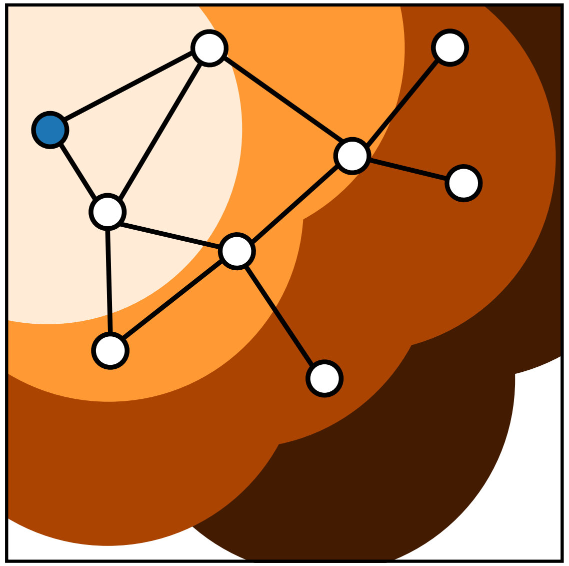
The Time-Triggered Wireless Architecture

Author(s):
Jacob, Romain; Zhang, Licong; Zimmerling, Marco; Beutel, Jan; Chakraborty, Samarjit; Thiele, Lothar

Publication date:
2020-07

Permanent link:
https://doi.org/10.3929/ethz-b-000422402

The Time-Triggered Wireless Architecture



Romain Jacob
Marco Zimmerling
Samarjit Chakraborty
Licong Zhang
Jan Beutel
Lothar Thiele

ECRTS 2020
July 7-10 – Virtual Conference

@RJacobPartner

Emerging Cyber-Physical Systems (CPS) applications have challenging requirements

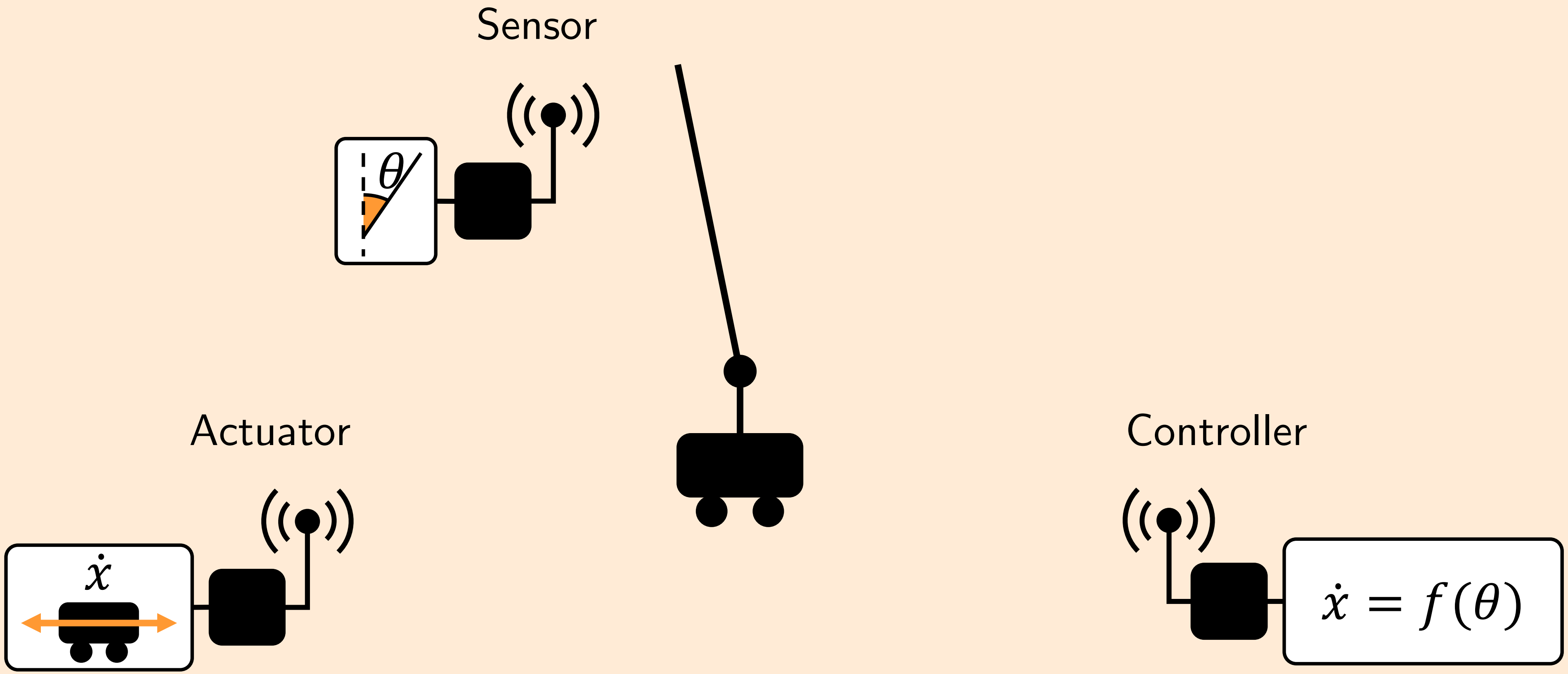
Fast and reliable

Mobile

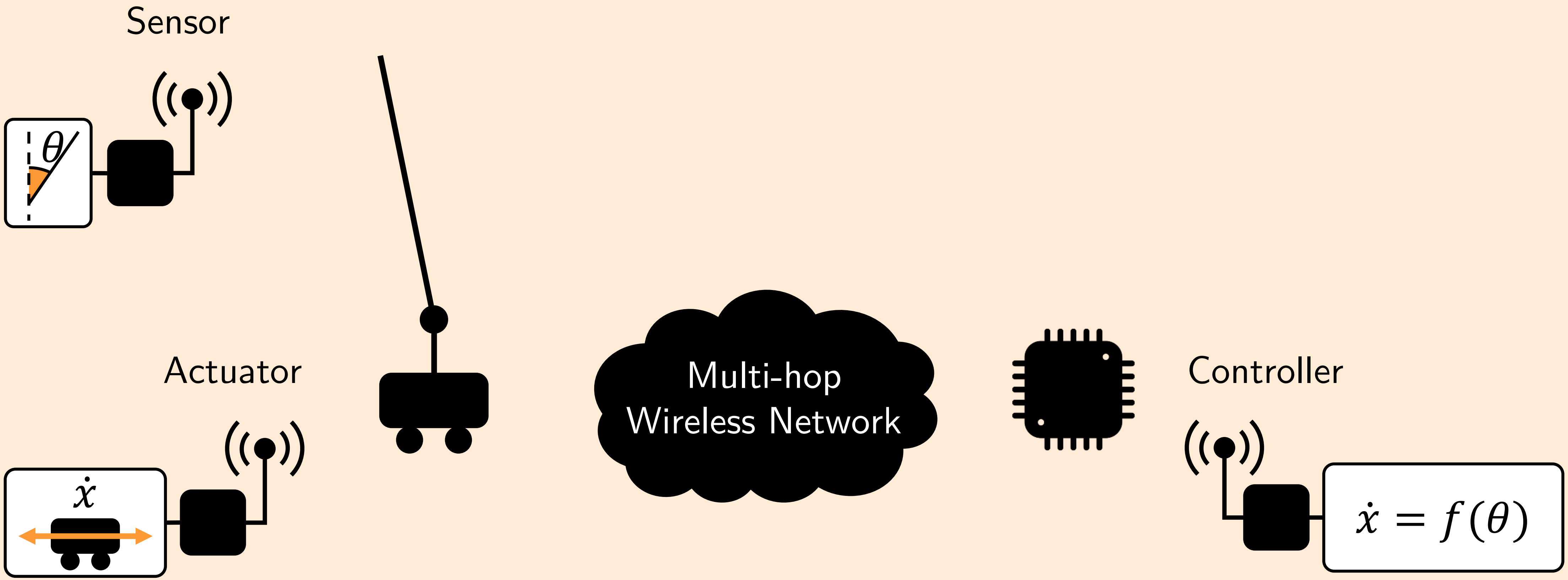
Scalable

Representative example

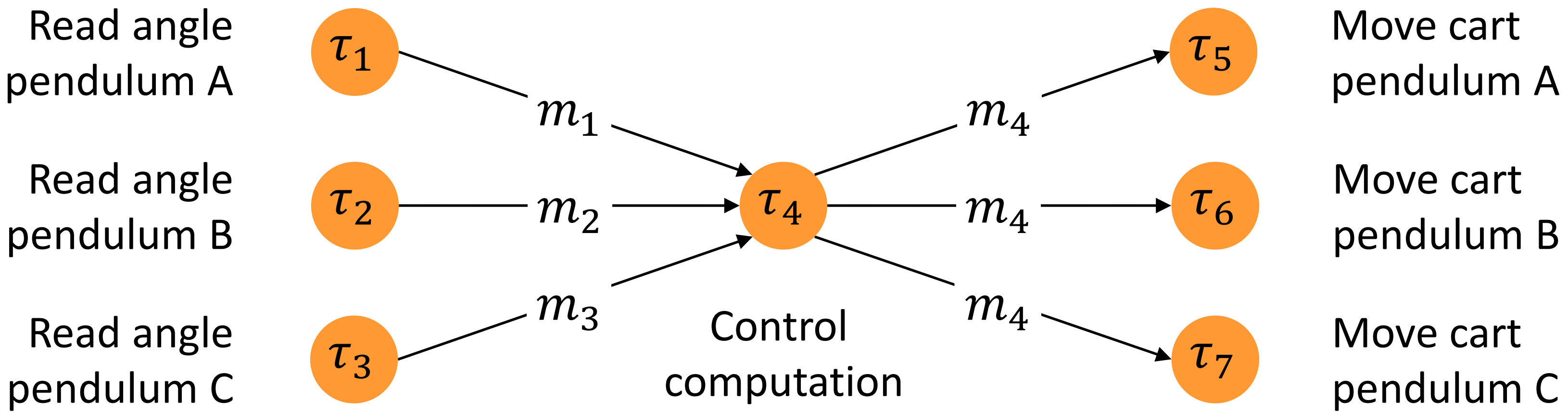
Stabilizing an inverted pendulum



Things get more complex when the controller is physically separated from the pendulum



Applications are sets of distributed tasks



The application defines the requirements

System must support

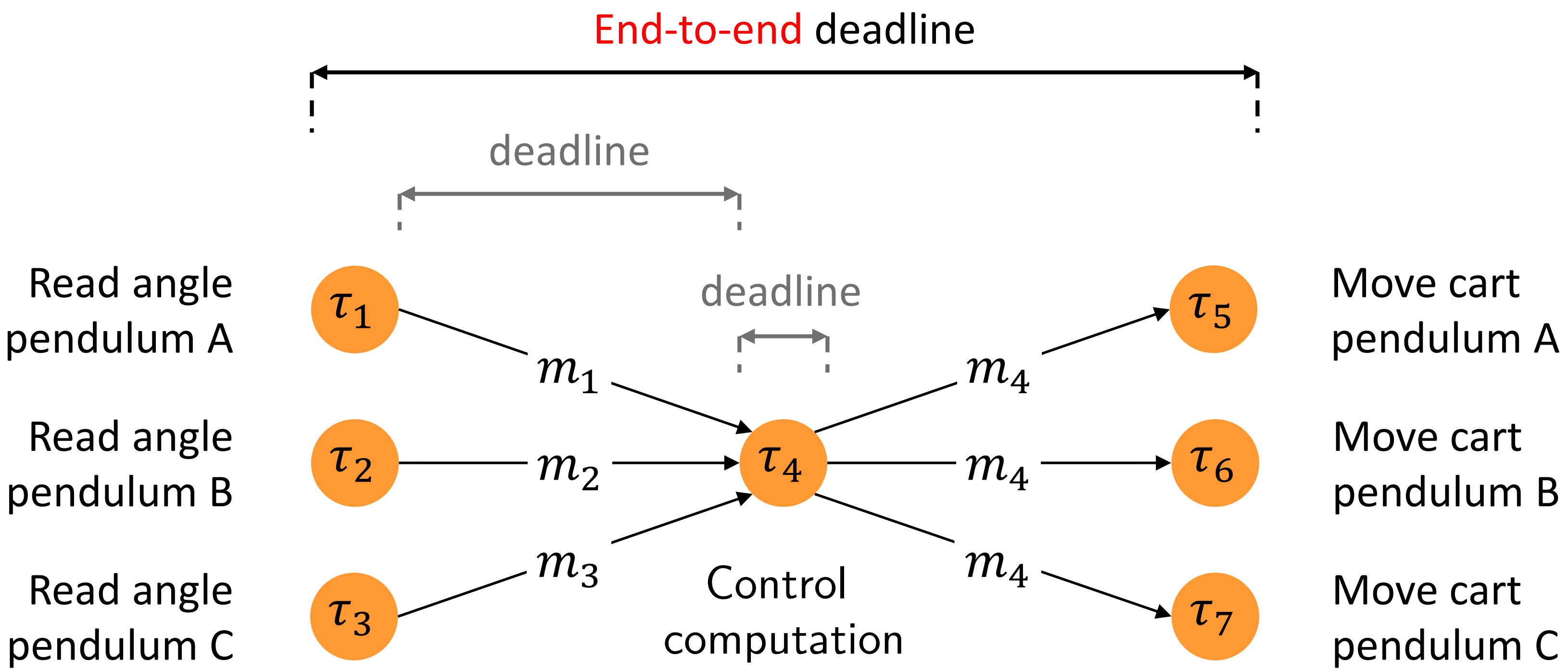
Primary goals

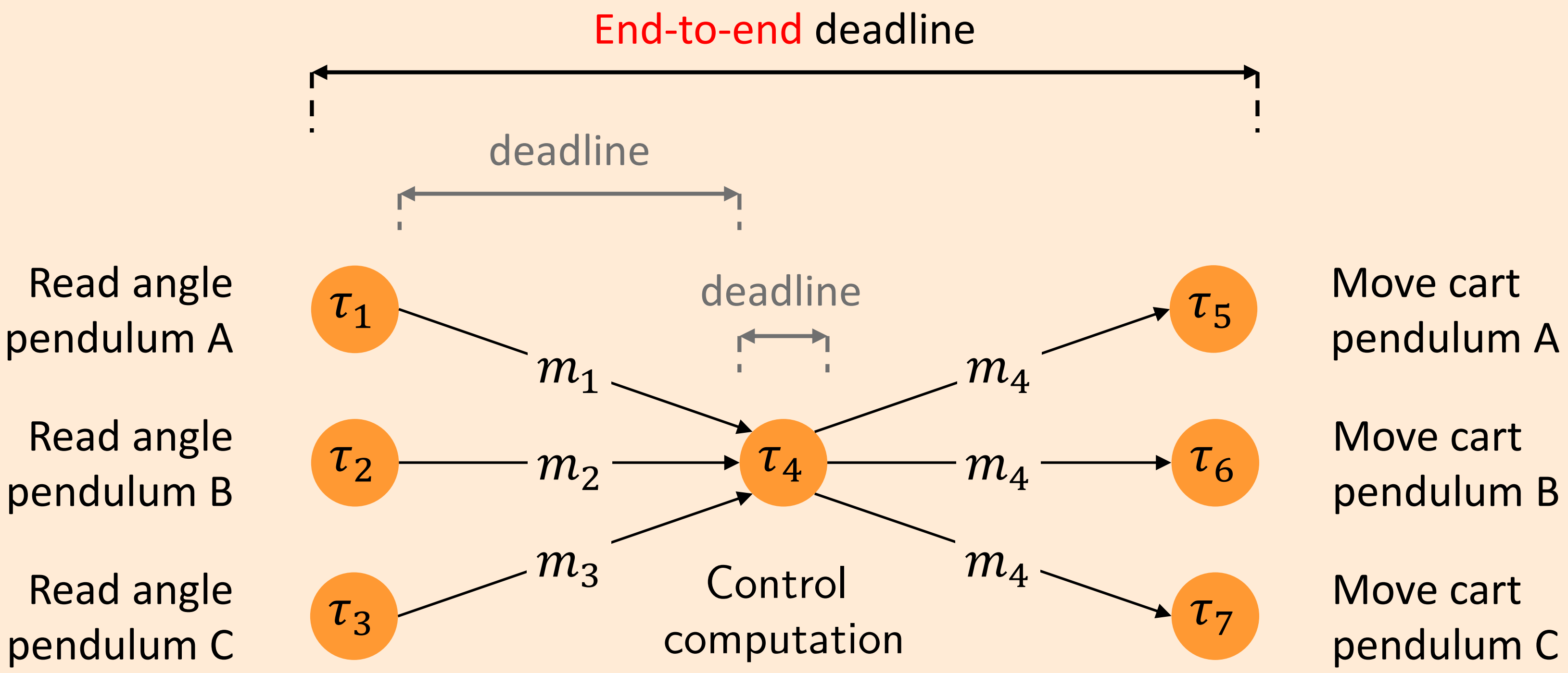
Predictability — End-to-end real-time guarantees

Adaptability
- Change in traffic demand
- Mobility

Secondary objectives

Efficiency
- Low latency
- Low energy consumption

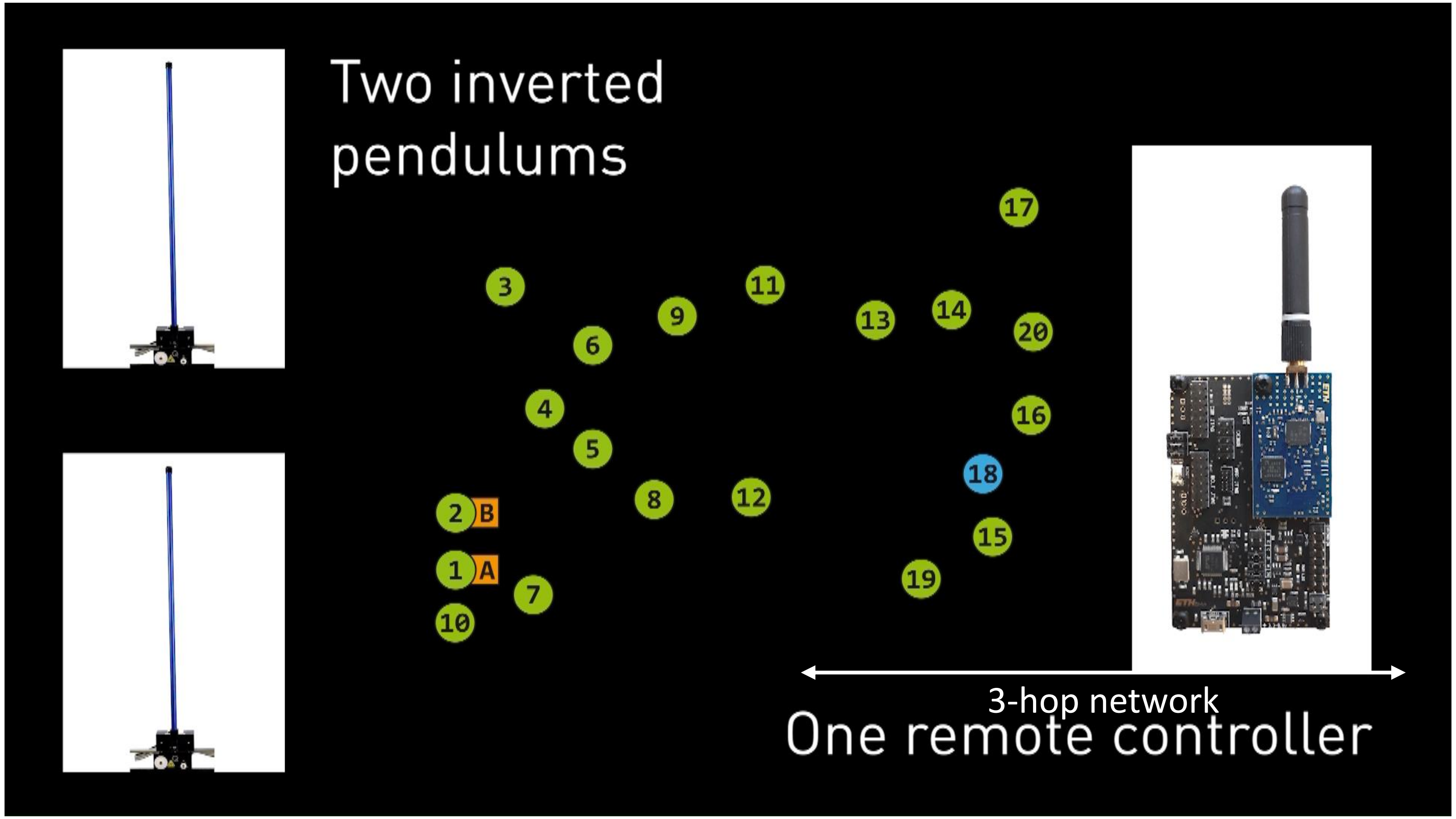

End-to-end deadline
deadline
deadline
Read angle pendulum A
Read angle pendulum B
Read angle pendulum C
τ_1
τ_2
τ_3
m_1
m_2
m_3
τ_4
Control computation
m_4
m_4
m_4
τ_5
τ_6
τ_7
Move cart pendulum A
Move cart pendulum B
Move cart pendulum C



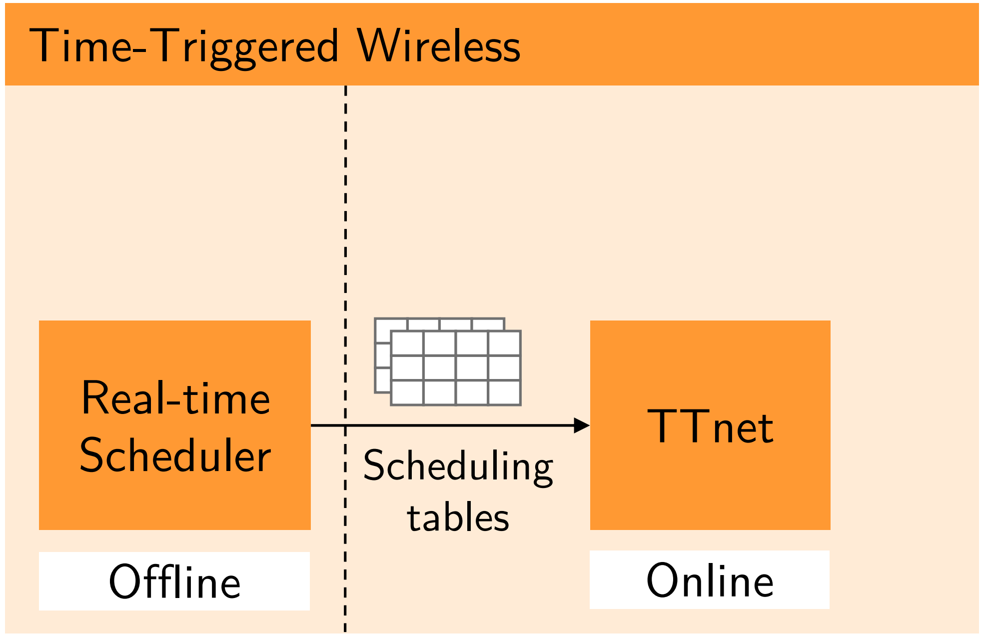
Goal of this work

Design complete architecture combining **end-to-end predictability** and **adaptability** while remaining **efficient**

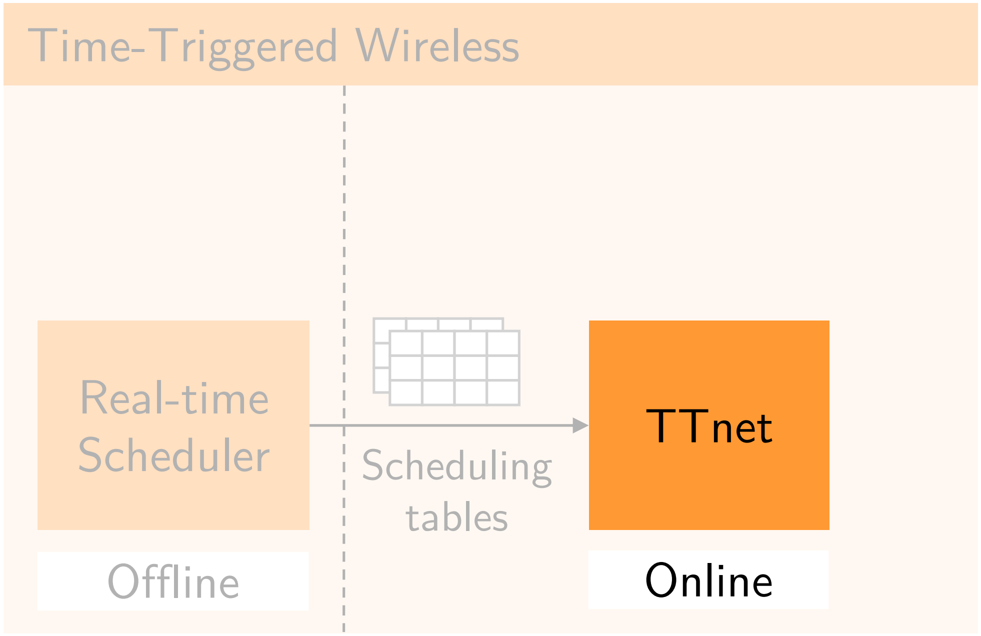
Guaranteed stability over low-power multi-hop network



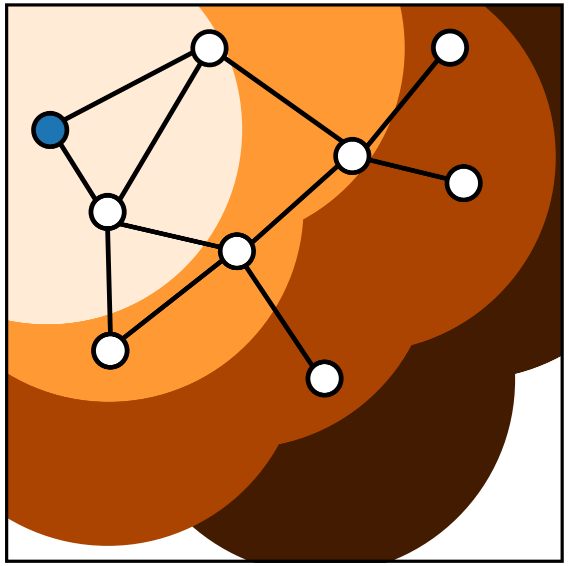
tiny.cc/WirelessCPSVideo



Time-Triggered Wireless
Real-time Scheduler
Offline
Scheduling tables
TTnet
Online


Time-Triggered Wireless
Real-time Scheduler
Offline
Scheduling tables
TTnet
Online

The Time-Triggered Wireless Architecture



Synchronous transmissions

abstract the network as a virtual bus

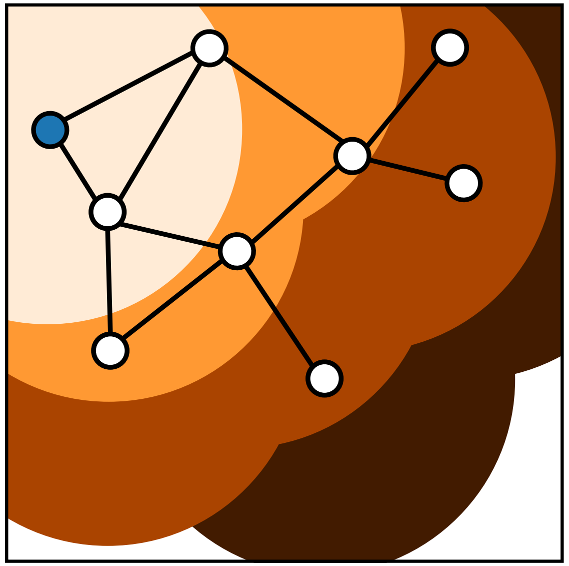
Co-scheduling approach

providing end-to-end guarantees

Predictable implementation

matching the scheduler's assumptions

The Time-Triggered Wireless Architecture



Wireless communication technique

- Multi-hop broadcast using flooding

Synchronous transmissions

abstract the network as a virtual bus

Co-scheduling approach

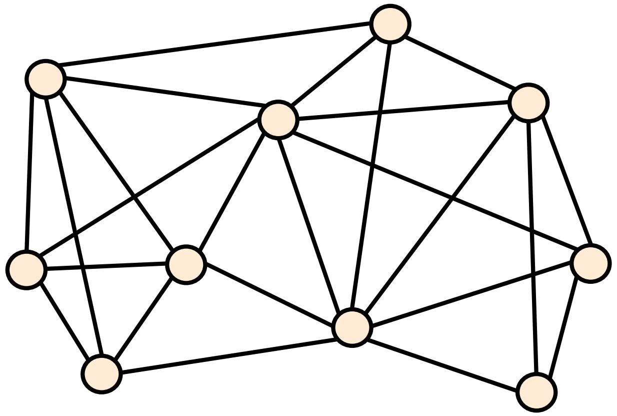
providing end-to-end guarantees

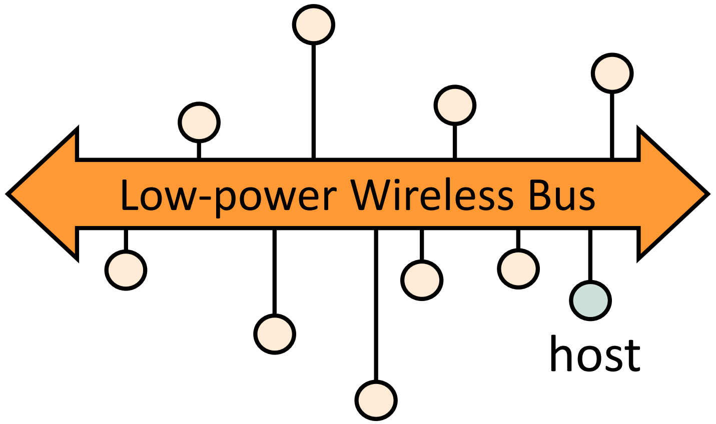
Predictable implementation

matching the scheduler's assumptions

Using synchronous transmissions,

multi-hop network can be abstracted as a shared bus

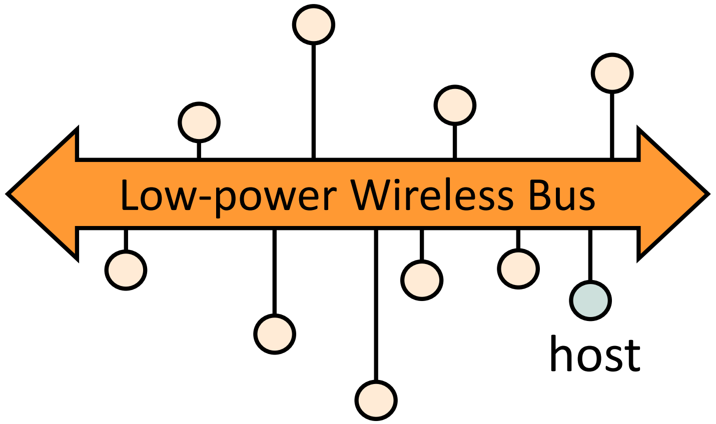


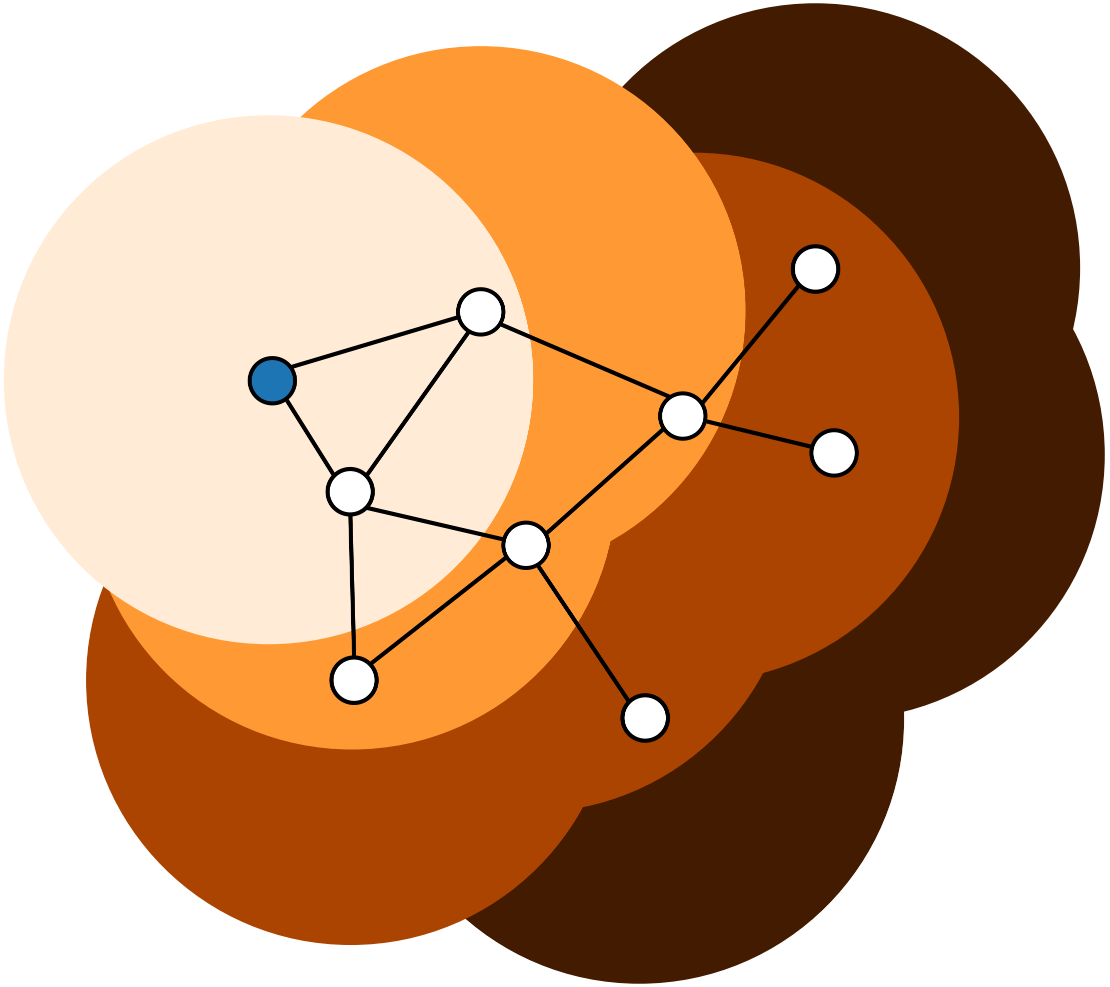


Low-Power Wireless Bus.
F. Ferrari, M. Zimmerling, L. Mottola, L. Thiele. SenSys, 2012.

Much simpler to schedule

shared bus

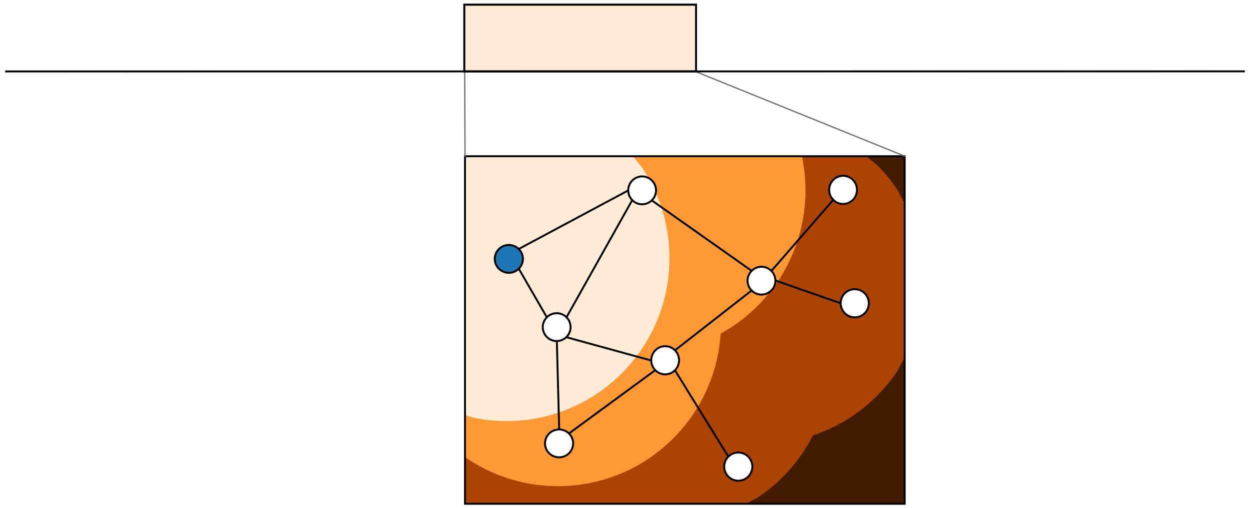




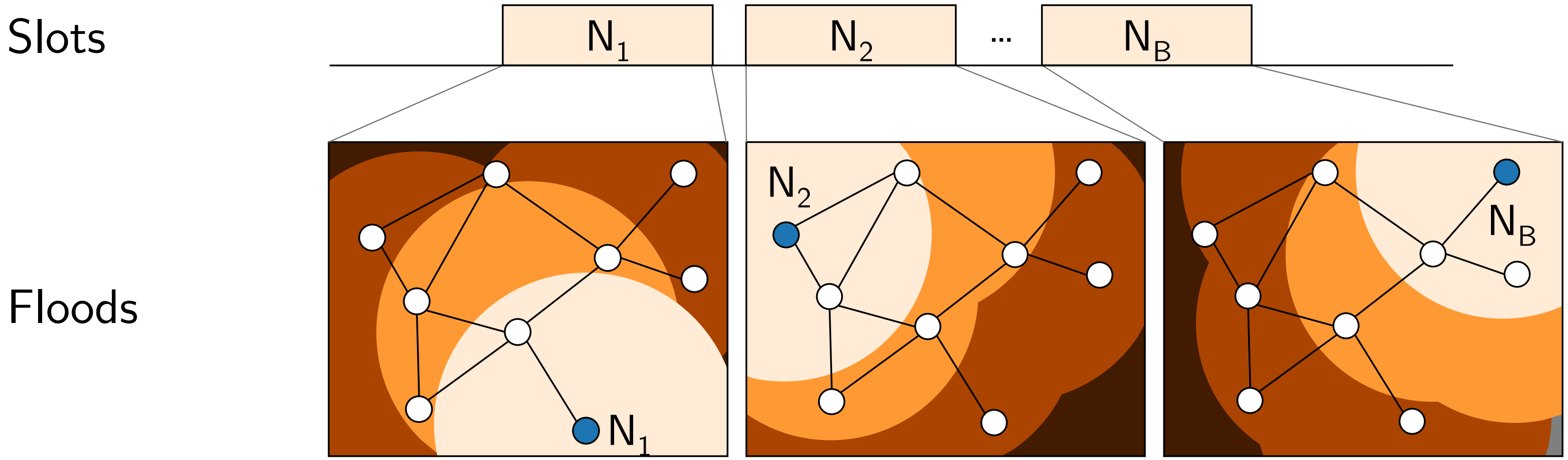
Typical network stack based on synchronous transmissions

Slots

Floods

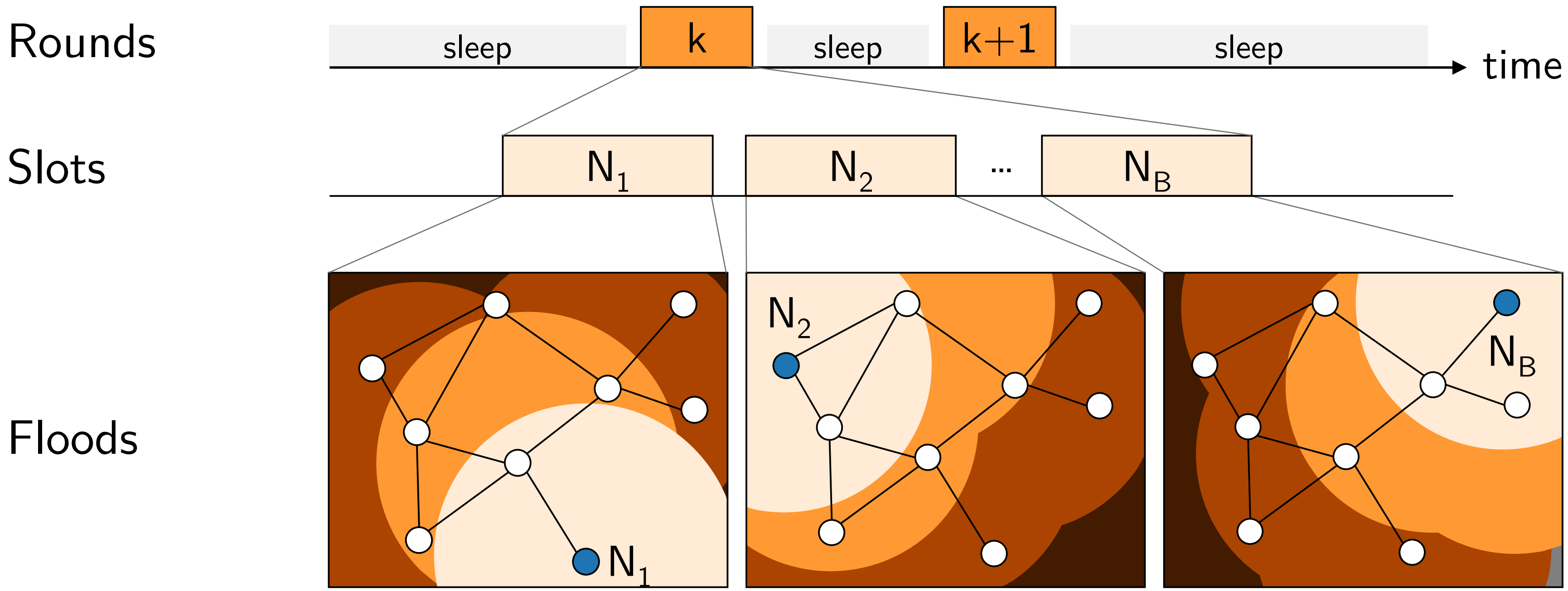


Typical network stack based on synchronous transmissions



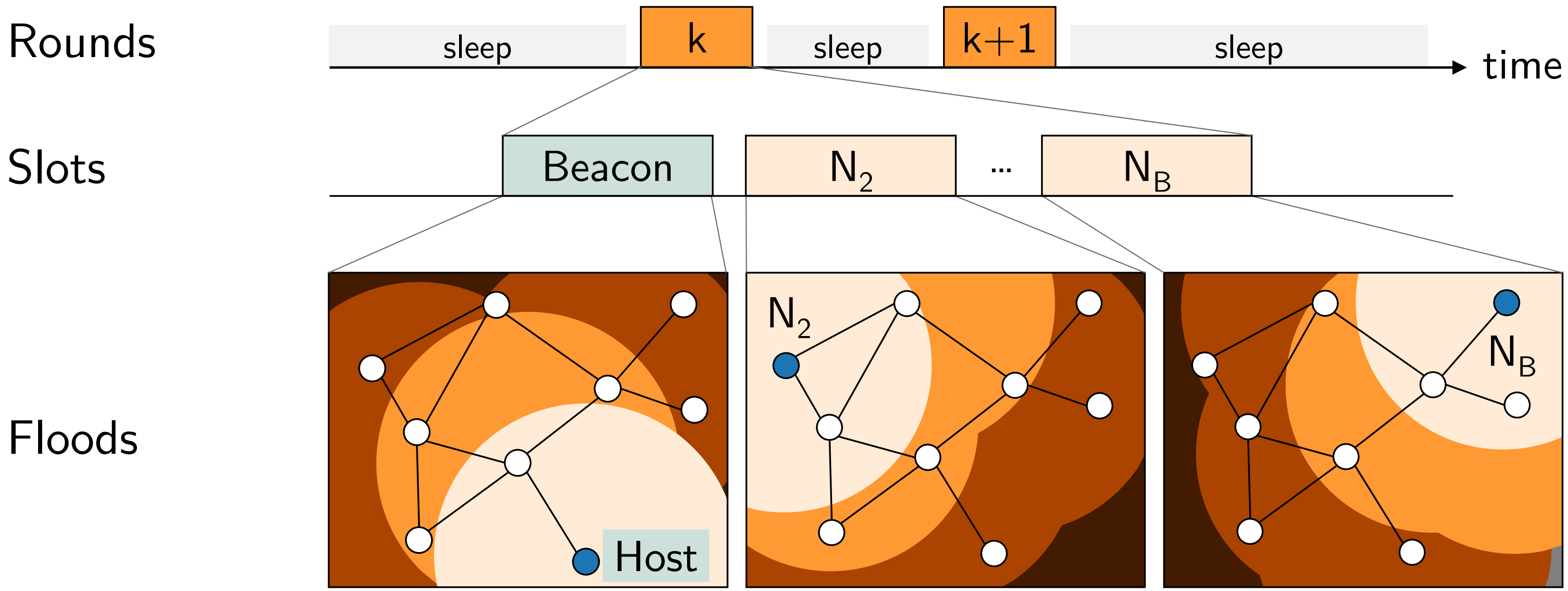
Nodes are assigned dedicated slots to send their messages

Typical network stack based on synchronous transmissions

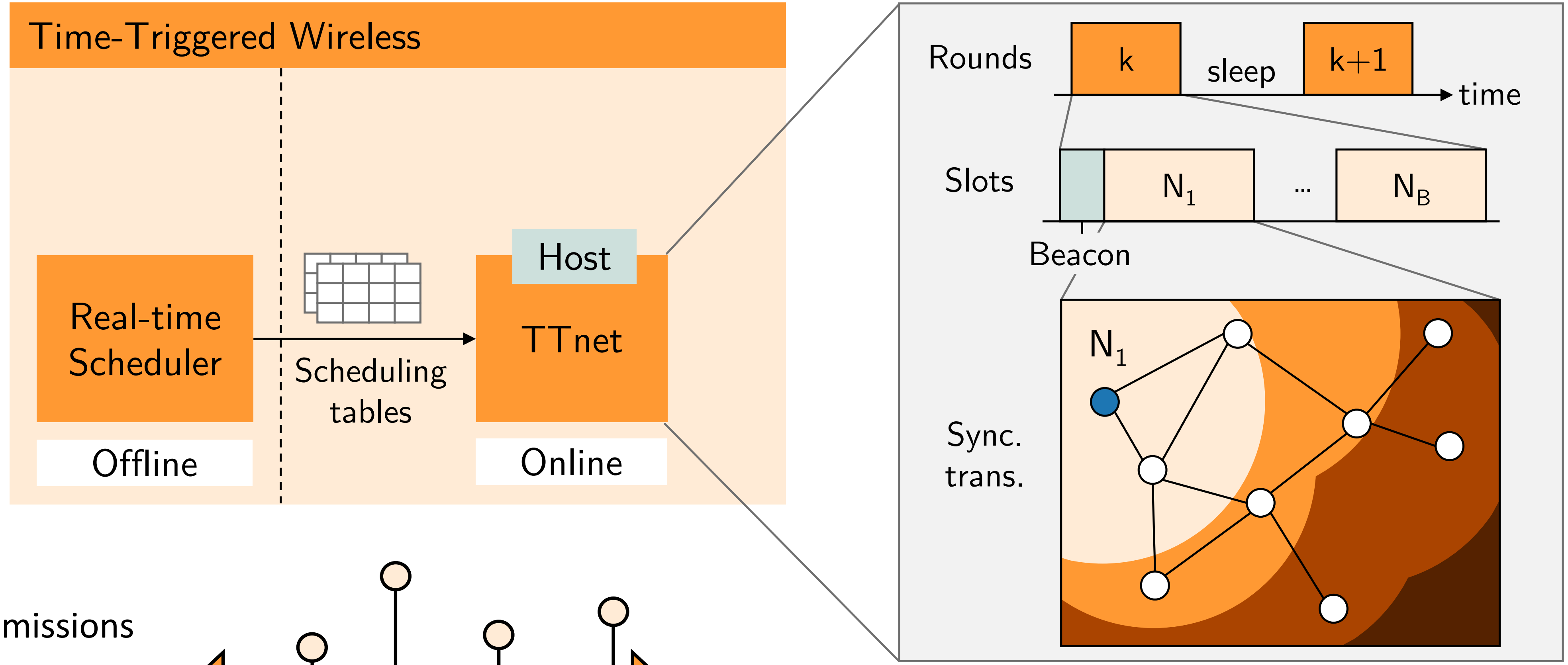


We save energy using rounds

Typical network stack based on synchronous transmissions

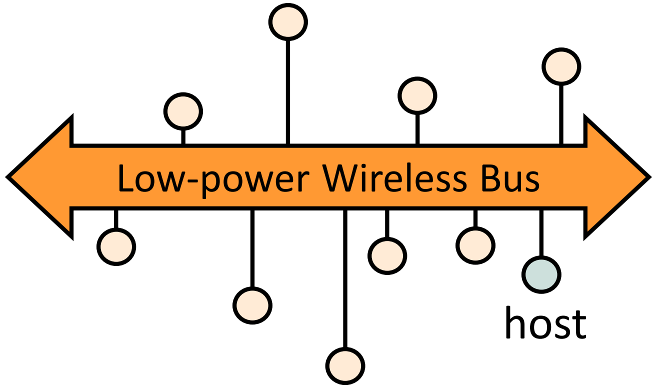


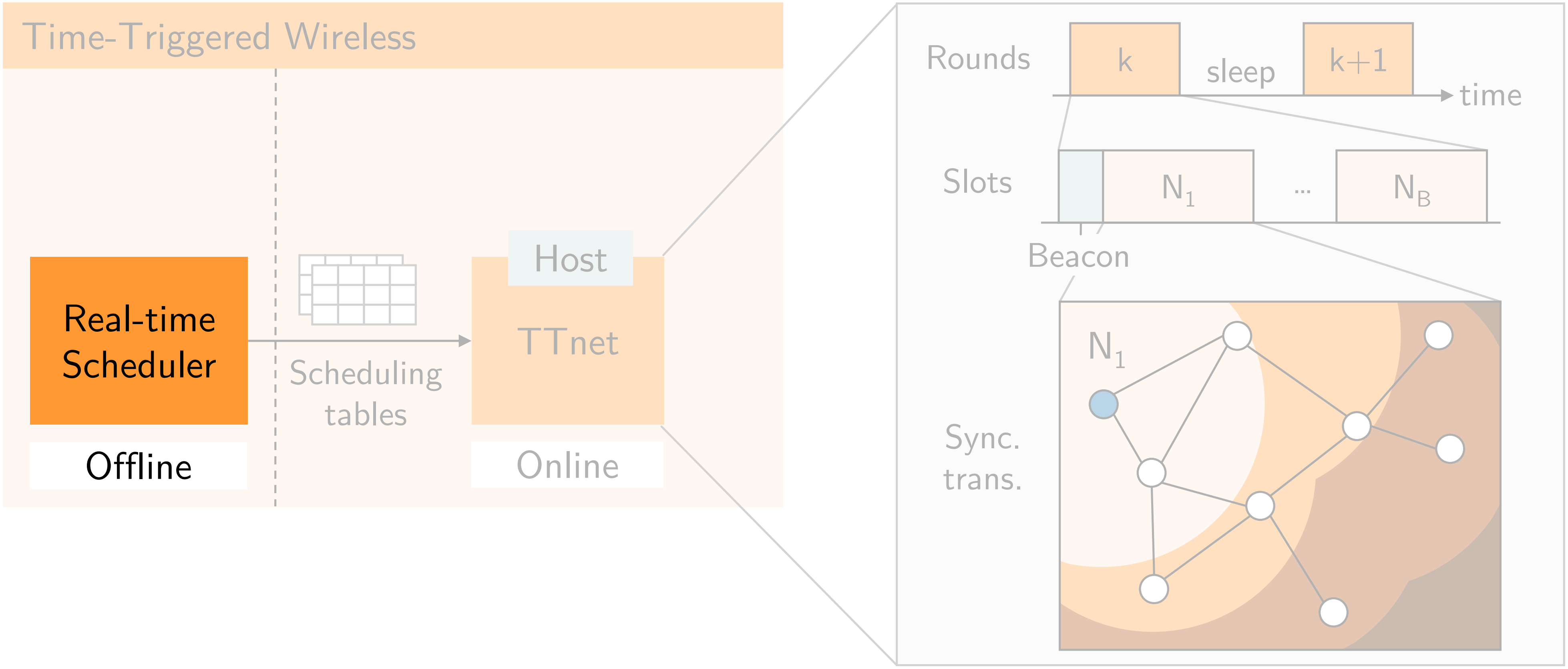
A central host controls the operations at runtime



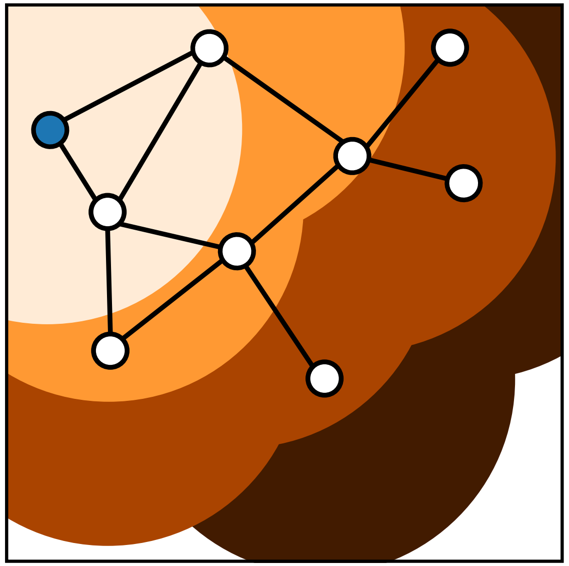
Synchronous transmissions

abstract the network as a virtual bus




Time-Triggered Wireless
Real-time Scheduler
Offline
Scheduling tables
Host
TTnet
Online
Rounds
k
sleep
k+1
time
Slots
N1
...
NB
Beacon
Sync. trans.
N1

The Time-Triggered Wireless Architecture



Synchronous transmissions

abstract the network as a virtual bus

Co-scheduling approach

providing end-to-end guarantees

Predictable implementation

matching the scheduler’s assumptions

System model

- Applications are strictly periodic

Periodic execution time of tasks

- No jitter

- Applications have arbitrary deadlines

$a.d$ is unrelated to $a.p$

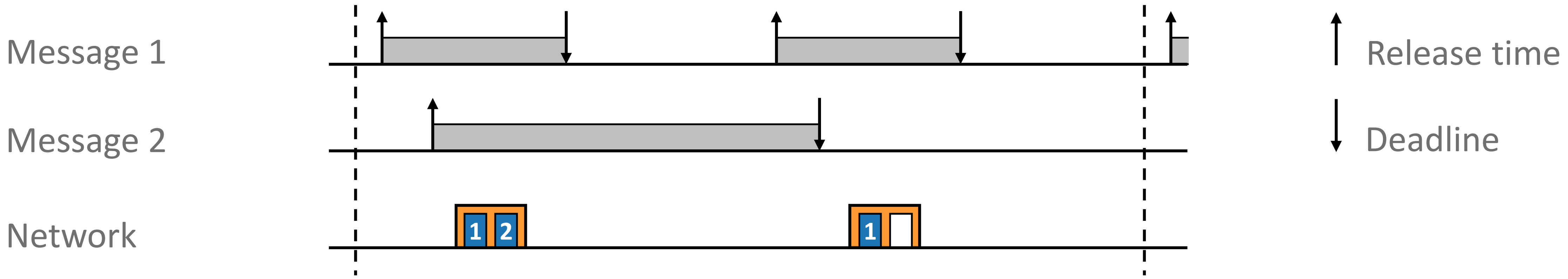
- Tasks and messages inherit period from their application

$\tau.p = m.p = a.p$

- Persistent applications remain undisturbed by changes in the rest of the system

Task execution remains periodic

How to schedule the messages and communication rounds?

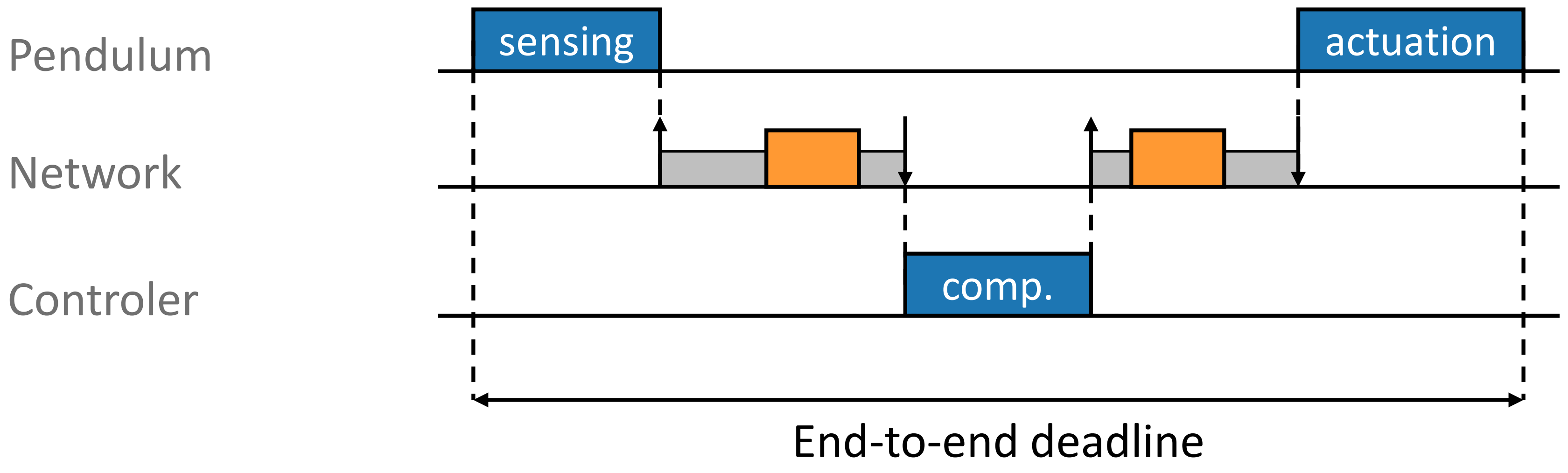


In practice

How do you get
the message deadlines?

From the execution
time of the tasks!

End-to-end guarantees require coupling between the task and message schedules



End-to-end guarantees require coupling between the task and message schedules

Chosen coupling	TTW statically co-schedules all tasks and messages

Chosen coupling

TTW statically co-schedules all tasks and messages

- Schedule offline based on a MILP
- Execute at runtime

Inspired by time-triggered wired networks
e.g., TTEthernet, FlexRay

Offline scheduling allows to minimize the achievable latency at runtime

Rounds save energy but complexify the synthesis

Messages must be served in a round that finishes before their deadline

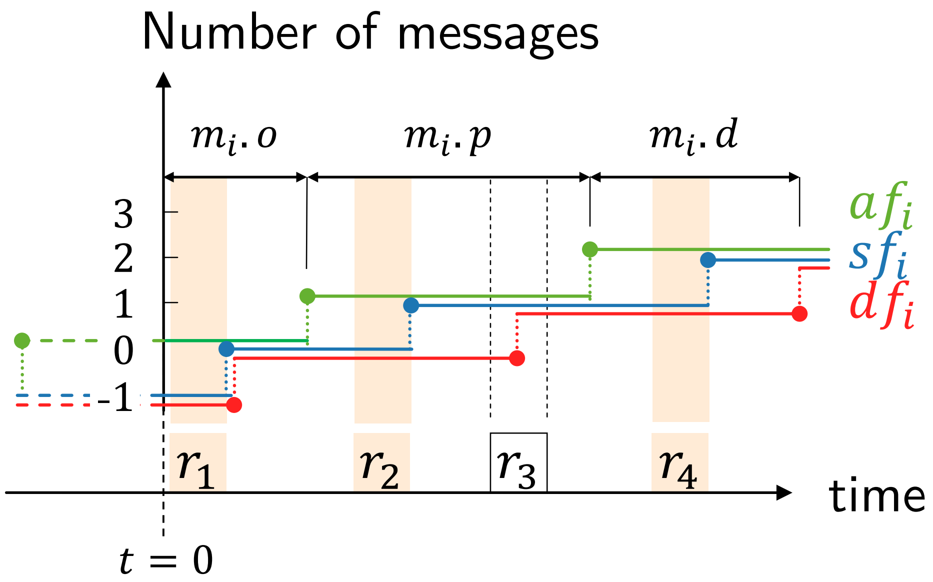
16 B	Payload	$\Rightarrow$	Reduction of ~ 30% in radio-on time
10	Slots per round		

$$\delta_{ij} = \begin{cases} 1 & \text{if } m_i \text{ is assigned to } r_j \\ 0 & \text{otherwise} \end{cases}$$

$$m_i.d \geq (r_j.o + T_r) * \delta_{ij}$$

Non-linear!

Our solution is inspired by network calculus



Count the number of message instances of m_i that

af_i have been released

sf_i have been served

df_i have passed their deadline

Our solution is inspired by network calculus

$$m_i.d \geq (r_j.o + T_r) * \delta_{ij}$$

$$\Leftrightarrow \quad sf(r_j.o) \geq df(r_j.o + T_r)$$

$$\Leftrightarrow \quad \sum_{k=1}^{j} \delta_{ij} \geq \left\lceil \frac{r_j.o + T_r - m_i.o - m_i.d}{m_i.p} \right\rceil$$

Linear

Piecewise-constent
can be handled with common MILP tricks

Count the number of message instances of m_i that

af_i have been released

sf_i have been served

df_i have passed their deadline

Static scheduling is nice but static

		System must support
Primary goals	Predictability	End-to-end real-time guarantees
	Adaptability	- Change in traffic demand - Mobility
Secondary objectives	Efficiency	- Low latency - Low energy consumption

System switches between operation modes at runtime

Well-known approach

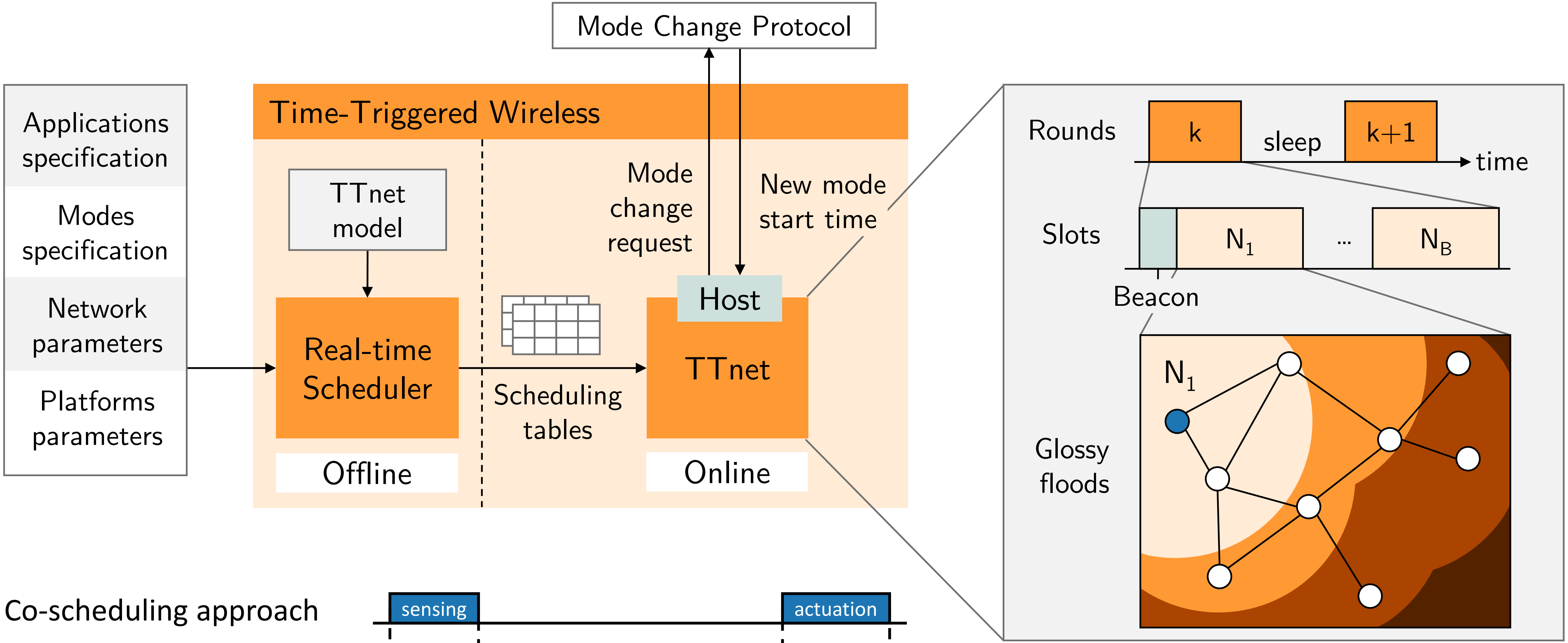
- Synthesize schedules for mulitple operation modes
- Switch between modes at runtime

Challenge

Preserve real-time guarantees across mode changes

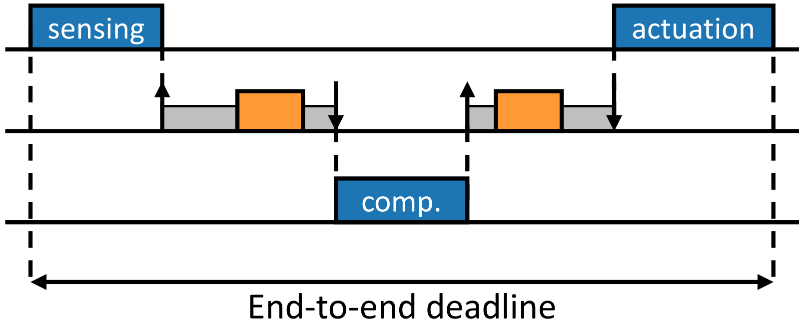
- Creates dependencies between modes
- Tackled in TTW while aiming to limit the impact on energy consumption

See the paper for more details



Co-scheduling approach

providing end-to-end guarantees



Primary goals

Predictability

Adaptability

Secondary objectives

Efficiency

System must support

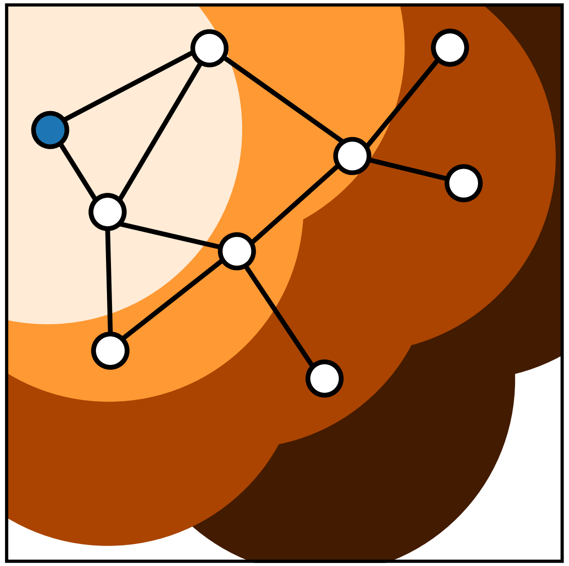
End-to-end real-time guarantees

- Change in traffic demand
- Mobility

All great! … on paper

- Can we implement this?
- Does it really work?

The Time-Triggered Wireless Architecture



Synchronous transmissions

abstract the network as a virtual bus

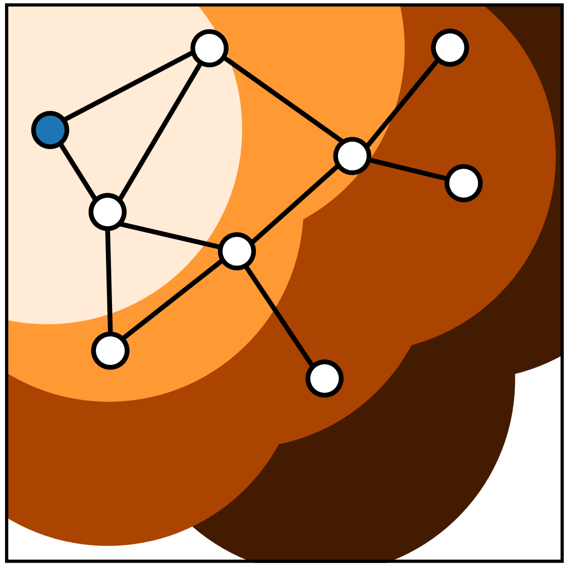
Co-scheduling approach

providing end-to-end guarantees

Predictable implementation

matching the scheduler's assumptions

The Time-Triggered Wireless Architecture



Synchronous transmissions

abstract the network as a virtual bus

Co-scheduling approach

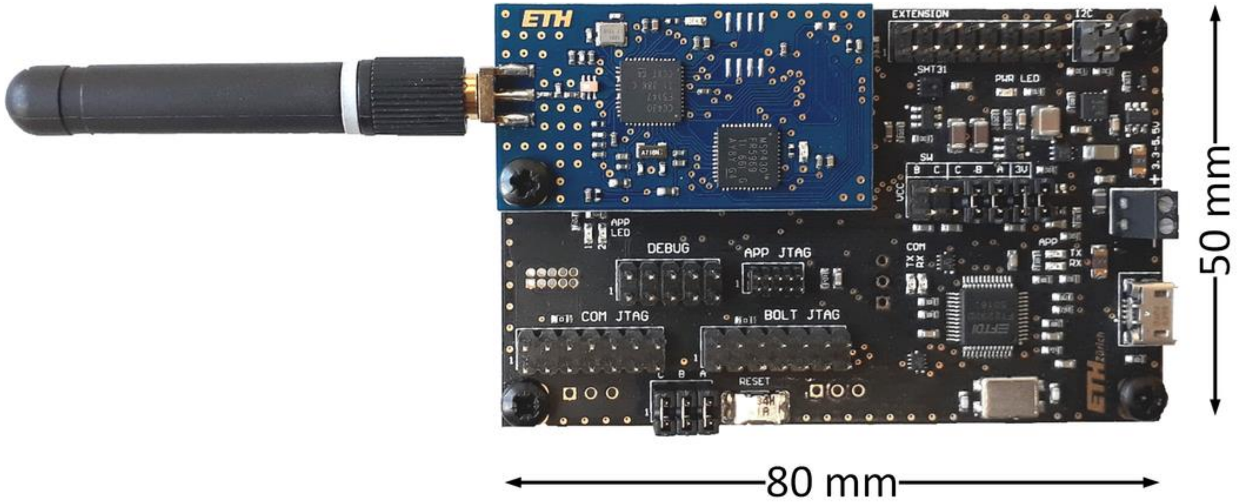
providing end-to-end guarantees

Predictable implementation

matching the scheduler's assumptions

- WCET of tasks
- WCET of messages

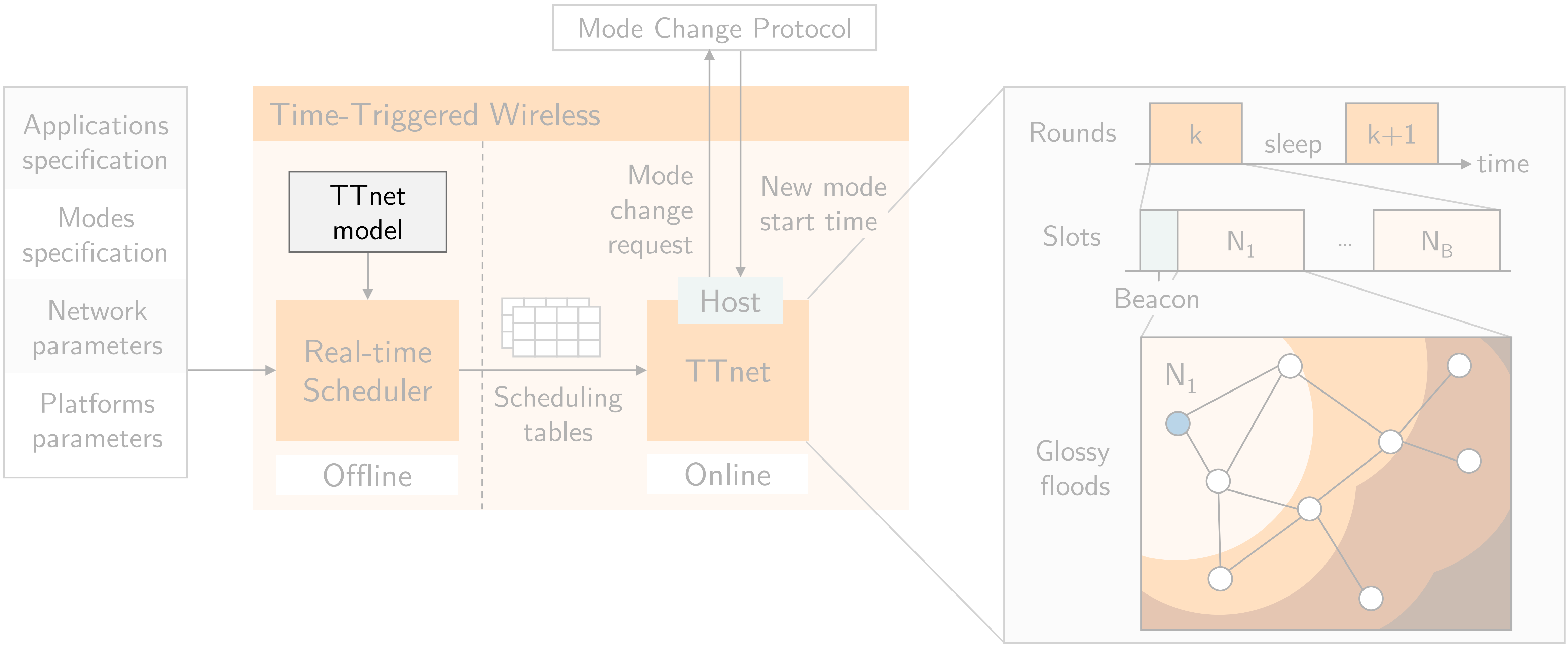
TTnet runs on embedded hardware



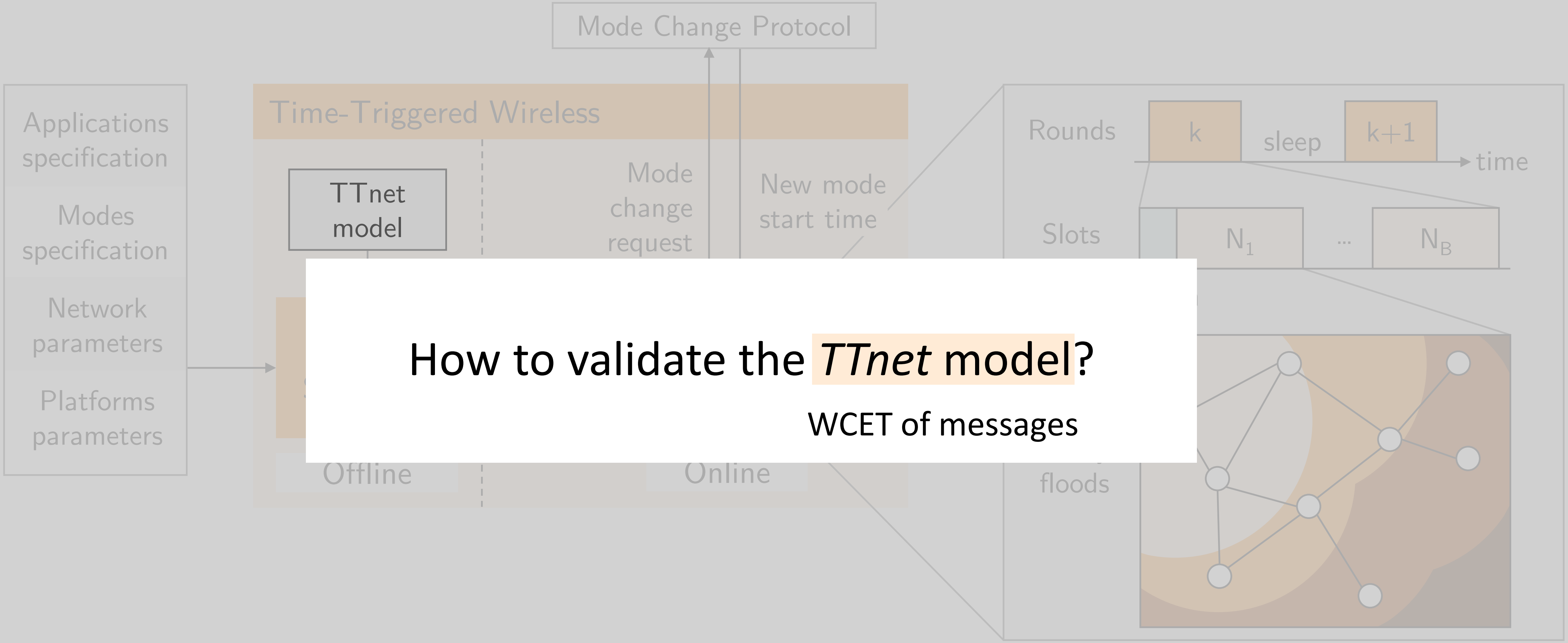
The Dual Processor Platform Architecture: Demo Abstract
J. Beutel *et al.*. IPSN, 2019

Provide a precise model of the execution time of communication rounds

Synchronous Transmissions Made Easy: Design Your Network Stack with Baloo
R. Jacob, J. Bächli, R. da Forno, L. Thiele. EWSN, 2019



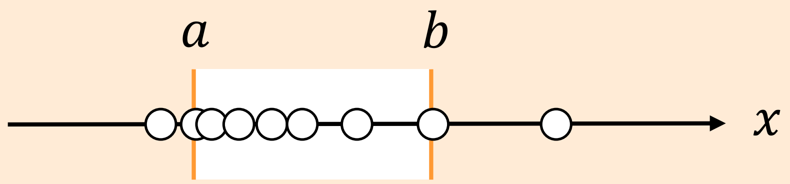
Mode Change Protocol
Applications specification
Modes specification
Network parameters
Platforms parameters
Time-Triggered Wireless
TTnet model
Real-time Scheduler
Offline
Mode change request
New mode start time
Host
TTnet
Online
Scheduling tables
Rounds
k
sleep
k+1
time
Slots
N1
...
NB
Beacon
Glossy floods
N1



How to validate the *TTnet* model?

WCET of messages

We used confidence intervals (CI)



Informally

Numerical interval
in which lies
the true value
(which you do not know)
of some parameter
with a certain probability,
called the confidence level

$[a, b]$ is a 95% CI for the median of x

which means that

The probability that the true median of x is within $[a, b]$ is larger or equal to 95%.

It is actually easy to compute CI for percentiles

Probability of any P_p to be between two consecutive samples

$$Pr\{x_k \le P_p \le x_{k+1}\} = \binom{N}{k} p^k (1-p)^{N-k}$$

Binomial distribution

For any confidence c

For any percentile P_p

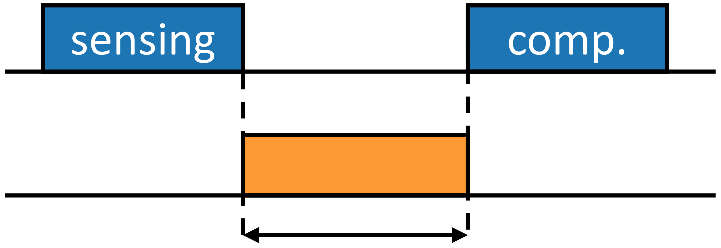
$$N \ge \frac{\log(1-c)}{\log(1-p)}$$

95% CI $c = 0{,}95$	Minimal number of repetitions
Median $p = 0{,}5$	6
90-th $p = 0{,}1$	29
99-th $p = 0{,}01$	**299**
99,9-th $p = 0{,}001$	**2995**

Is the *TTnet* model providing a safe and tight upper-bound?

Metric	Maximum measured round time T_r among all nodes
CI	95^{th} percentile 95% confidence level

Aim to upper-bound the 95^{th} percentile of the maximum round time T_r with 95% confidence



$T_r = f($ Number slots, Payload size $)$

Round model f

- Must be safe — No overruns
- Should be tight — Little wasted time

Metric

Maximum measured round time T_r among all nodes

CI

95th percentile

95% confidence level

Aim to upper-bound the 95th percentile of the maximum round time T_r with 95% confidence

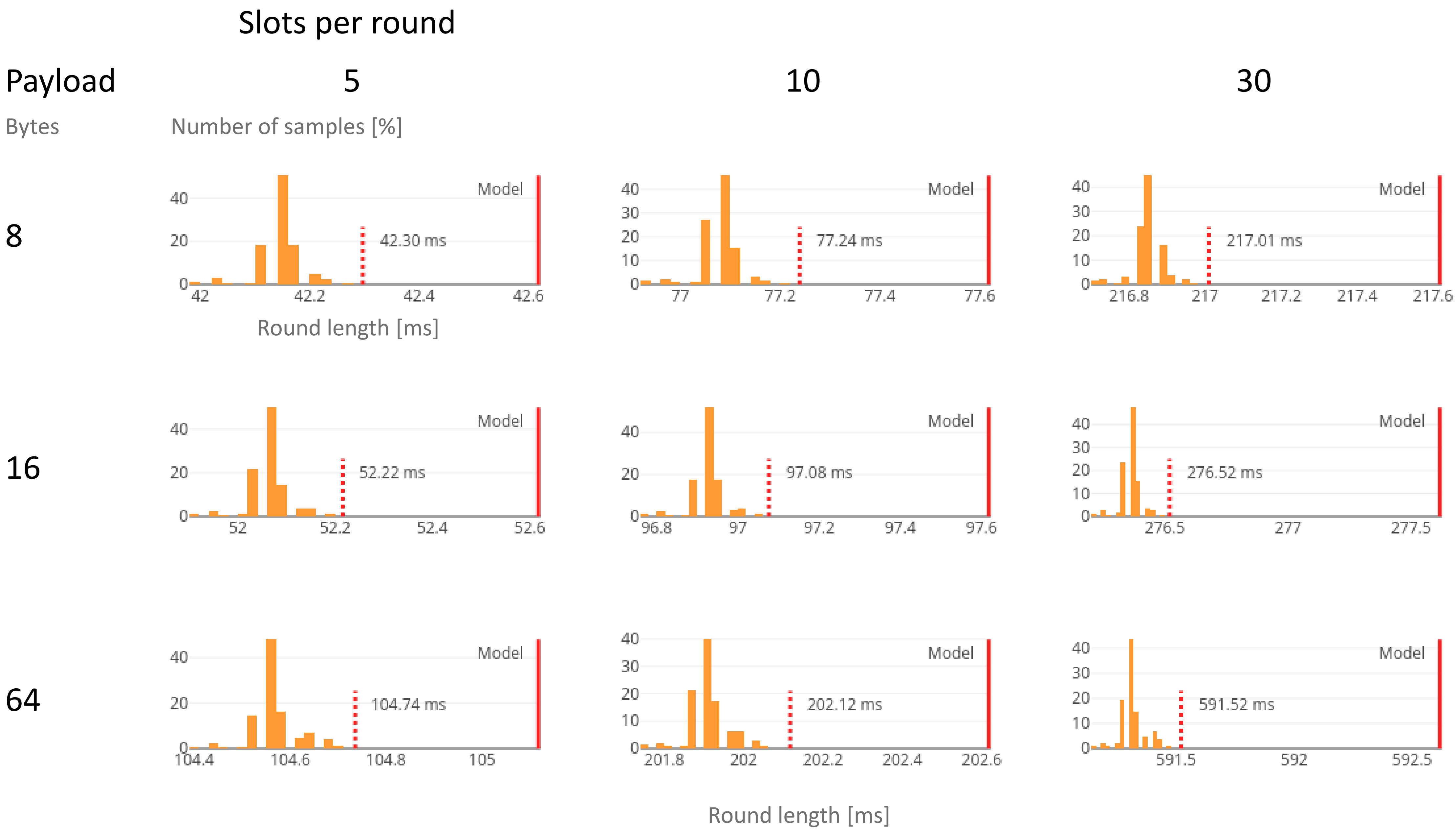
$c = 0{,}95$

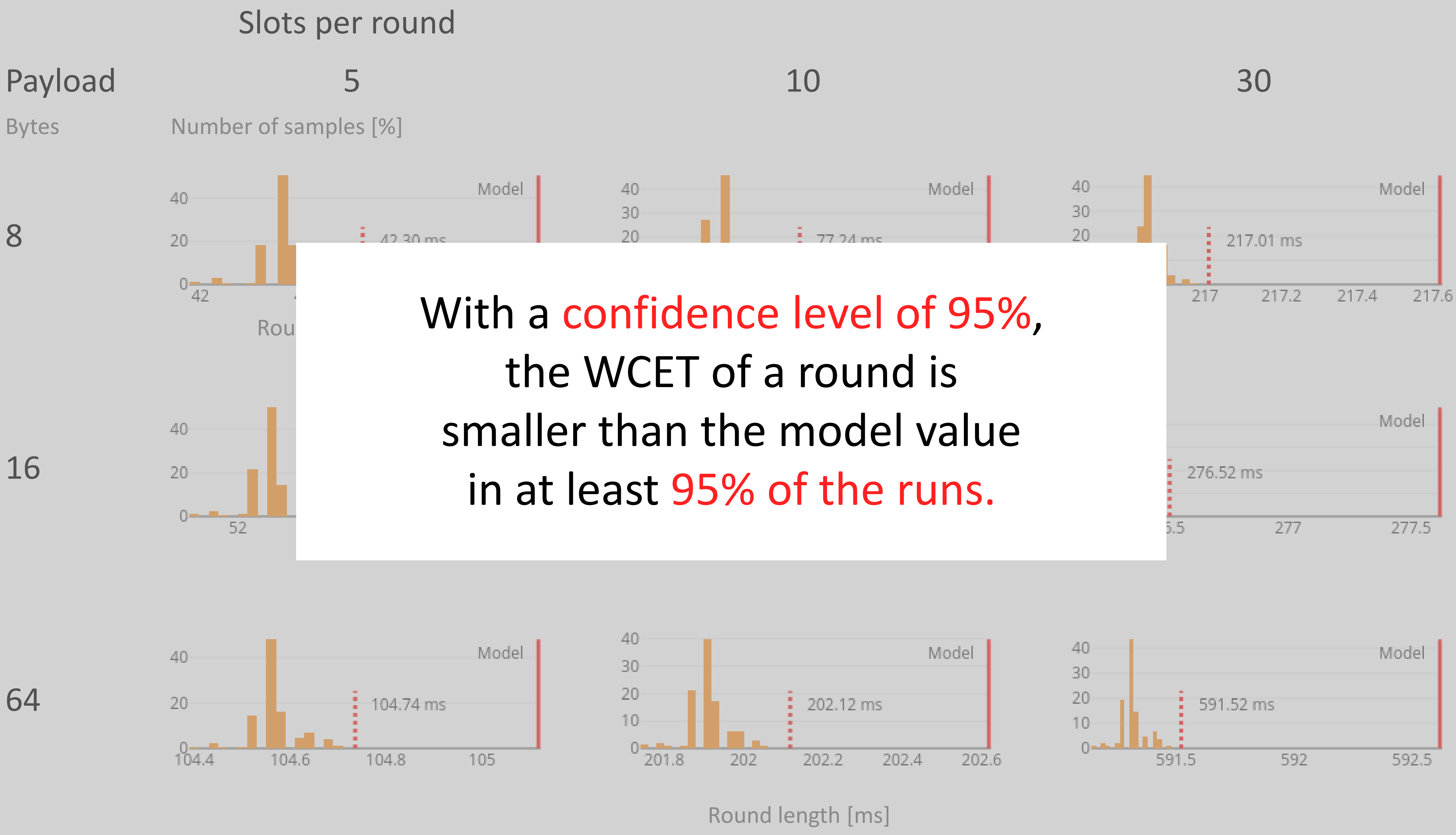
$p = 0{,}05$

$\Rightarrow$ $N \geq 59$

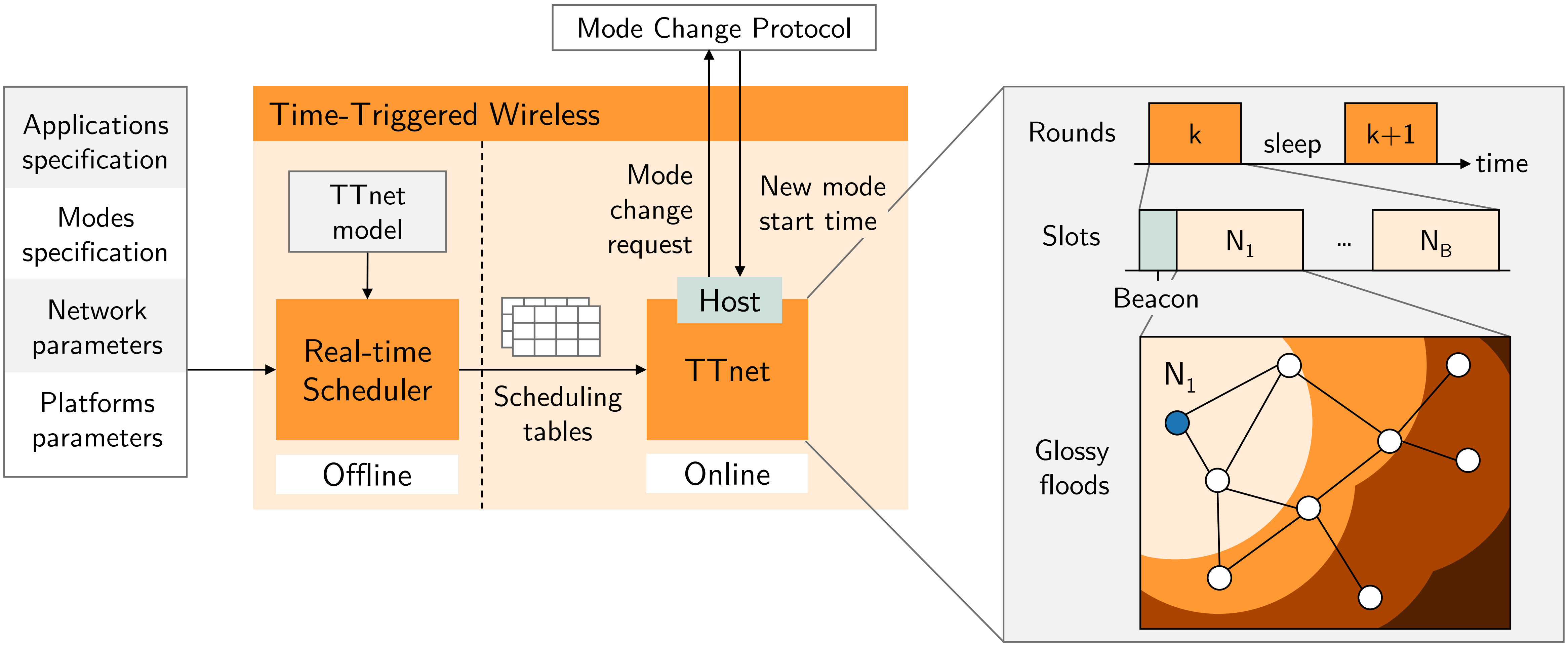
Minimal number of repetitions

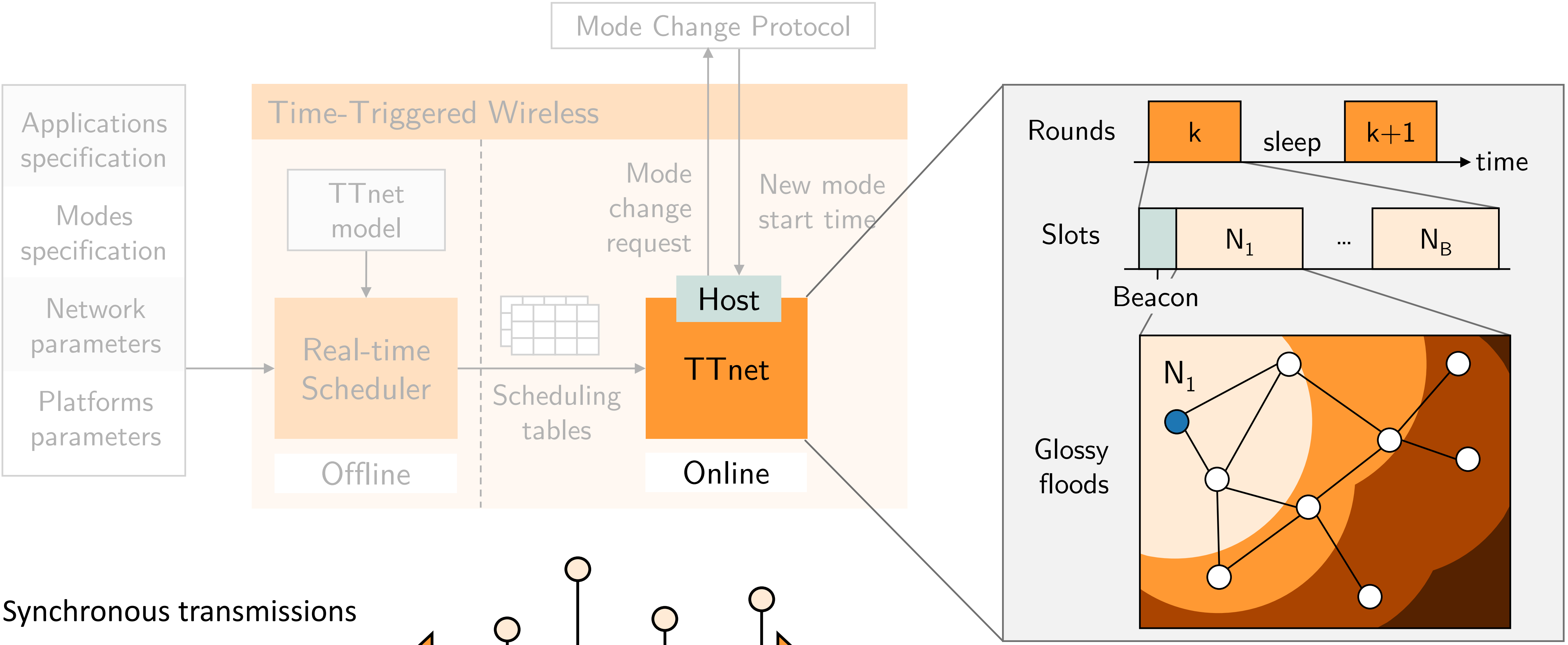
$\Rightarrow$ Scheduled 60 runs per series, randomly distributed over one week


Slots per round
Payload
Bytes
5
10
30
Number of samples [%]
8
16
64
Model
42.30 ms
77.24 ms
217.01 ms
52.22 ms
97.08 ms
276.52 ms
104.74 ms
202.12 ms
591.52 ms
Round length [ms]
Round length [ms]



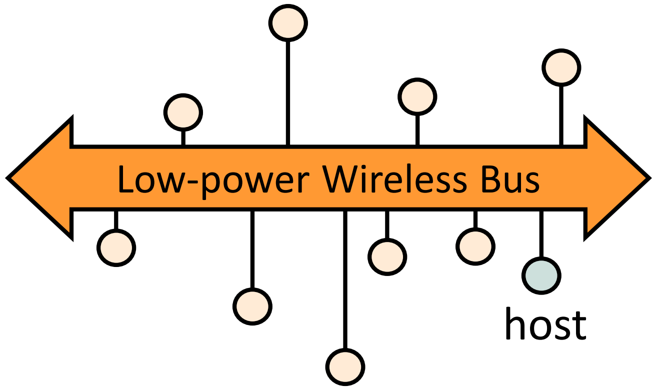
With a confidence level of 95%, the WCET of a round is smaller than the model value in at least 95% of the runs.

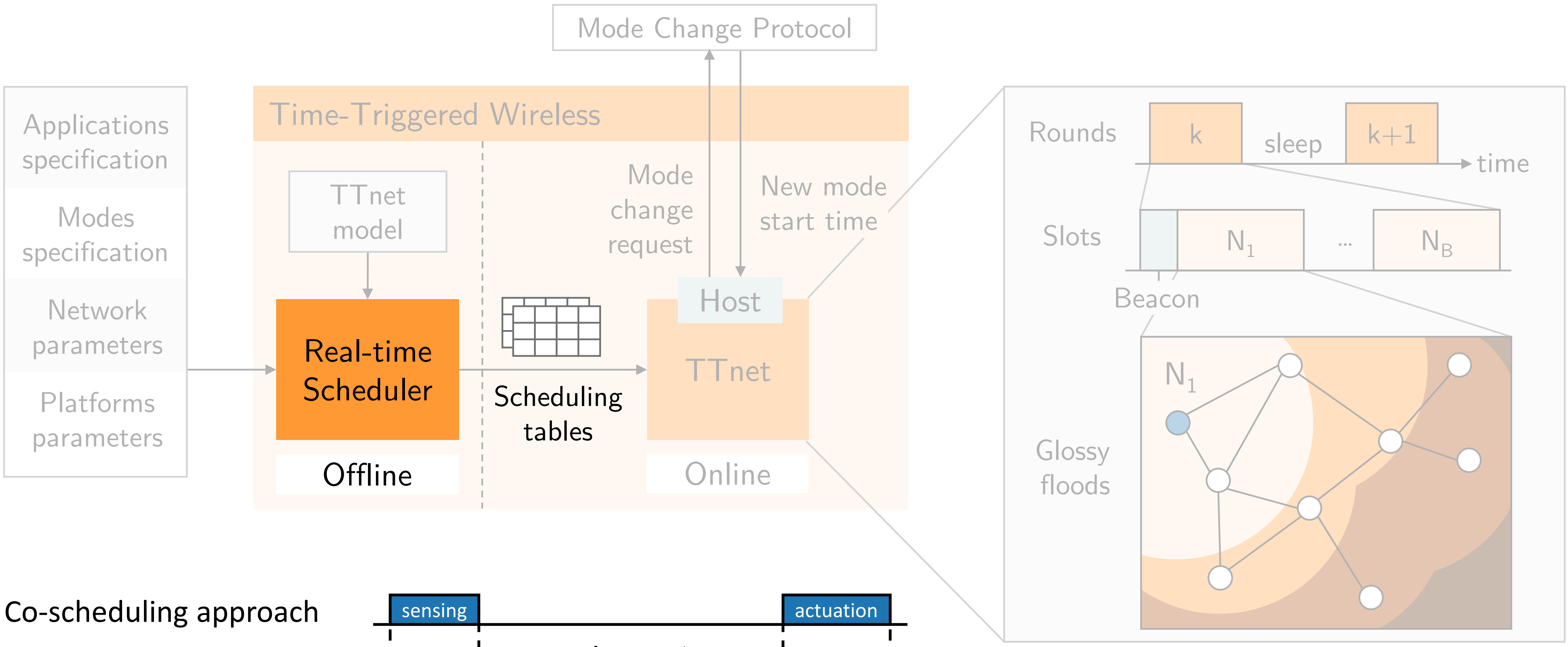

Mode Change Protocol
Applications specification
Modes specification
Network parameters
Platforms parameters
Time-Triggered Wireless
TTnet model
Real-time Scheduler
Offline
Scheduling tables
Mode change request
New mode start time
Host
TTnet
Online
Rounds
k
sleep
k+1
time
Slots
N_1
...
N_B
Beacon
Glossy floods
N_1



Synchronous transmissions

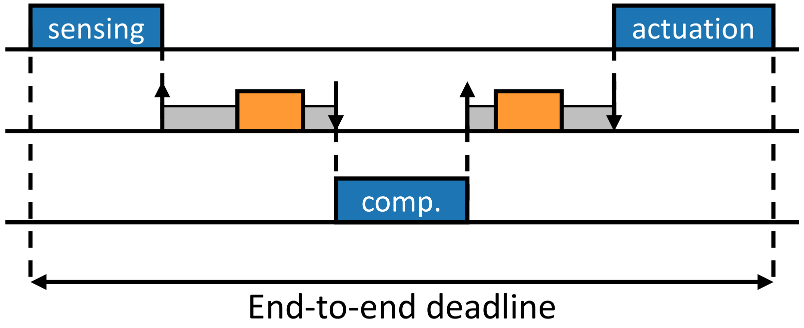
abstract the network
as a virtual bus

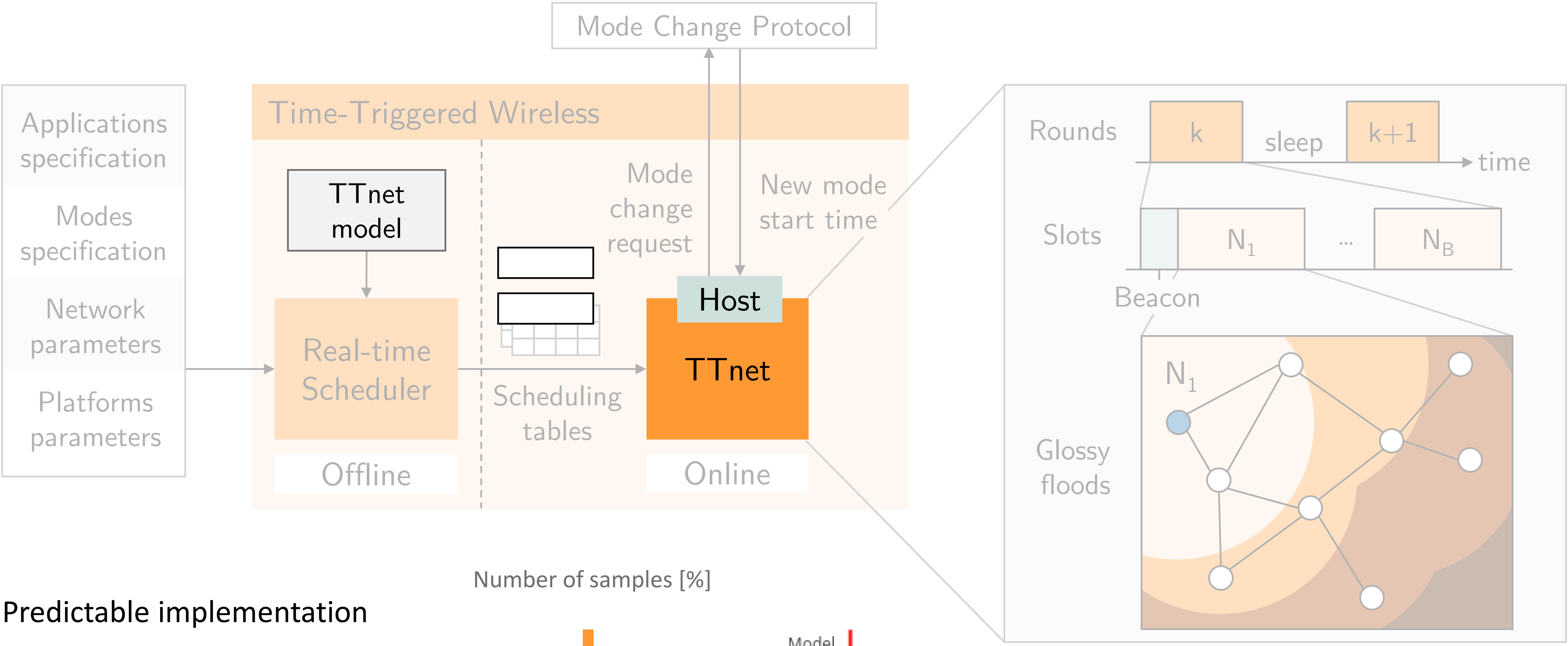




Co-scheduling approach

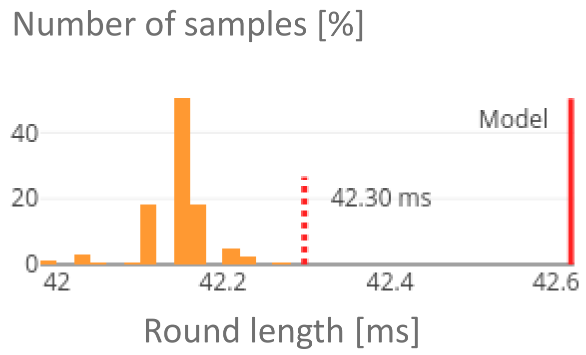
providing end-to-end guarantees



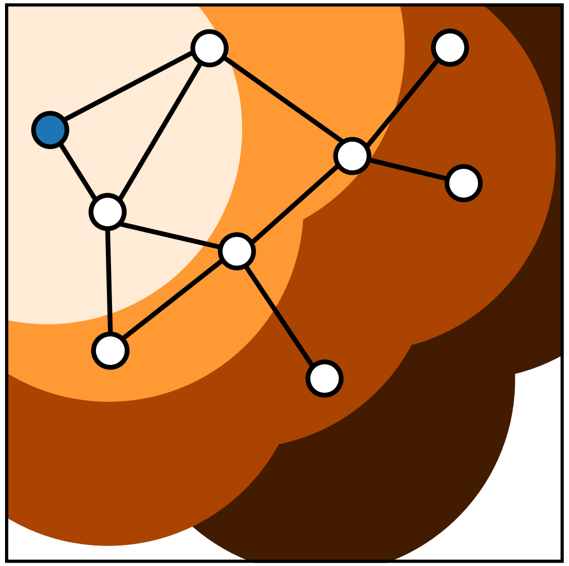


Predictable implementation

matching the scheduler's assumptions



All documents, software, data are openly available



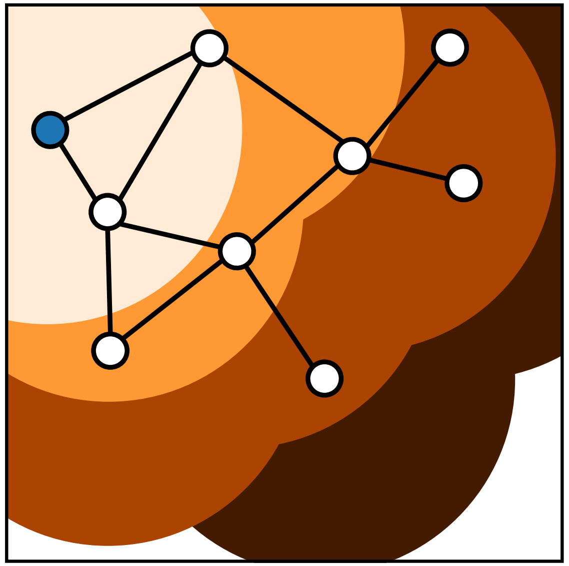
github.com/romain-jacob/TTW-Artifacts

zenodo

DOI 10.5281/zenodo.3835478

Many thanks for offering an Artifact Evaluation!

The Time-Triggered Wireless Architecture



Project webpage

ttw.ethz.ch

Romain Jacob
Licong Zhang
Marco Zimmerling
Jan Beutel
Samarjit Chakraborty
Lothar Thiele

ETH Zurich
TU Dresden
University of North Carolina at Chapel Hill

www.romainjacob.net
@RJacobPartner