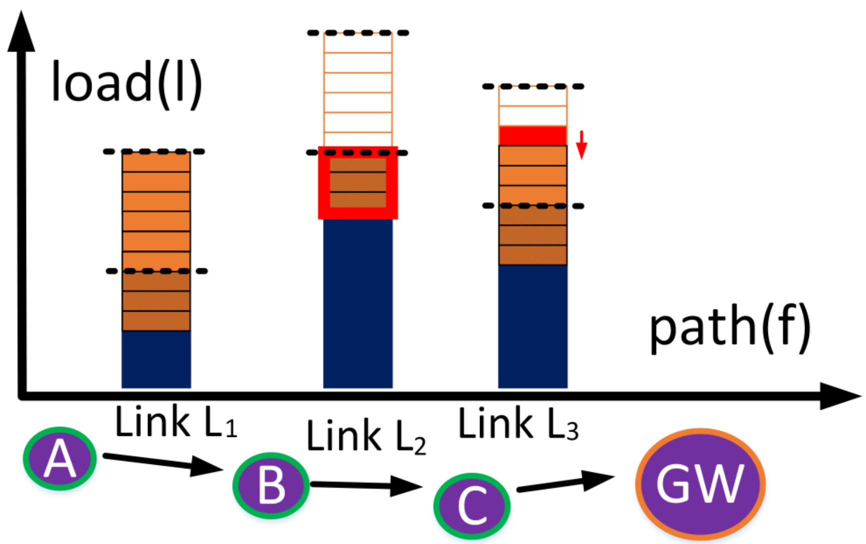
e.g. 1 message of 3 fragments



Contrib. 2: hop-by-hop cell provisioning including retransmissions

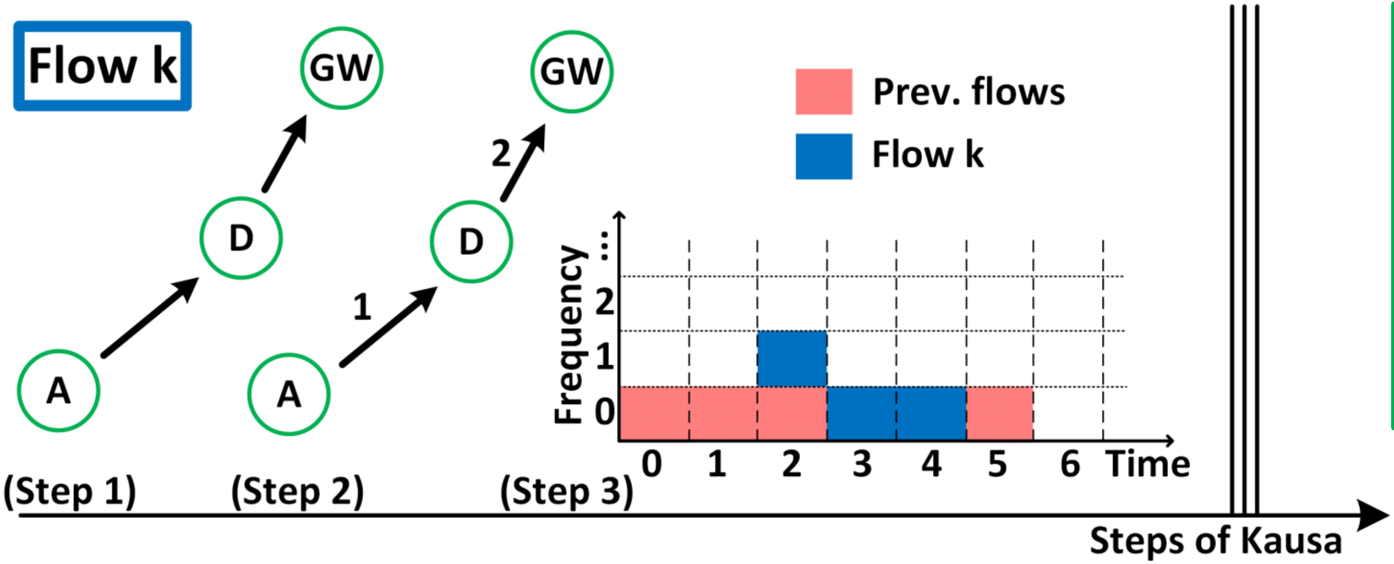


- Assumptions:
  - Flow path
  - Packet Error Rates (PERs)
  - Already allocated load
  - Flow traffic
- Calculation of hop cell count:
  - Greedy algorithm
  - Decrease from max. number of retransmissions
- Objectives :
  - PDR constraint satisfaction
  - Minimize maximum load



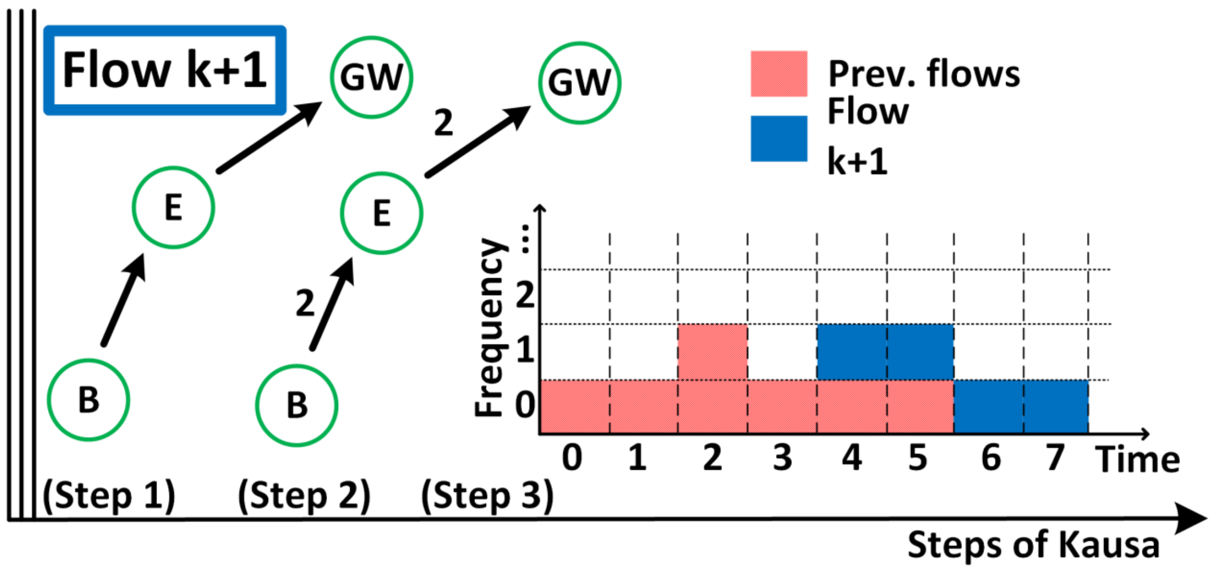
Enabling Flow level Reliability on FTDMA Schedules with efficient Hop-by-hop Overprovisioning," INRIA Grenoble Rhône-Alpes, RR-8866, 2016.

Steps of KAUSA



- **Step 0**:
  Order flows by constraint
- **Step 1**:
  Build path

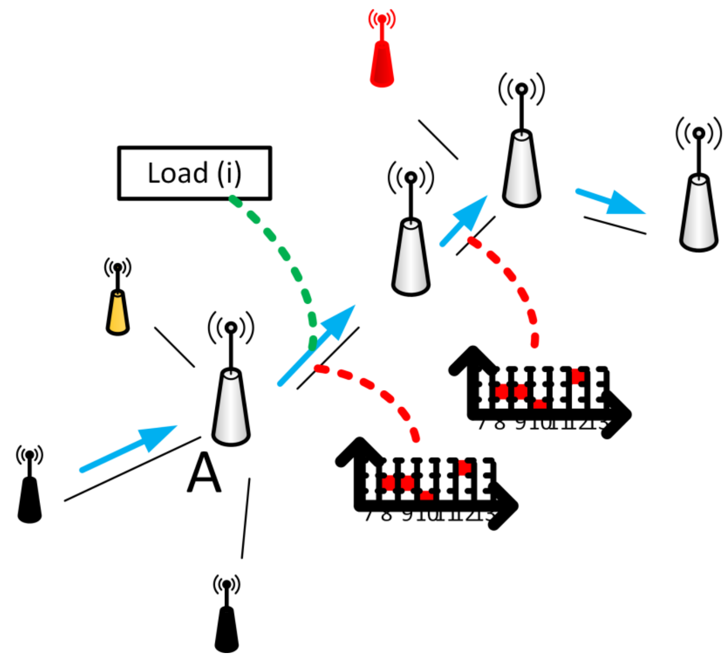
- **Step 2**:
  Compute (re)transmission cells
- **Step 3**:
  Allocate Cells
  - by message
  - by range



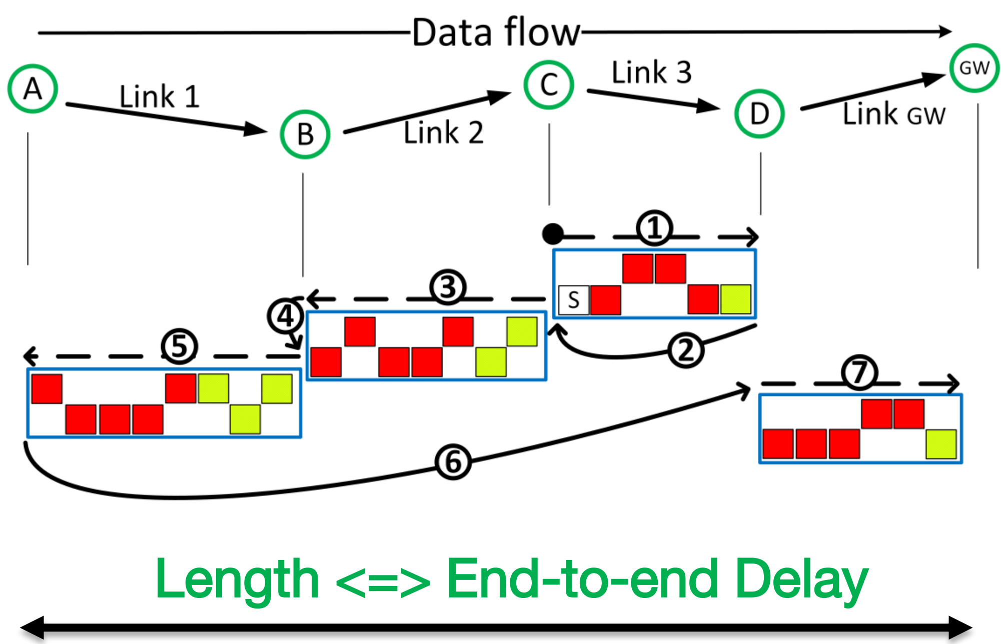
Contrib. 3: Allocation by ranges

Satisfying the delay with KAUSA

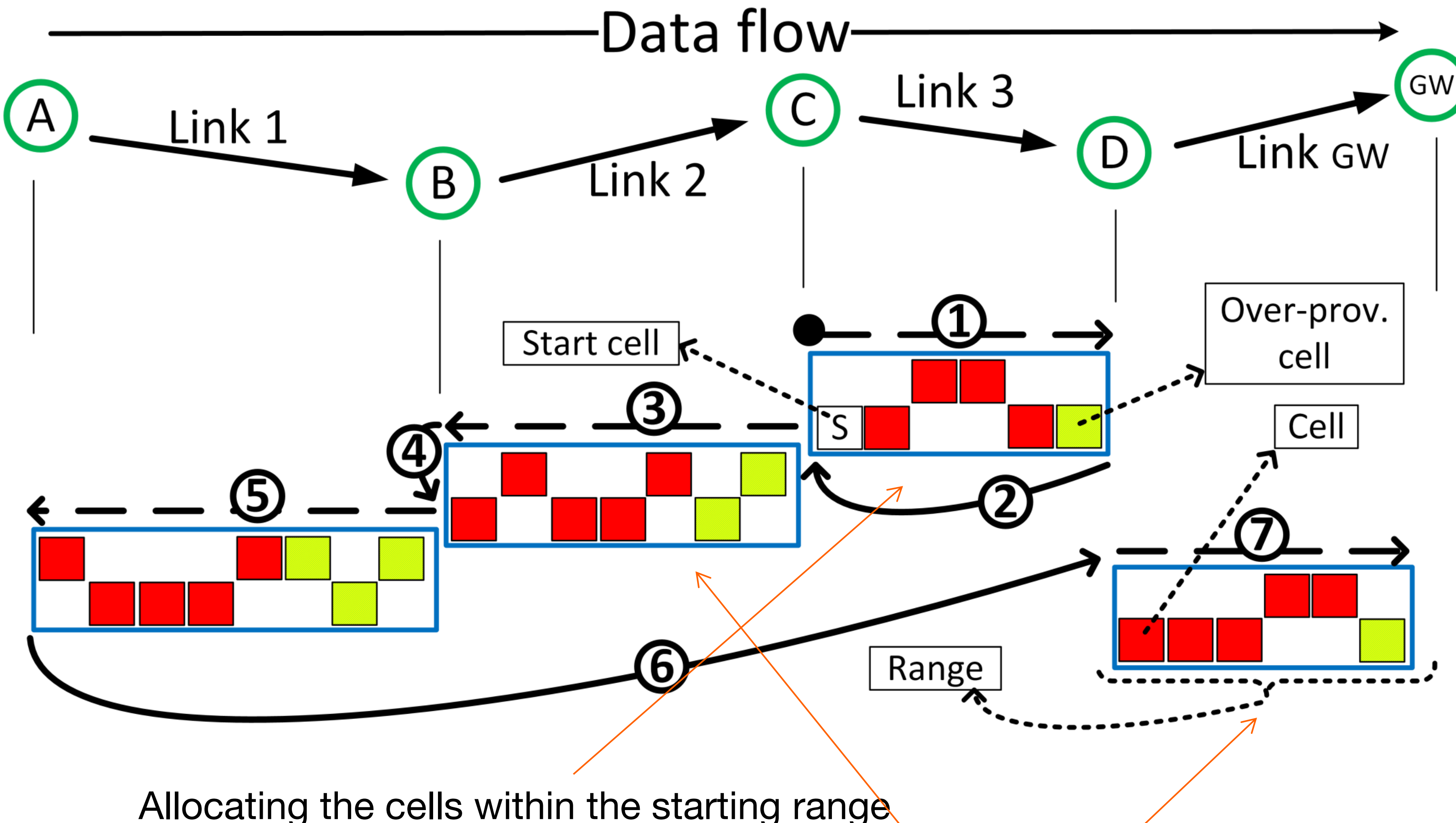
- Assumptions:
  - Flow path
  - hop cell counts
  - conflicts of each link
  - Already allocated cells



- Cell Allocation:
  - range by range
  - on most loaded link first



Cell allocation by range, for one message of a flow



Allocating the cells within the starting range

Allocating the cells for the previous hops

Allocating the cells within the following hops

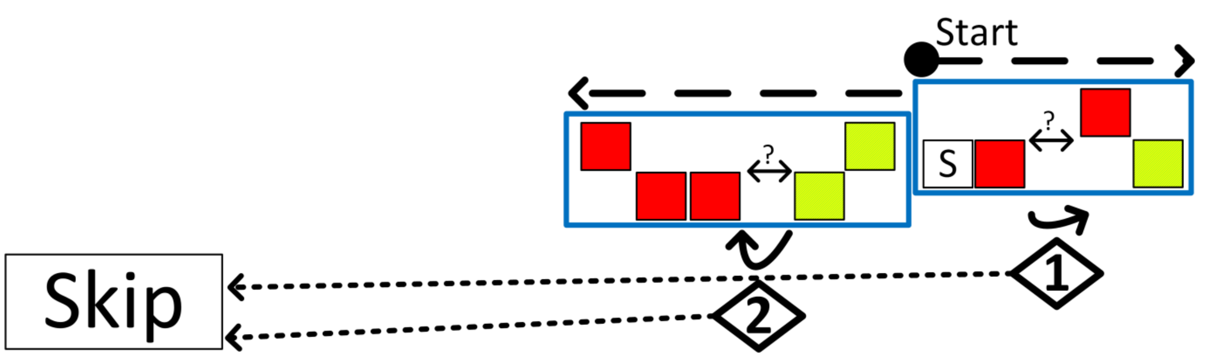
Modifying the starting range if necessary

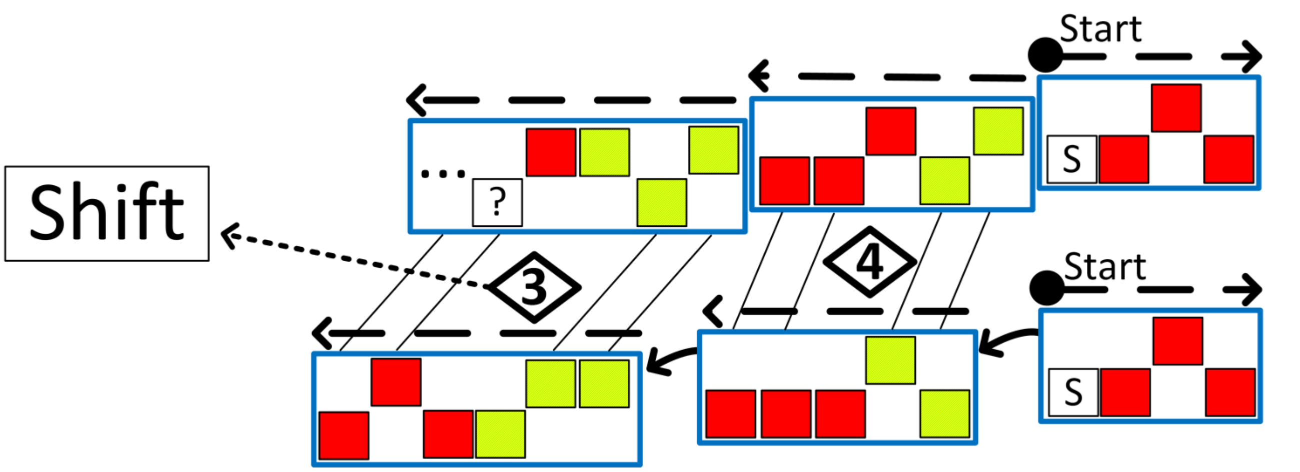
Problems during the allocation

- Problem on a given cell :
  - Buffer
  - Conflict
  - Busy

  => Skip the cell
  => Shift the range

- Delay no longer satisfied
  - Backtracking
    - Link level
    - Flow level
  - Failure





Evaluation Scenario

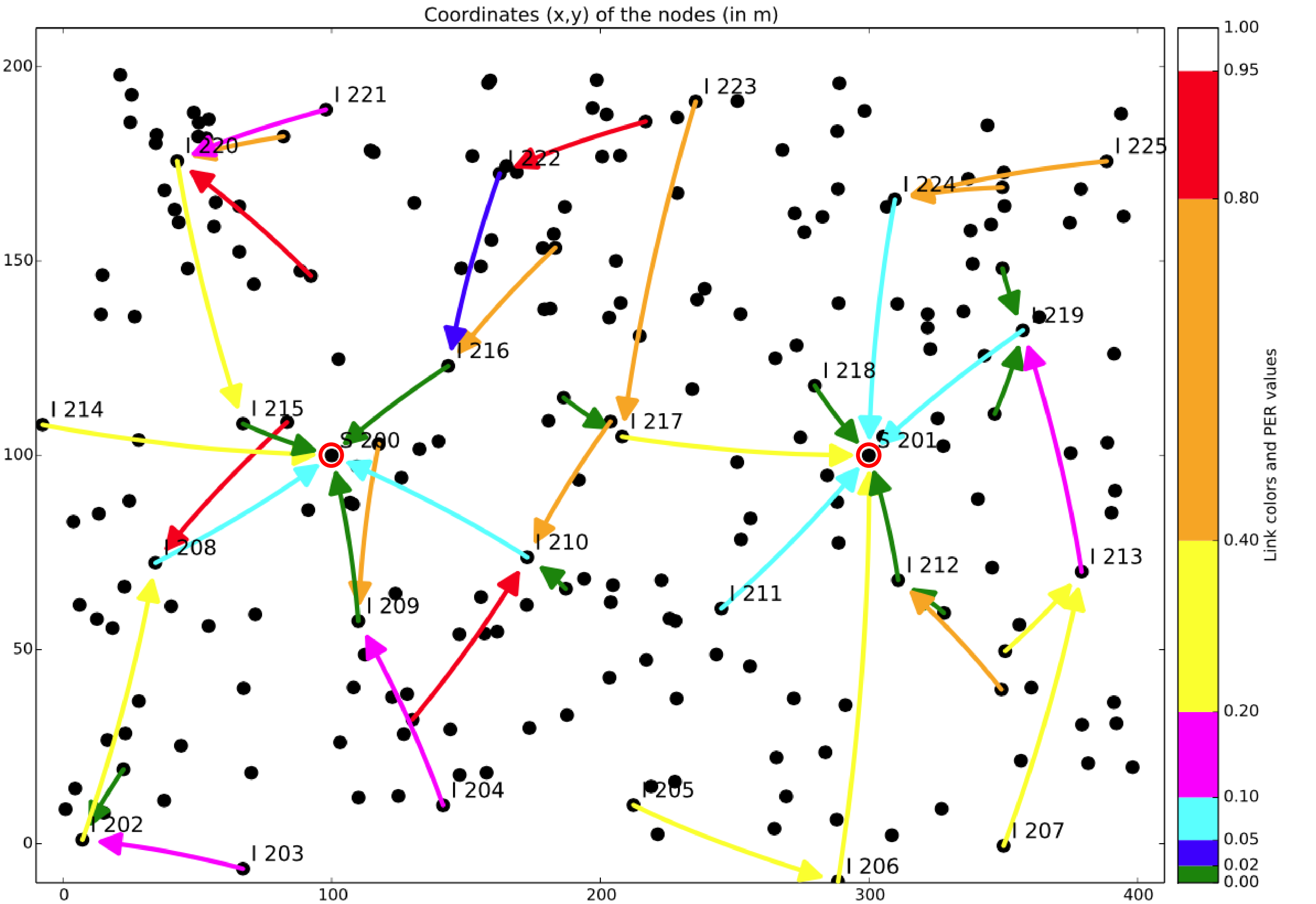
=> Simulation:
- Monte Carlo simulations (Python)
- Periodical traffic pattern (bounded arrivals)
- Comparison with TASA + retransmissions

=> 2 applications:
- Strong delay constraint: 60 slots
- Strong PDR constraint: 0.97

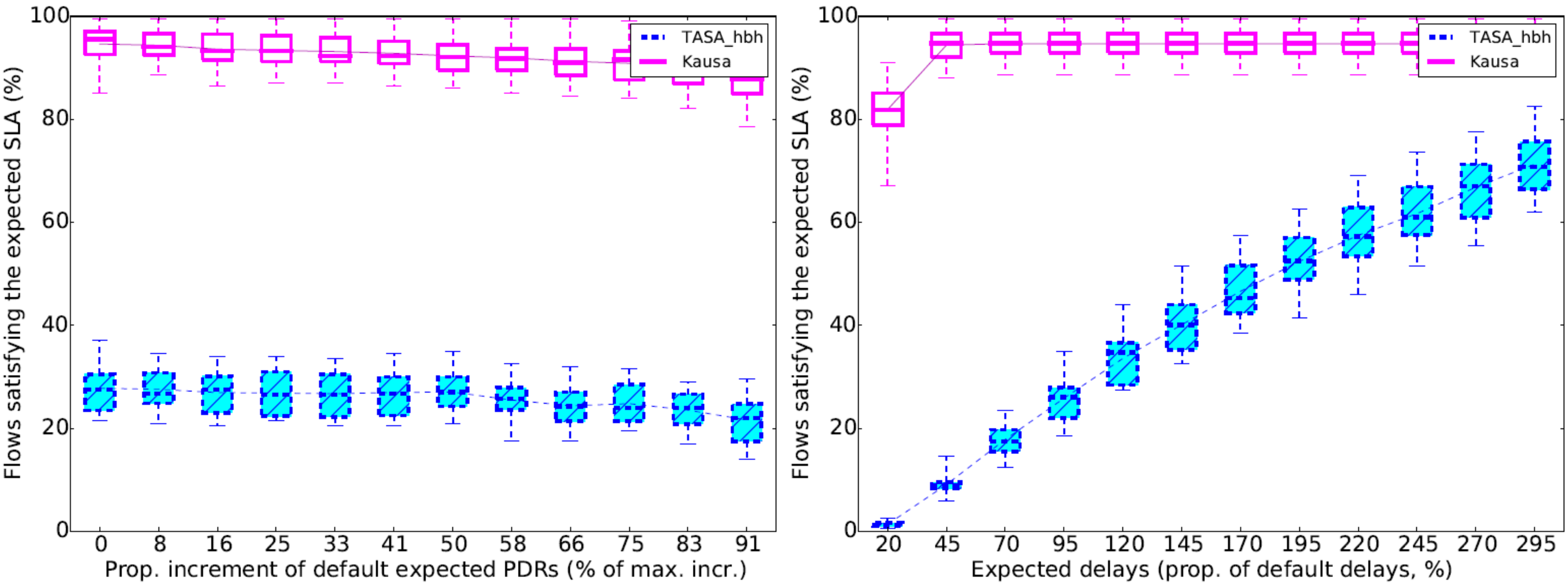
=> Evaluation on a 400x200 m topology:
- Number of nodes: 200 leaves, 24 relays, 2 GW
- 16 topology instances: uniform (leaf), triangle mesh (relay)
- IEEE 802.15.4 16 channels
- PER: Path-loss, time-invariant, link type
- Max buffer: 20 fragments

=> Variation of 4 parameters:
- Traffic intensity, slotframe size, expected PDR, expected delay
- Results : SLA satisfaction, allocation load, max buffer occupation

Topology example with a set of paths built on ETX

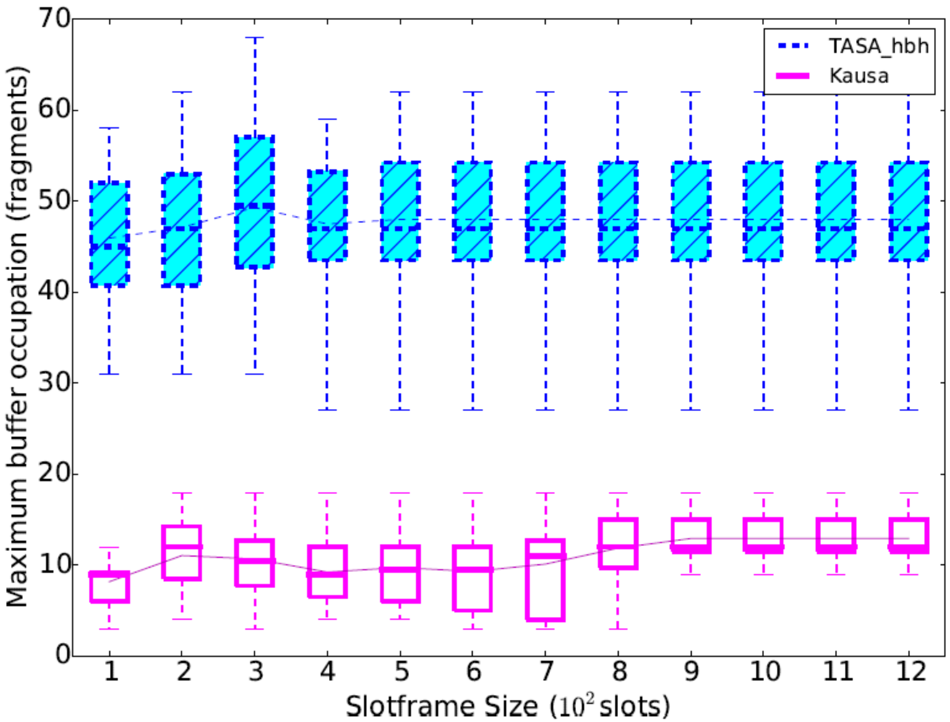


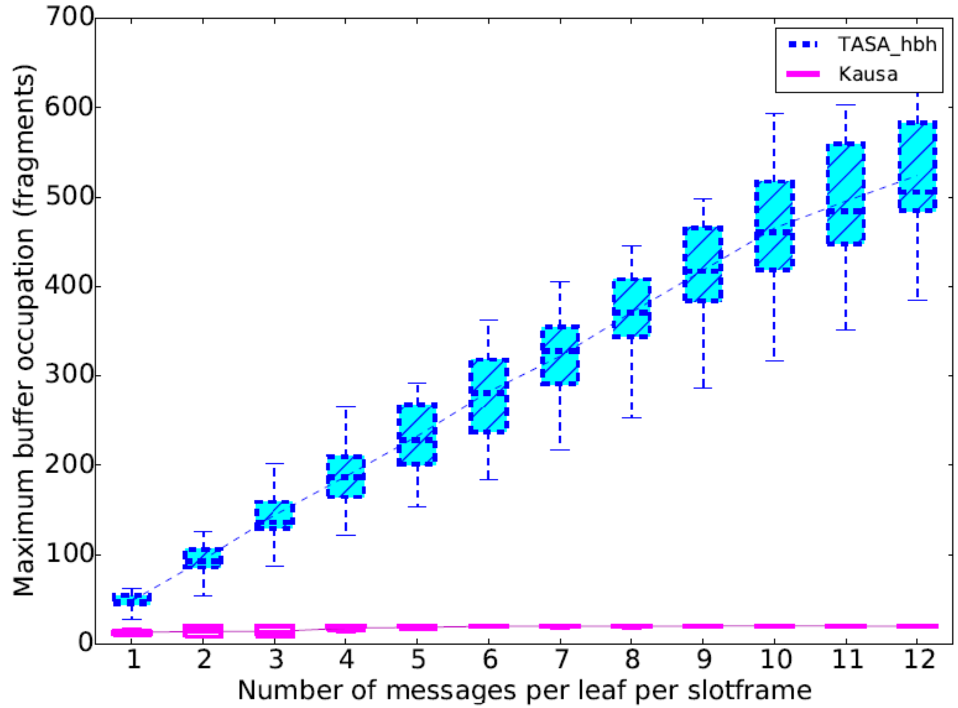
Flow satisfying the SLA



KAUSA satisfies strong PDR an delay constraints

Maximum buffer occupation





KAUSA maintains limited buffer occupation

Conclusion and perspectives

- Flow-level resource allocation enables wireless multi-hop network **operation**
- KAUSA efficiently realizes central allocation of FTDMA resources:
  - Satisfaction of PDR an Delay
  - Load balancing
  - Multi-flow
  - Buffer occupation
- Comparison with extended TASA shows performance improvement
- **Perspectives**:
  - Experimentation: FIT IoT-Lab, SensOrLabs test-beds
  - Standardization: IETF 6TiSCH
  - Extension: dynamic or bursty traffic

Questions ?

guillaume.gaillard1@insa-lyon.fr

INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES LYON

citi lab

UrbaNet

inria informatiques mathématiques

iCUBE

orange™

Thanks