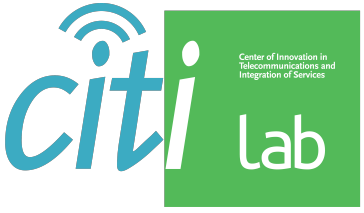
Formal Verification of Real-Time Wireless Sensor Networks Protocols: Scaling Up

Alexandre Mouradian
Isabelle Augé-Blum

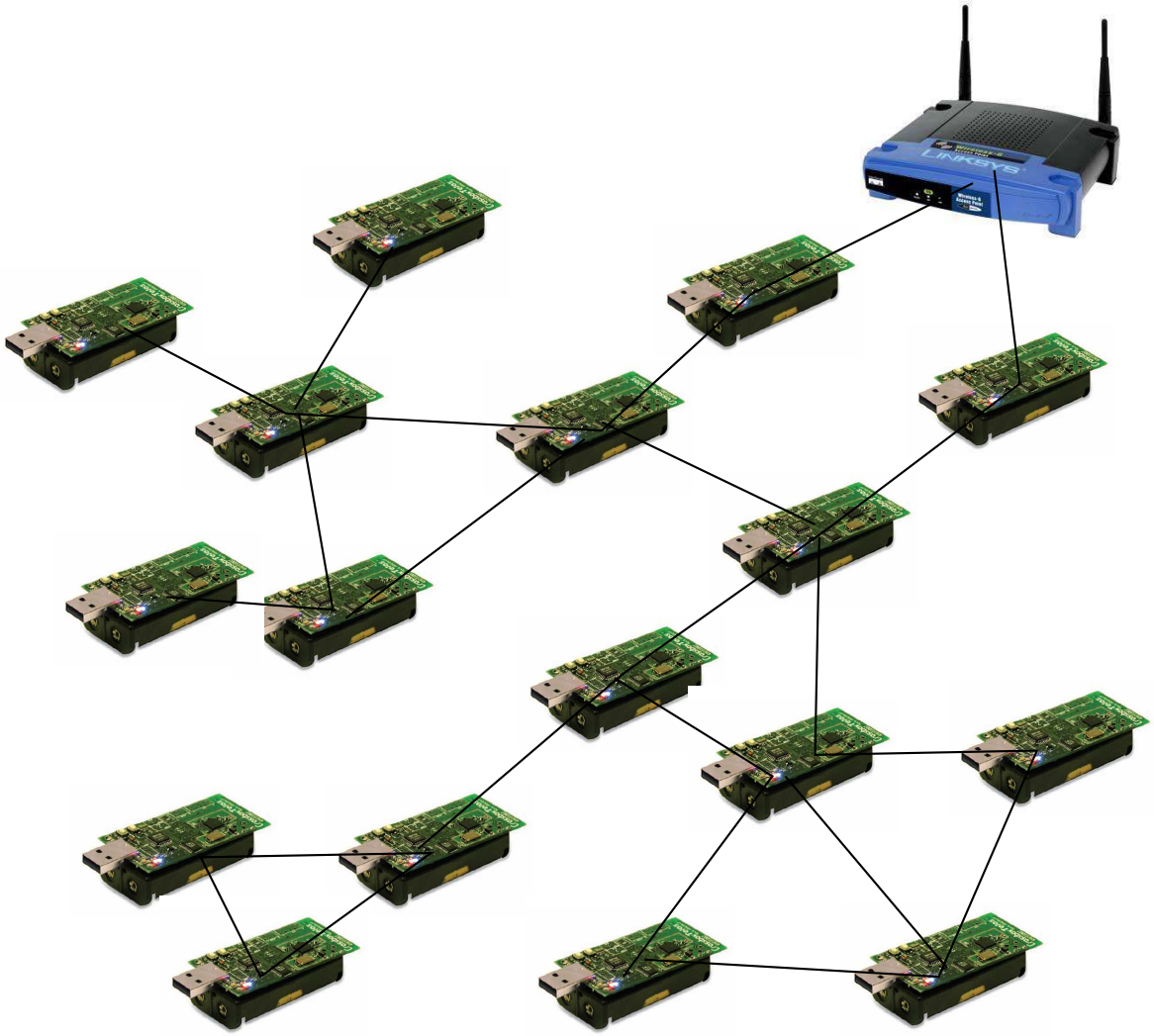


ECRTS

9th July 2014

Context

Wireless Sensor Networks



Constraints :

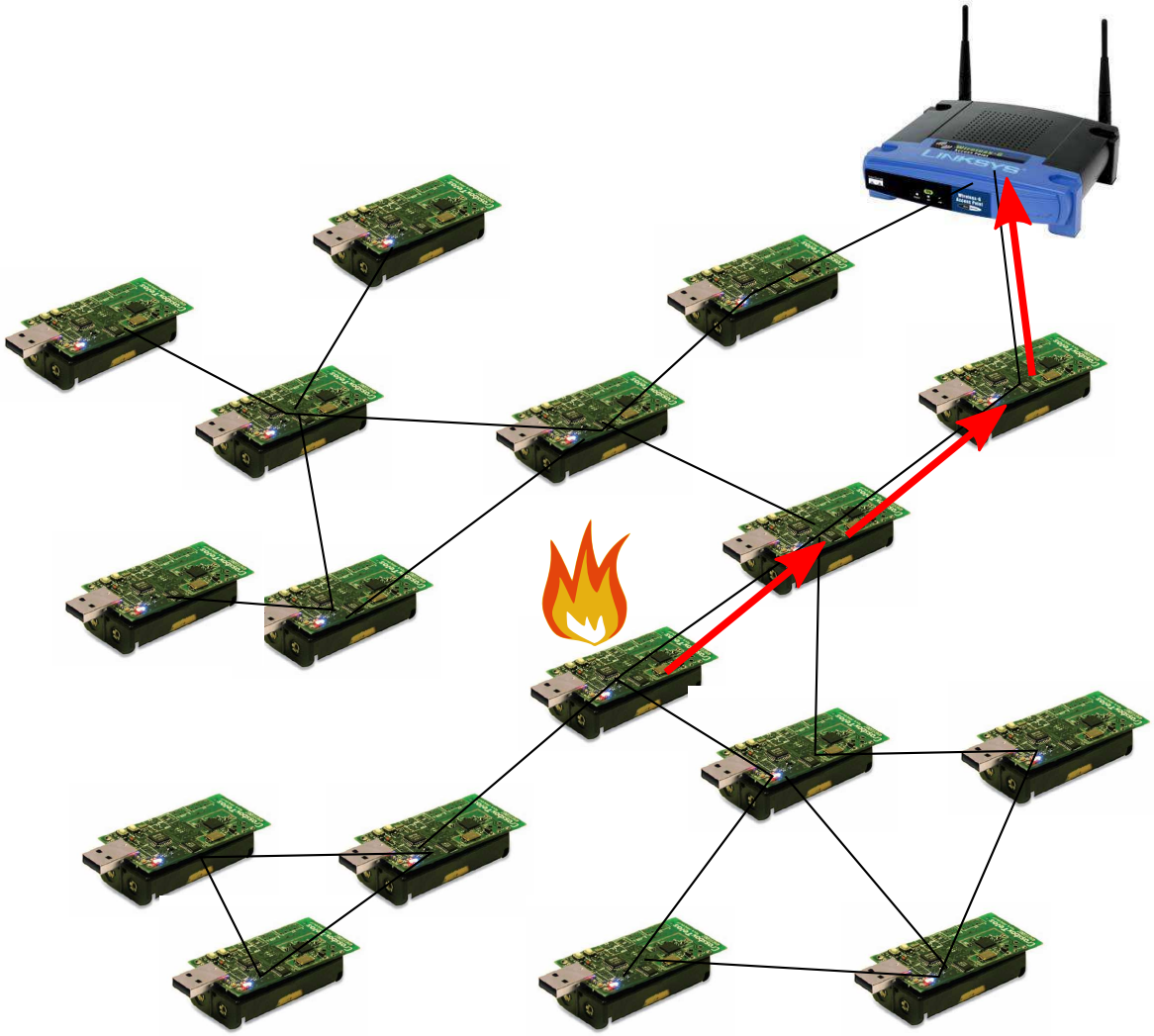
- Limited hardware capabilities
- No fixed infrastructure
- Unreliable Links

Goals :

- Energy efficiency
- Self-organization
- Reliability
- Scalability
- Constrained delays

Context

Wireless Sensor Networks



Constraints :

- Limited hardware capabilities
- No fixed infrastructure
- Unreliable Links

Goals :

- Energy efficiency
- Self-organization
- Reliability
- Scalability
- Constrained delays

Problematic

Critical WSN applications

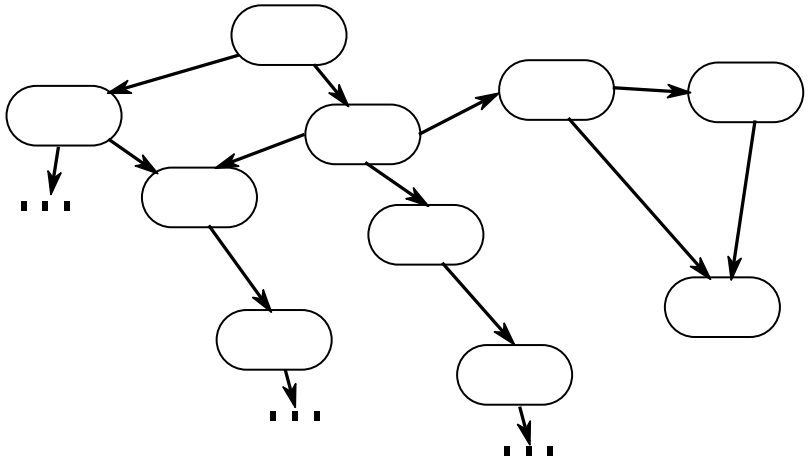
How to guarantee end-to-end delays in WSNs ?

↳ Formal Methods

The goal of this work is to adapt timed formal verification to WSNs

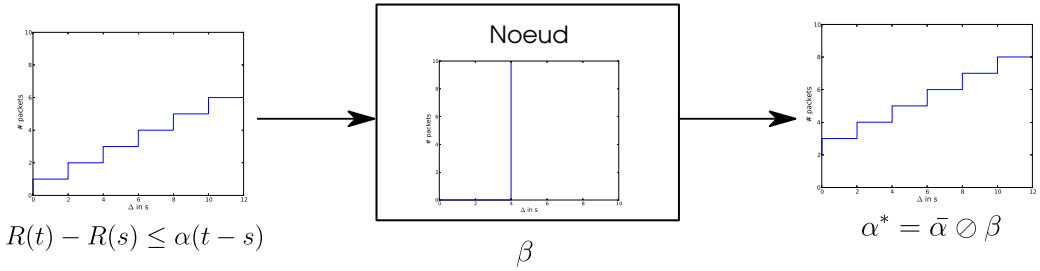
Formal verification of real-time properties

Model Checking



Explores all the possible behaviors of a model of the system, BUT combinatorial explosion

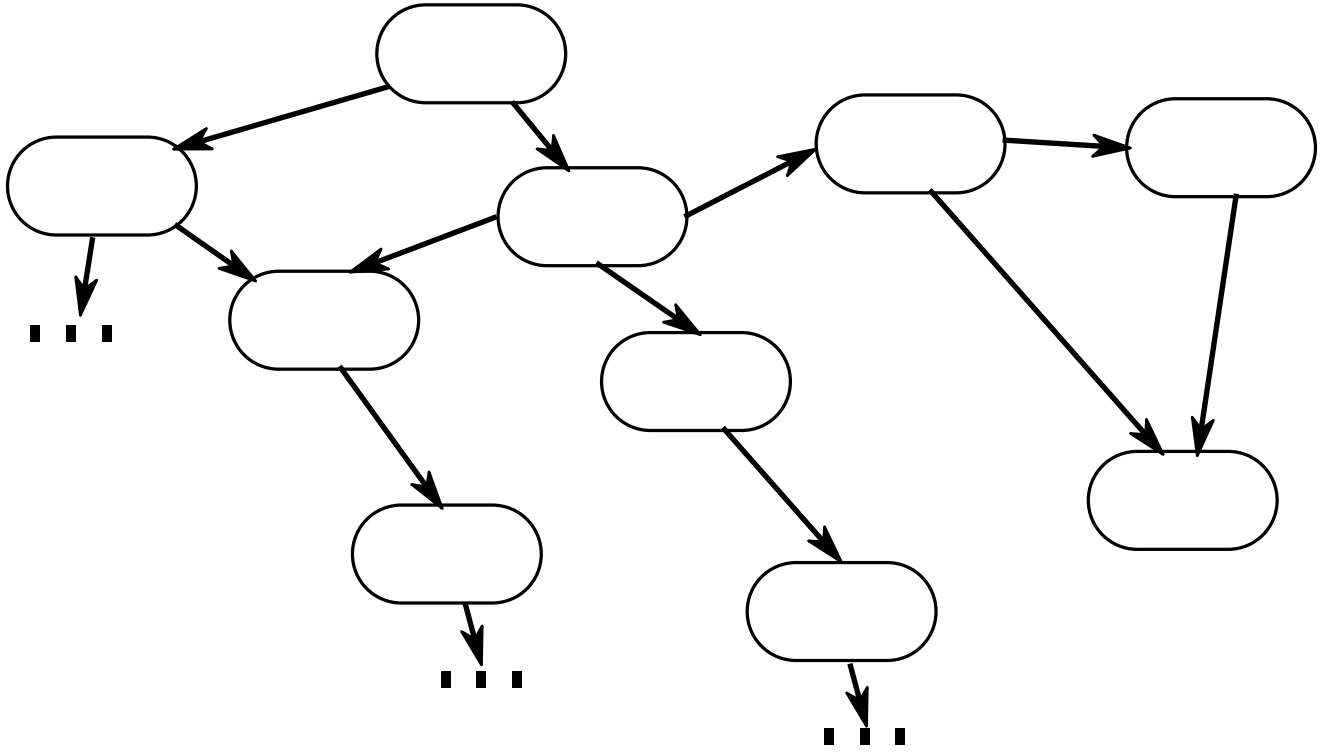
Network Calculus

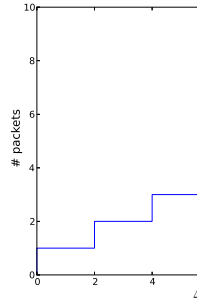


Abstraction of the behavior with composable functions

Allows to work on large scale systems, BUT abstraction not proven

Model Checking



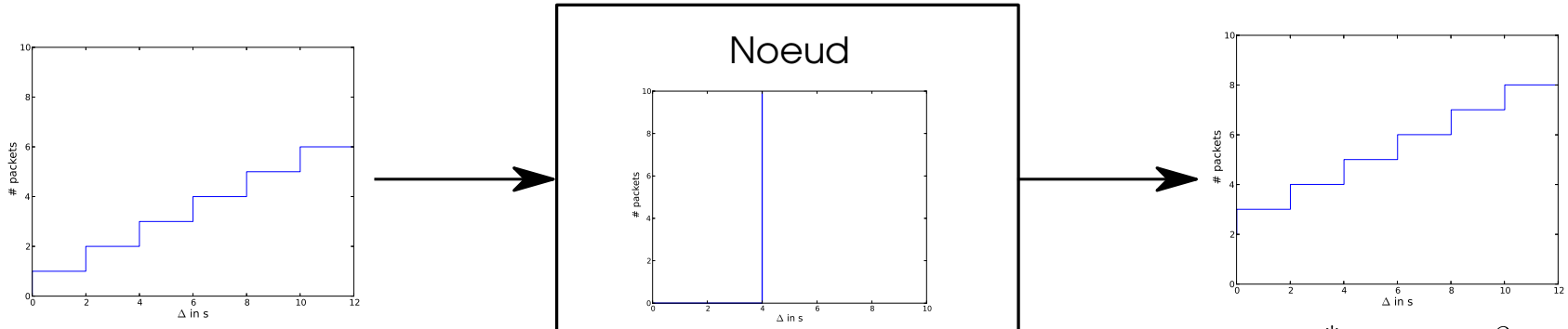


$R(t) - R(s$

Explores all the possible behaviors of a model of the system, BUT combinatorial explosion

Allo

Network Calculus

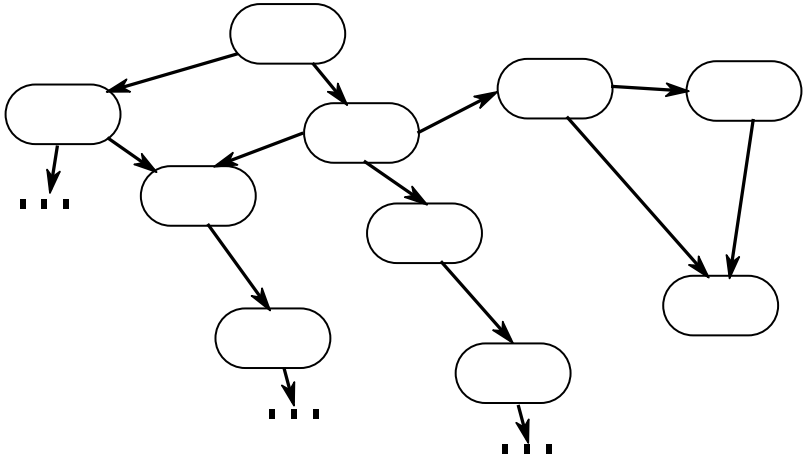


Abstraction of the behavior
with composable functions

Allows to work on large scale systems,
BUT abstraction not proven

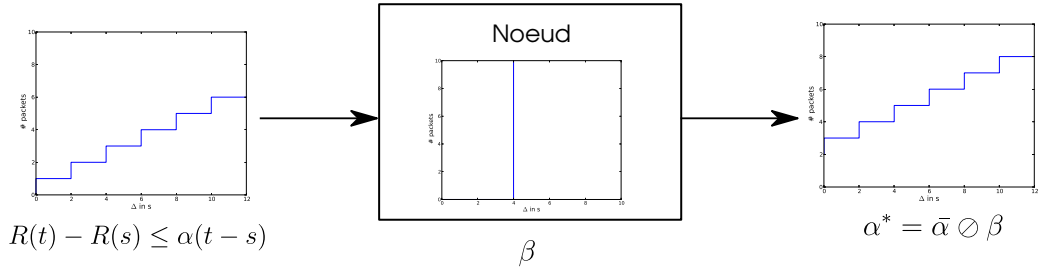
Formal verification of real-time properties

Model Checking



Explores all the possible behaviors of a model of the system, BUT combinatorial explosion

Network Calculus

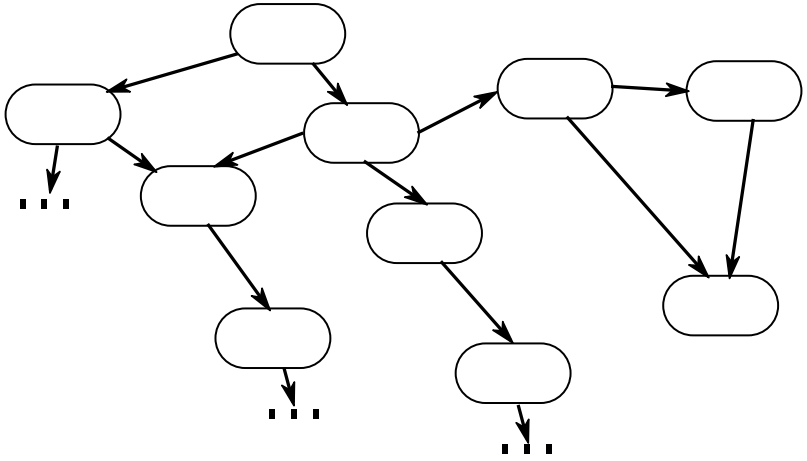


Abstraction of the behavior with composable functions

Allows to work on large scale systems, BUT abstraction not proven

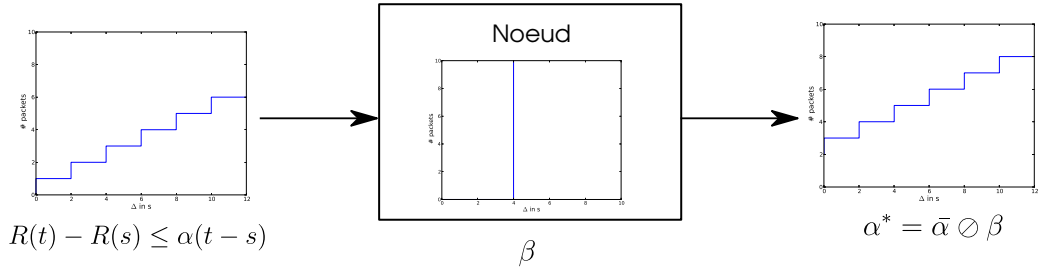
Formal verification of real-time properties

Model Checking



Explores all the possible behaviors of a model of the system, BUT combinatorial explosion

Network Calculus



Abstraction of the behavior with composable functions

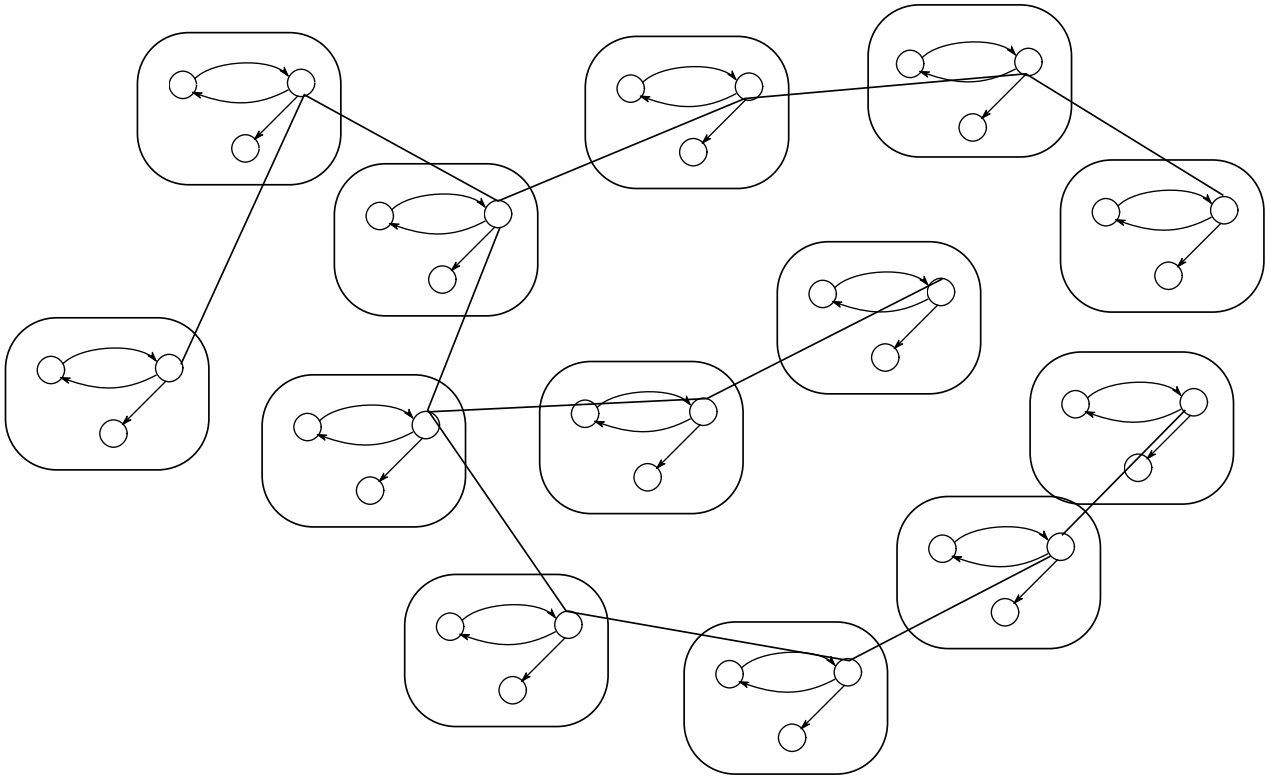
Allows to work on large scale systems, BUT abstraction not proven

Model Checking seems more convincing at first glance but less applicable to realistic WSNs

Timed Model Checking

The issue:

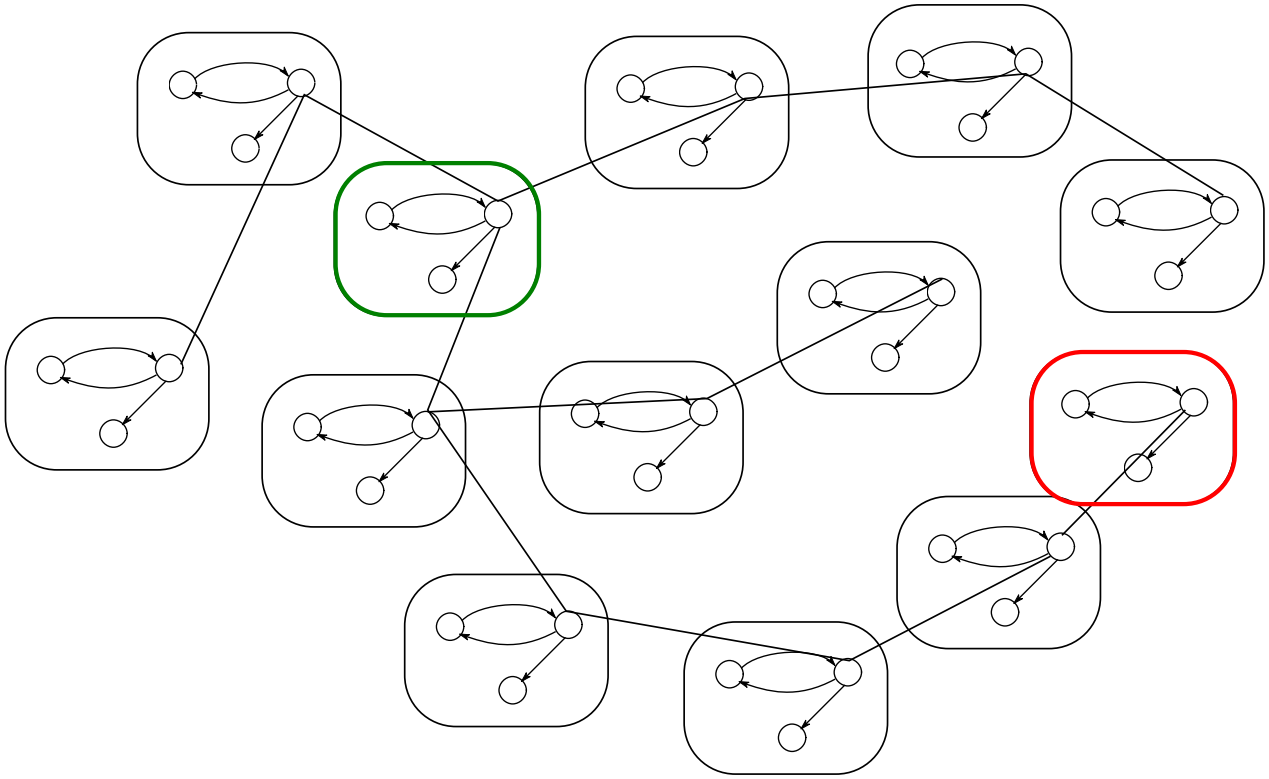
- A node is represented with a Timed Automaton (with clocks and variables representing its internal state)
- The network is a composition of such automata
- The tree of executions of the network is exponential in the number of clocks and variables



Timed Model Checking

The issue:

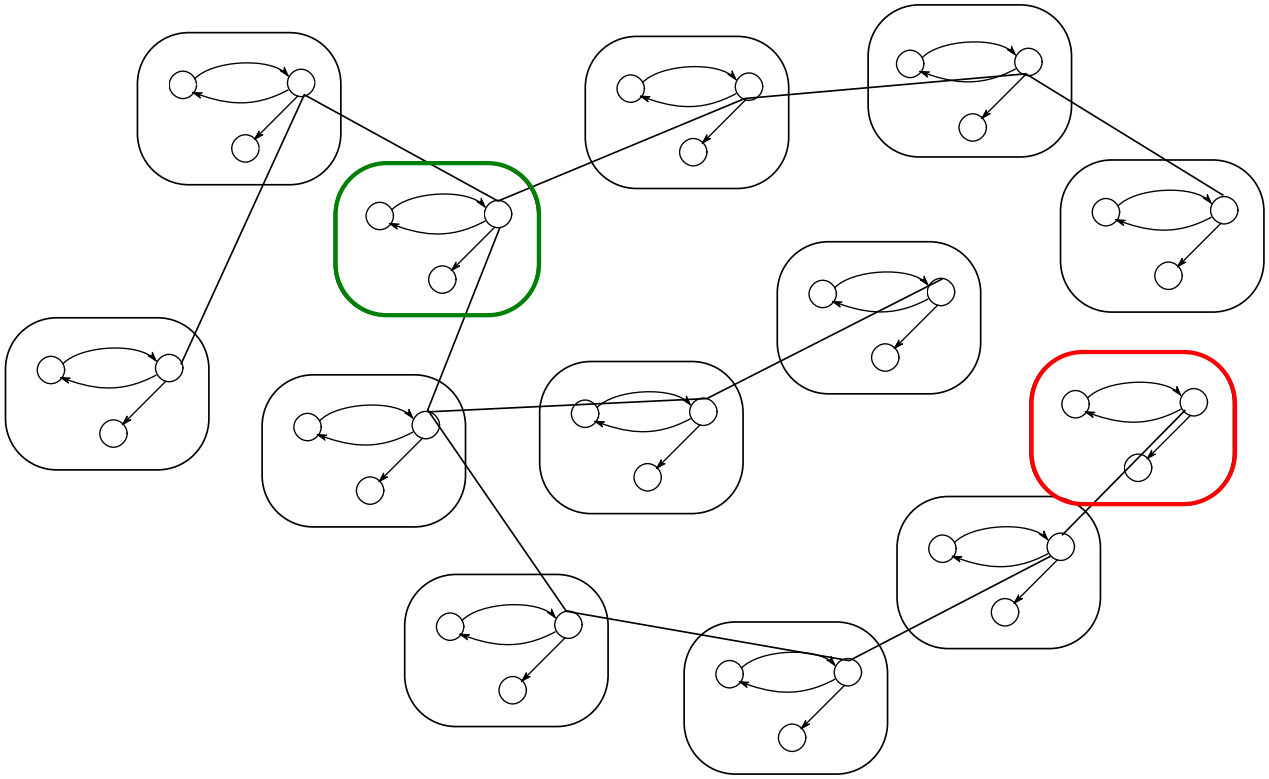
- A node is represented with a Timed Automaton (with clocks and variables representing its internal state)
- The network is a composition of such automata
- The tree of executions of the network is exponential in the number of clocks and variables



Timed Model Checking

The issue:

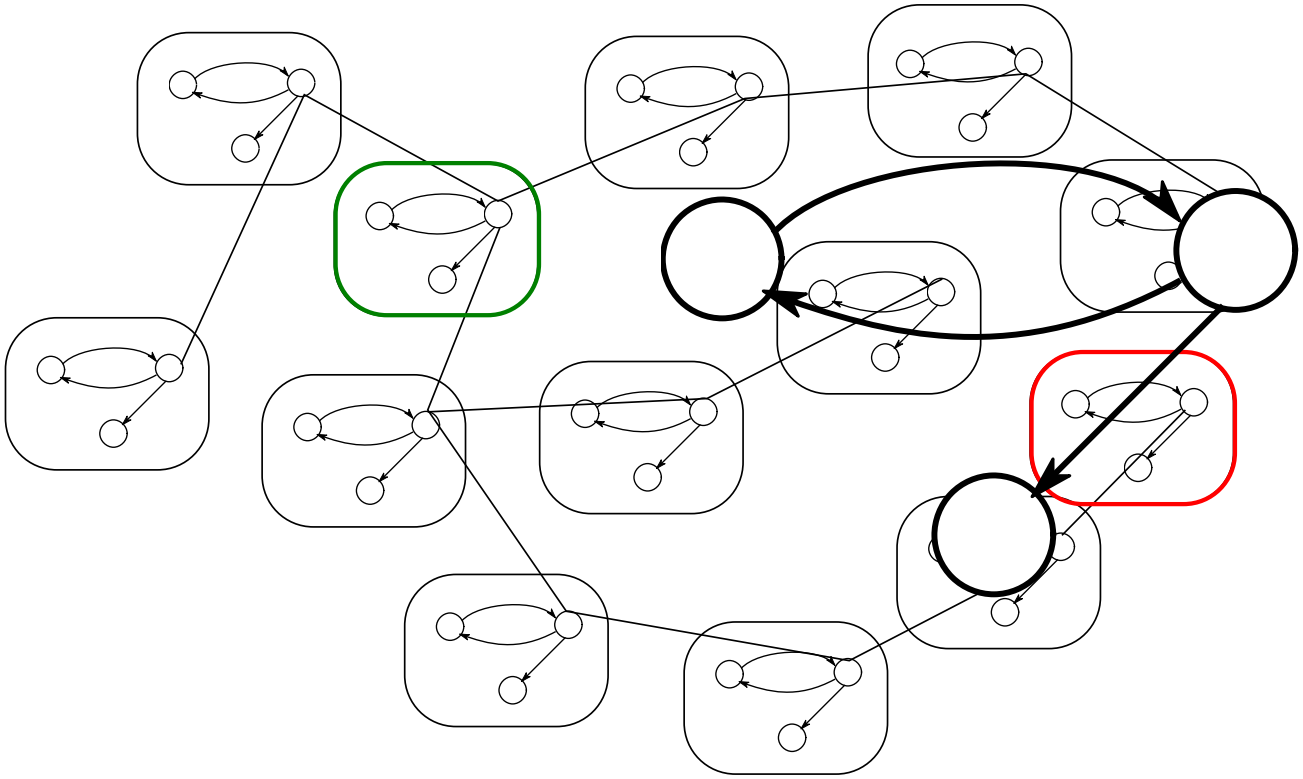
- A node is represented with a Timed Automaton (with clocks and variables representing its internal state)
- The network is a composition of such automata
- The tree of executions of the network is exponential in the number of clocks and variables



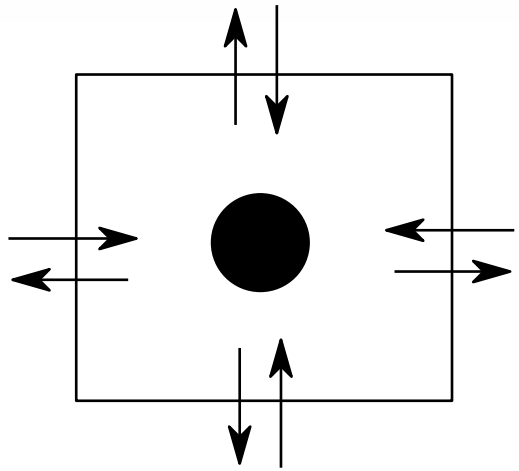
Timed Model Checking

The issue:

- A node is represented with a Timed Automaton (with clocks and variables representing its internal state)
- The network is a composition of such automata
- The tree of executions of the network is exponential in the number of clocks and variables



Overview of the scheme



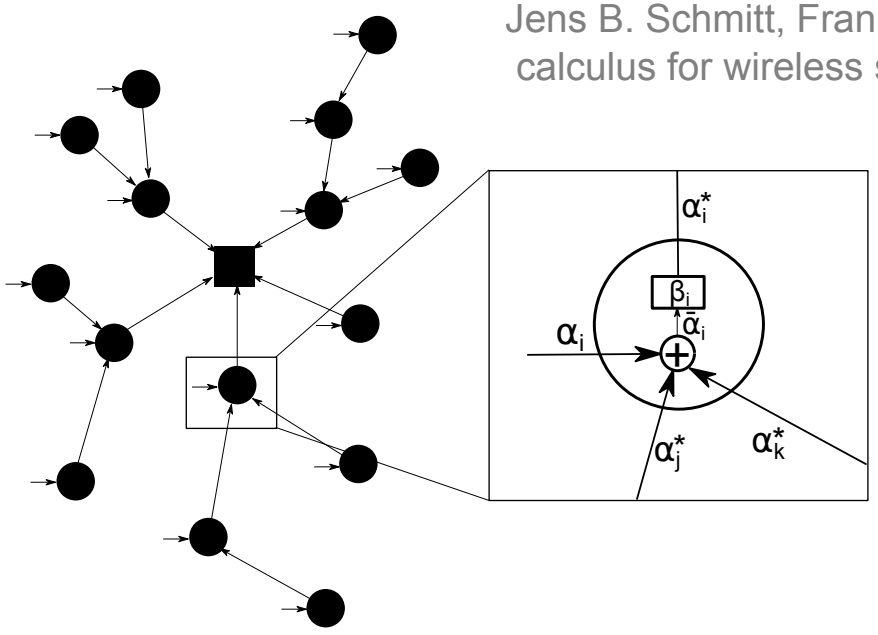
For each node:

1- Express the interactions of each node with the rest of the network: Network Calculus

2- Verify that the node can actually deal with these interactions in bounded time: Model Checking

Sensor Network Calculus

Jens B. Schmitt, Frank A. Zdarsky, and Lothar Thiele. "A comprehensive worst-case calculus for wireless sensor networks with in-network processing." RTSS 2007.



$$\bar{\alpha}_i = \alpha_i + \sum_{j \in Ch(i)} \alpha_j^*$$

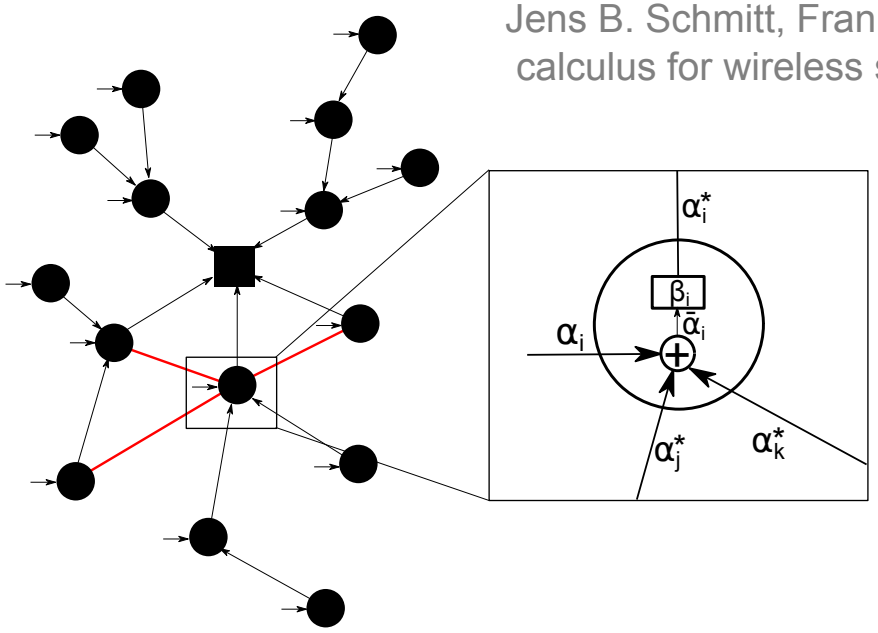
$$\alpha_i^* = \bar{\alpha}_i \oslash \beta_i = (\alpha_i + \sum_{j \in Ch(i)} \alpha_j^*) \oslash \beta_i$$

β_i is the service provided by the protocol

Sensor Network Calculus

Sensor Network Calculus

Jens B. Schmitt, Frank A. Zdarsky, and Lothar Thiele. "A comprehensive worst-case calculus for wireless sensor networks with in-network processing." RTSS 2007.



$$\bar{\alpha}_i = \alpha_i + \sum_{j \in Ch(i)} \alpha_j^*$$

$$\alpha_i^* = \bar{\alpha}_i \oslash \beta_i = (\alpha_i + \sum_{j \in Ch(i)} \alpha_j^*) \oslash \beta_i$$

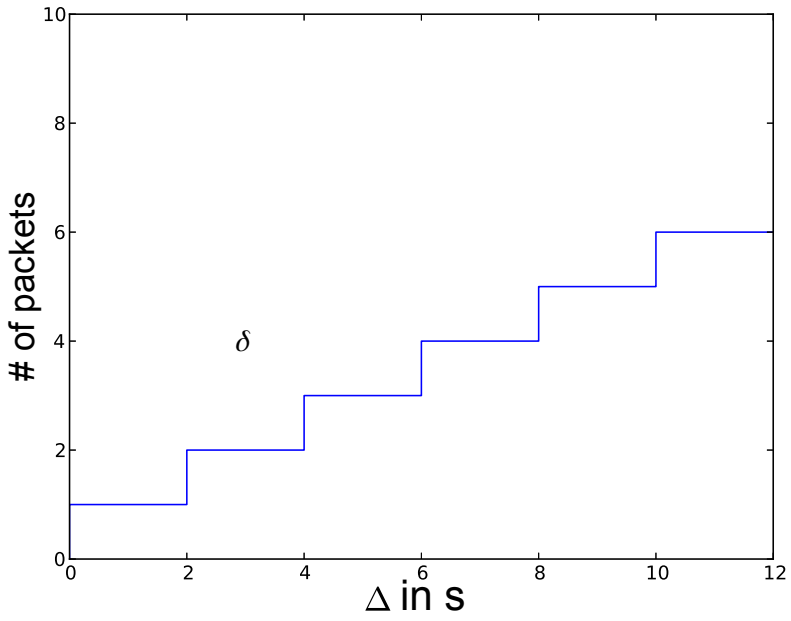
β_i is the service provided by the protocol

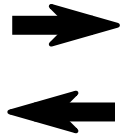
Adding medium access competitors :

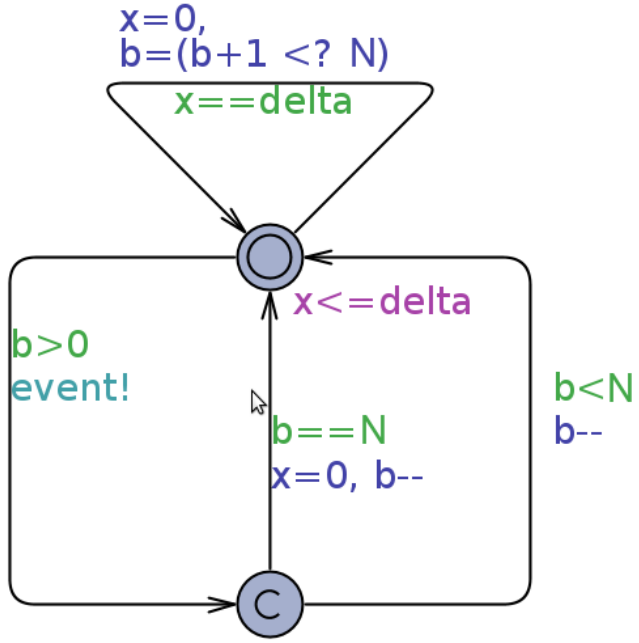
$$\alpha_i^c = \sum_{j \in Cp(i)} \alpha_j^*$$

From curves to automata

Kai Lampka, Simon Perathoner, and Lothar Thiele. "Analytic real-time analysis and timed automata: a hybrid method for analyzing embedded real-time systems." EMSOFT 2009.





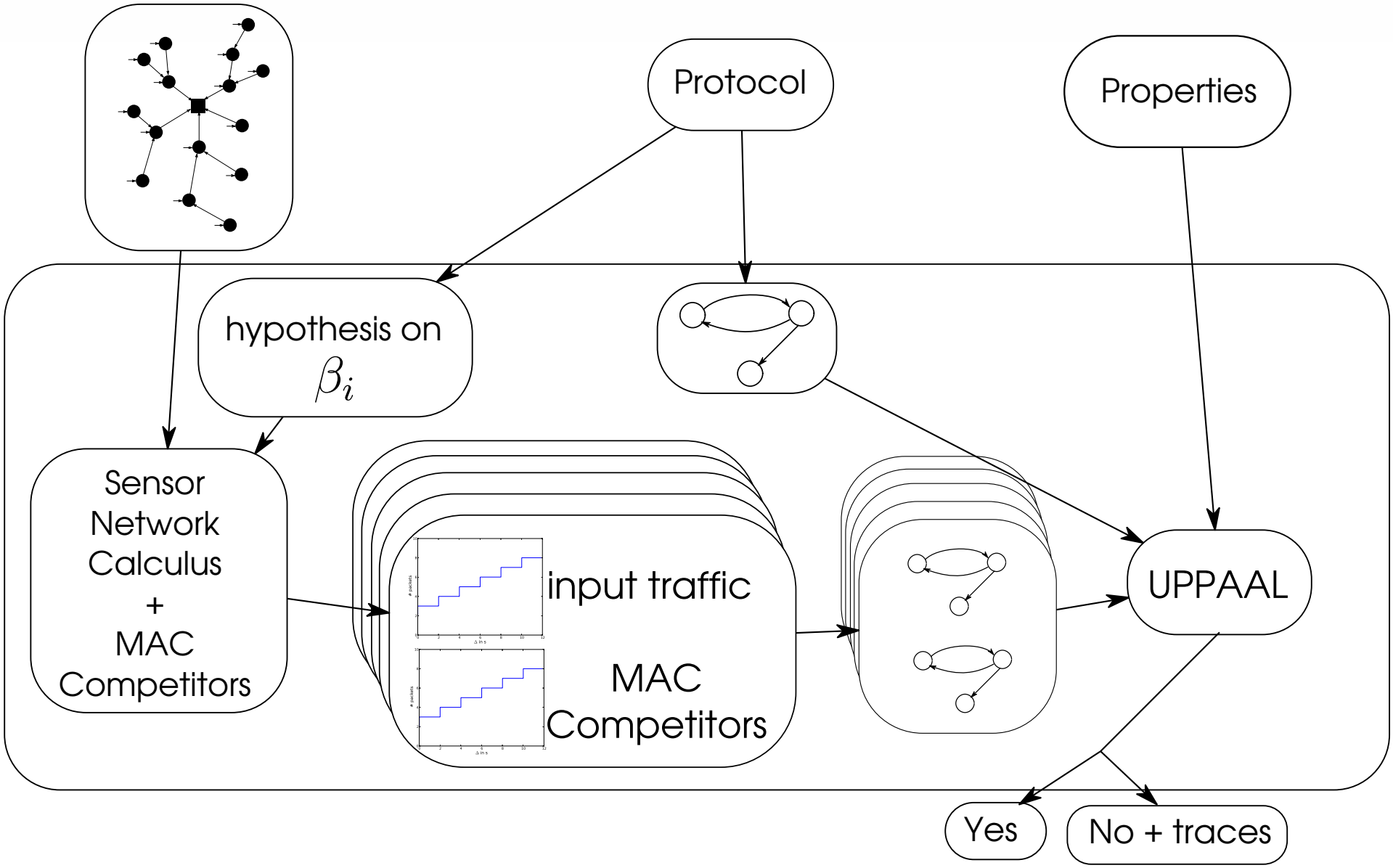


Global declarations:
broadcast chan event;

Local declarations:
clock x;
const int BMAX=N;
int[0,BMAX] b=0;
const int delta=δ;

$$\gamma_N(\Delta) = N + \lceil \frac{\Delta}{\delta} \rceil$$

Proposed verification algorithm



Application of the method

Application to RTXP, a distributed real-time protocol for WSNs

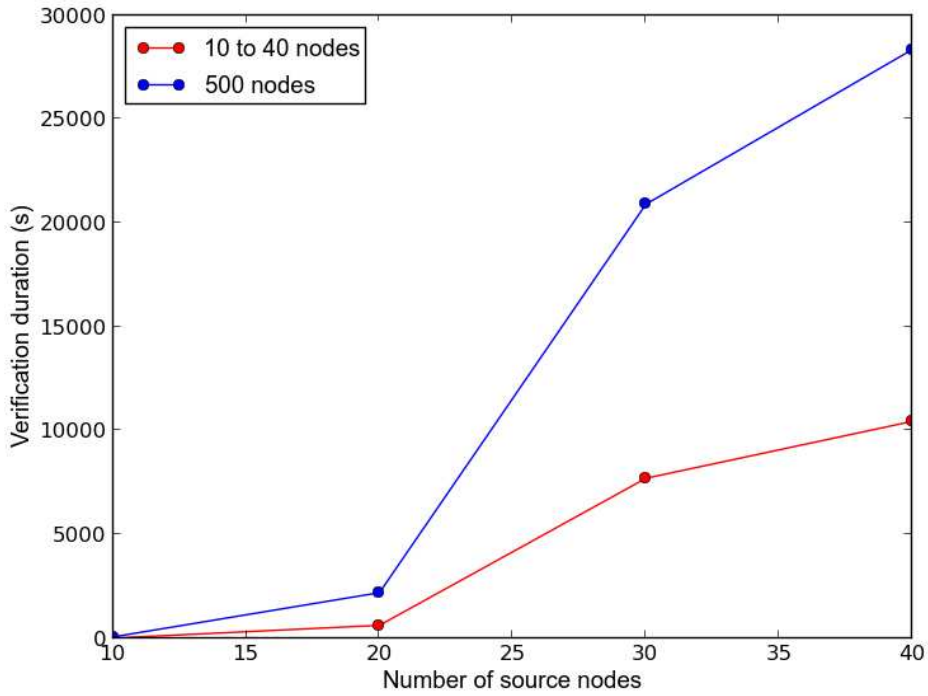
Alexandre Mouradian, Isabelle Augé-Blum, and Fabrice Valois. "RTXP: A localized real-time MAC-routing protocol for wireless sensor networks." Computer Networks 67 (2014): 43-59.

UPPAAL TA model for one node : ~ 30 states, ~ 40 transitions, 3 clocks

Random network graphs :

- 10 to 40 nodes topologies
- 500 nodes topologies
- Number of sources : 10 to 40

We observe that the real-time capacity of RTXP is exceeded with 40 sources.



Conclusion and perspectives

Conclusions

- Novel approach useful for large scale distributed wireless networks
- Take advantage of both Network Calculus and Model Checking
- Scales up to hundreds of nodes

Future works

- Increase the tightness of the bound
- How to represent the network dynamic in Network Calculus ?