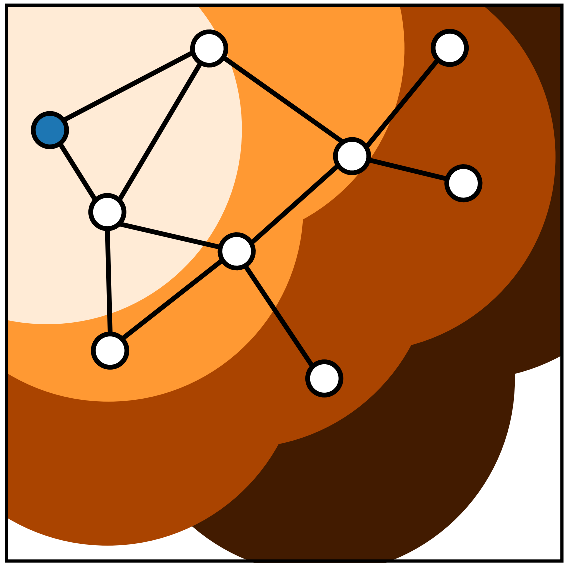
Leveraging Synchronous Transmissions for the Design of Real-time Wireless Cyber-Physical Systems



Romain Jacob

Doctoral Examination

December 17, 2019

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

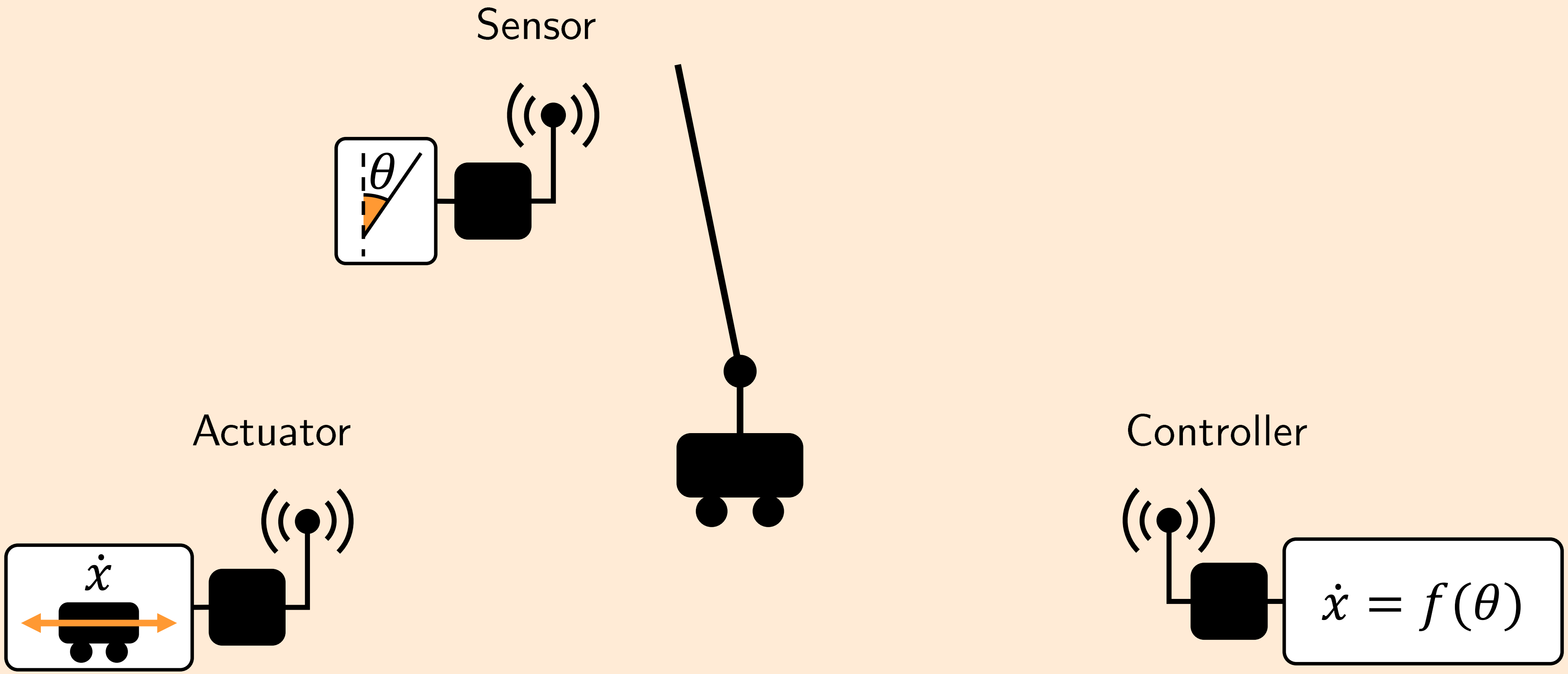
Fast and reliable

Mobile

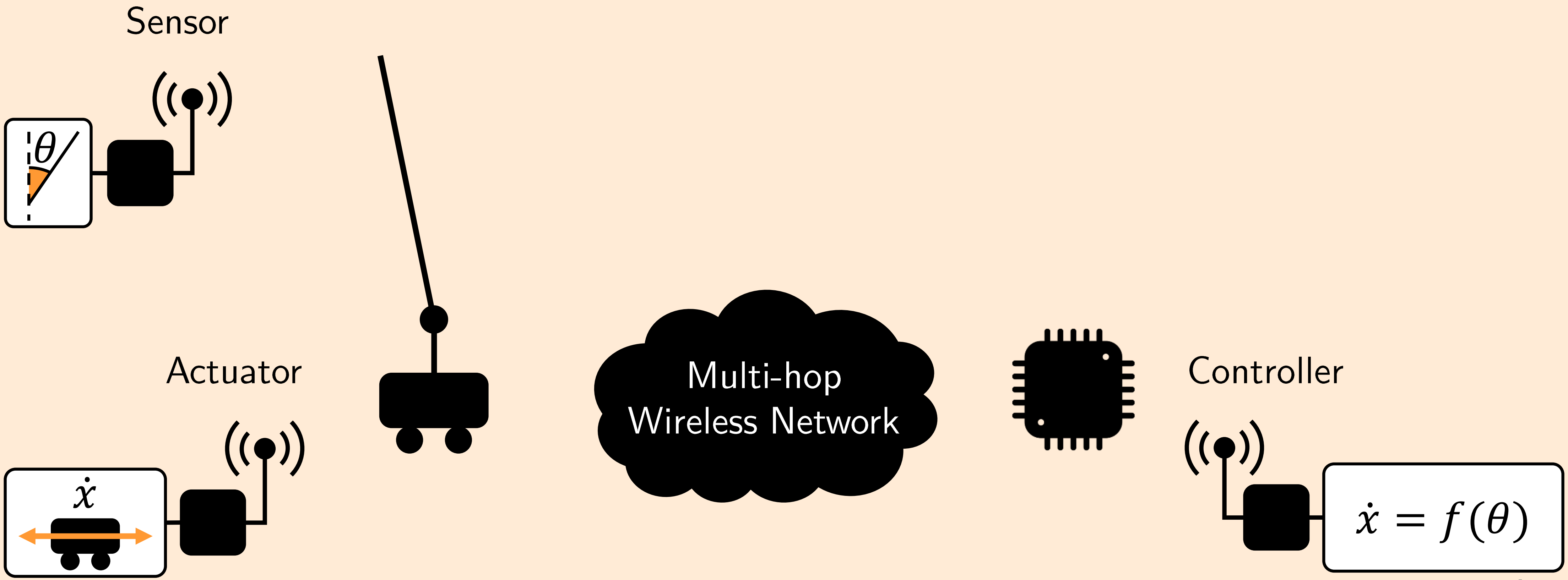
Scalable

Representative example

Stabilizing an inverted pendulum



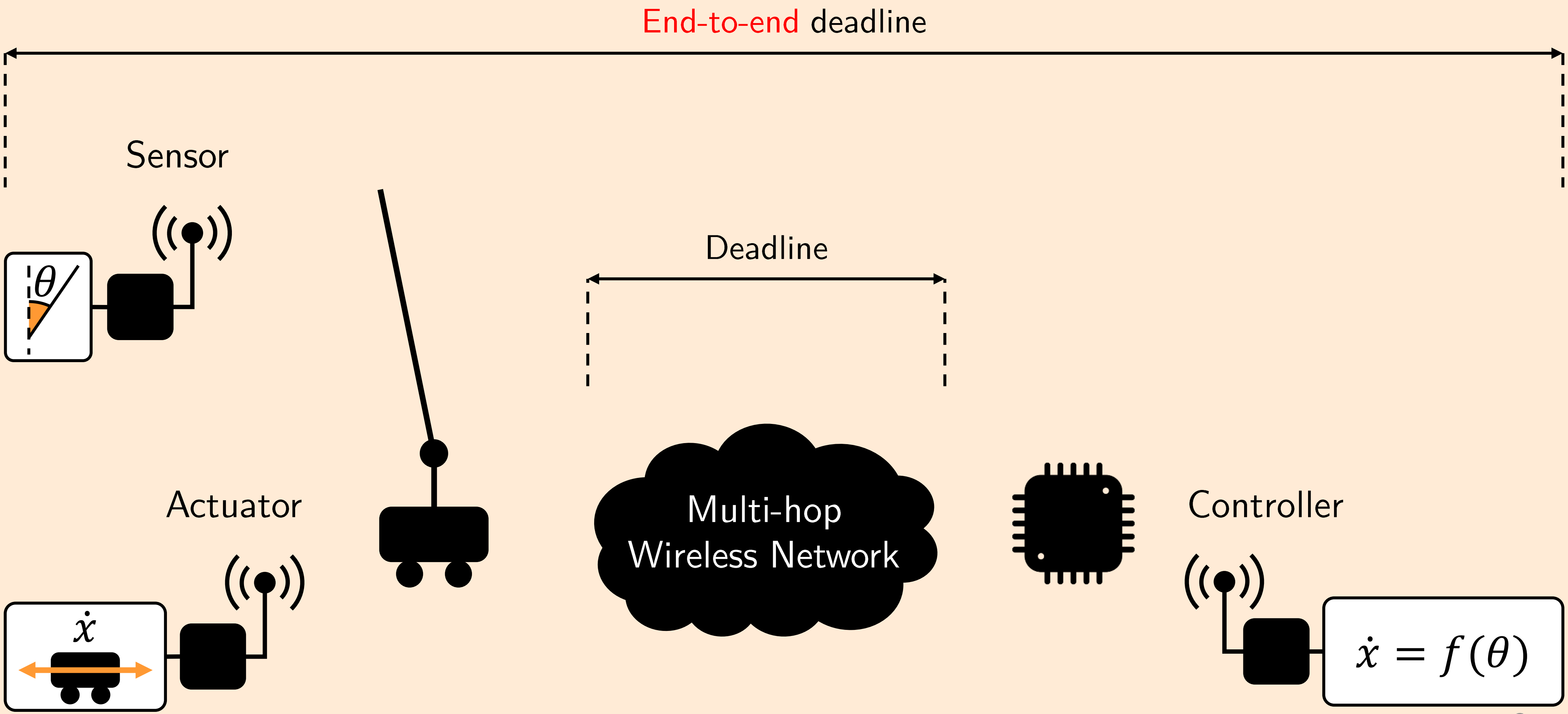
Things get more complex when the controller is physically separated from the pendulum



The application sets 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
Sensor
θ
Deadline
Actuator
$\dot{x}$
Multi-hop
Wireless Network
Controller
$\dot{x} = f(\theta)$

Existing wireless system designs are either predictable or adaptive

		For example
	Adaptive	Splash
not	Predictable	Staffeta
	Predictable	WirelessHART
not	Adaptive	TSCH

and Nothing proposed
to provide end-to-end
real-time guarantees

Existing wireless system designs are either predictable or adaptive

		For example
	Adaptive	Splash
not	Predictable	Staffeta
	Predictable	WirelessHART
not	Adaptive	TSCH

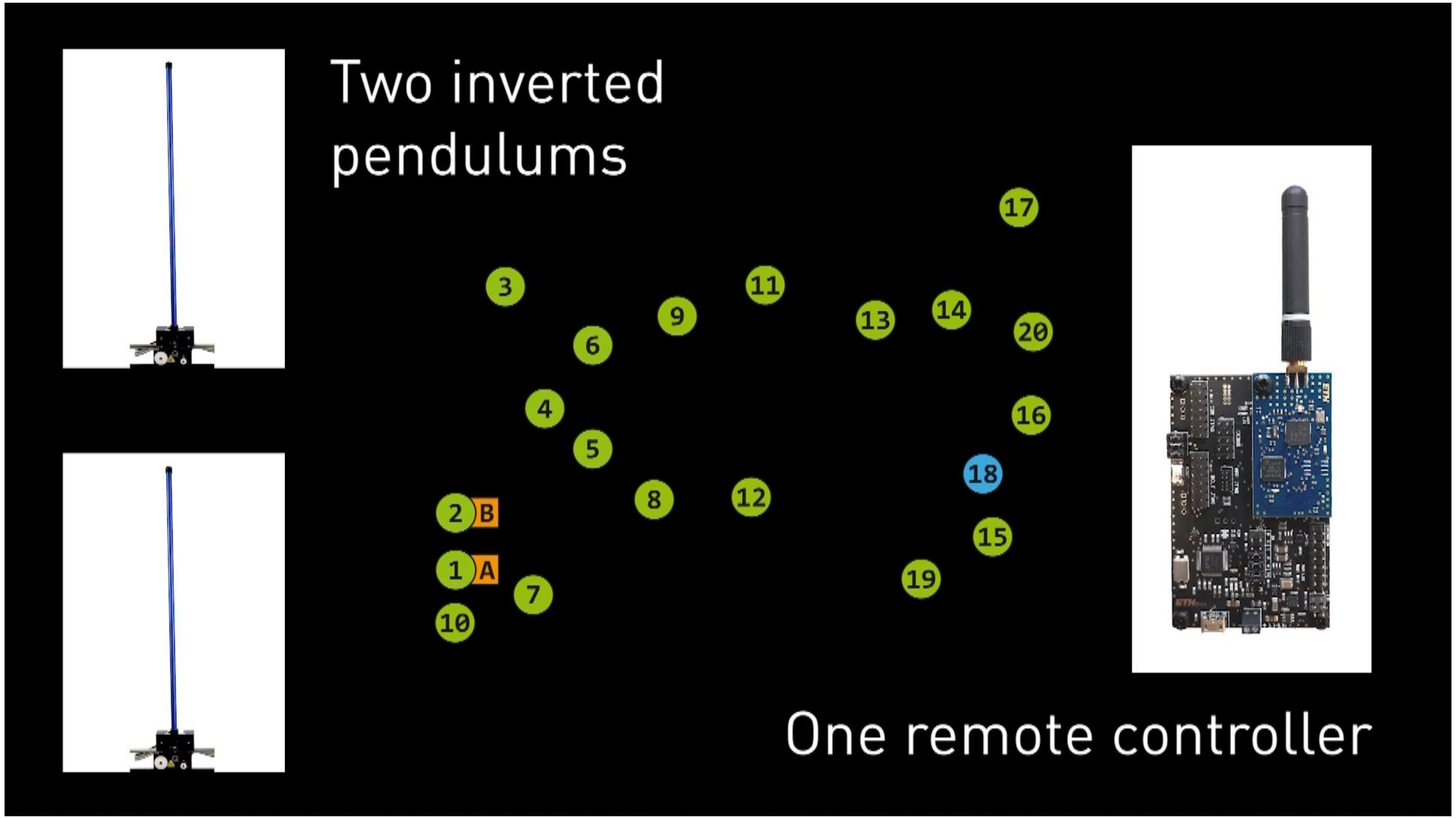
and

Nothing proposed
to provide end-to-end
real-time guarantees

Dissertation objective

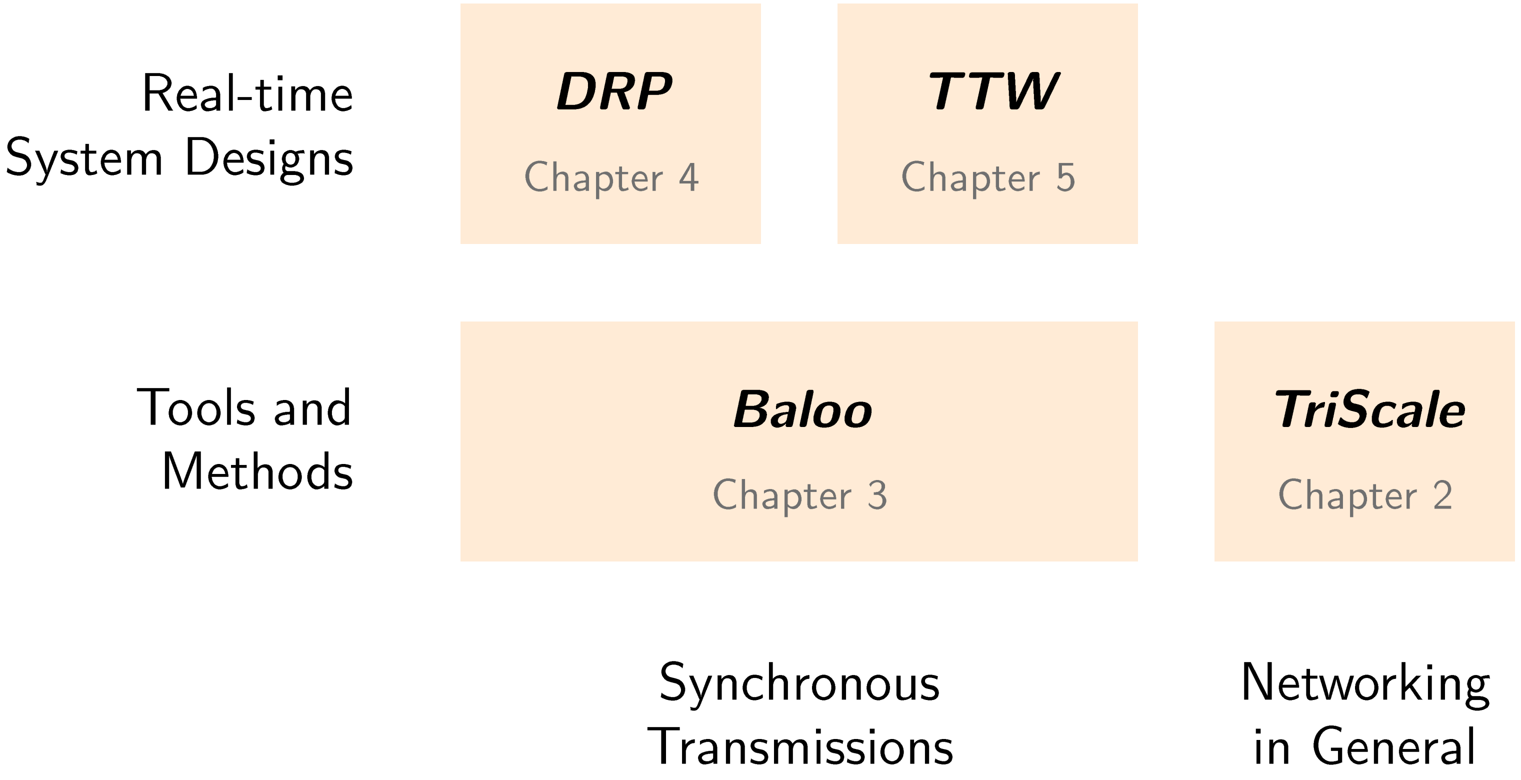
Design wireless systems combining **end-to-end predictability** and **adaptability** while remaining **efficient**

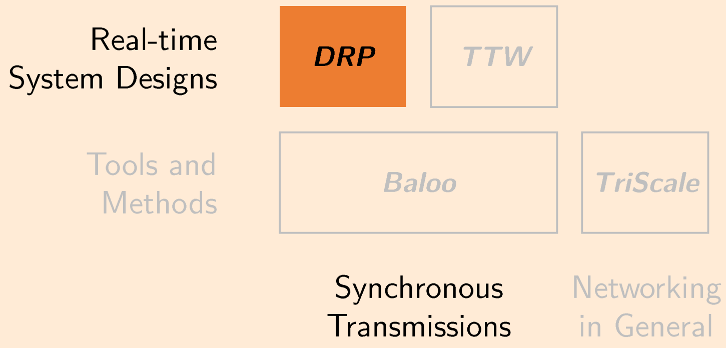
Guaranteed Stability over Low-power Multi-hop Network



tiny.cc/WirelessCPSVideo

Synchronous Transmissions are well suited to design real-time wireless CPS.


Real-time
System Designs
DRP
Chapter 4
TTW
Chapter 5
Tools and
Methods
Baloo
Chapter 3
TriScale
Chapter 2
Synchronous
Transmissions
Networking
in General



Chapter 4

DRP: End-to-end Real-time Guarantees in Wireless Cyber-Physical Systems

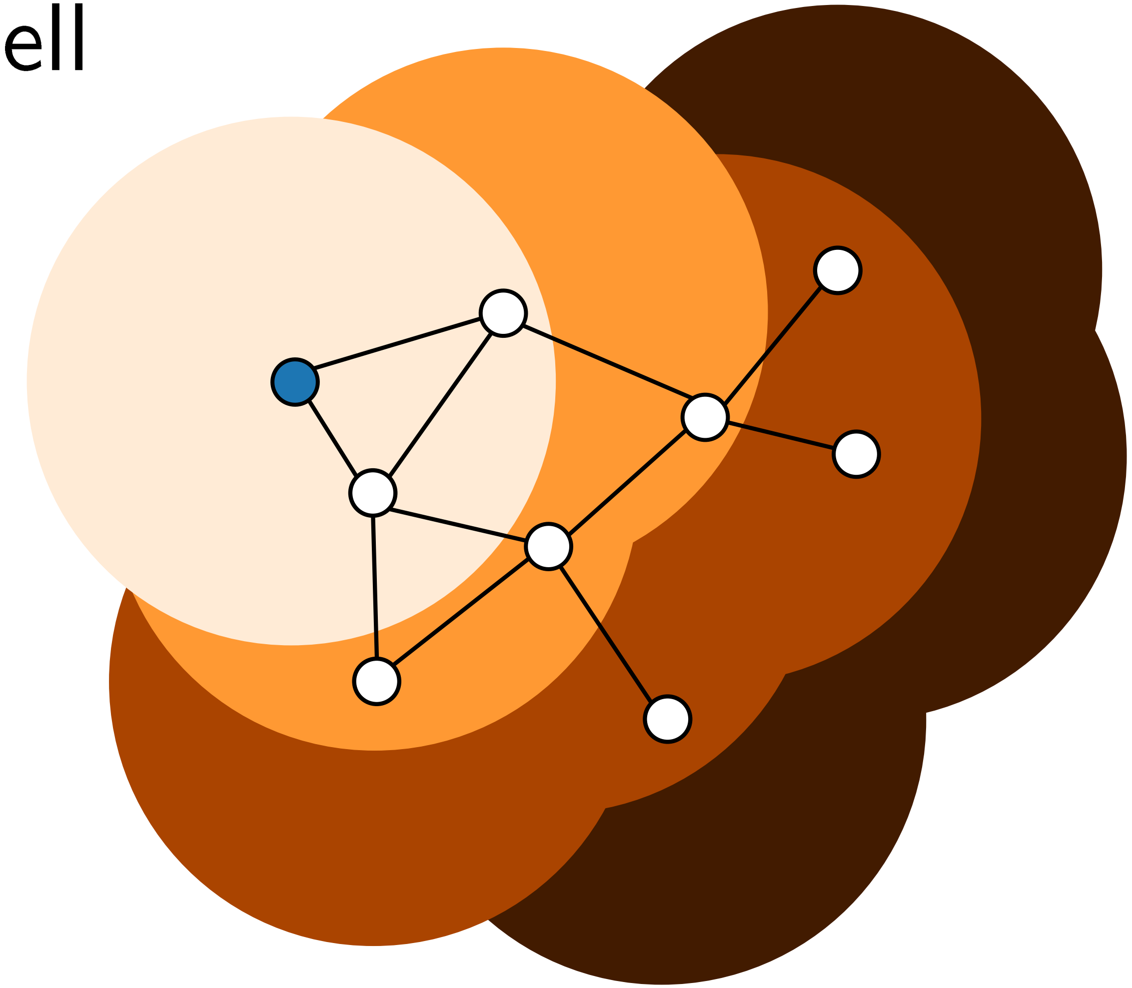
End-to-End Real-Time Guarantees in Wireless Cyber-Physical Systems
R. Jacob, M. Zimmerling, P. Huang, J. Beutel, L. Thiele, 2016

Chapter 5

TTW: A Time-Triggered Design for Wireless Cyber-Physical Systems

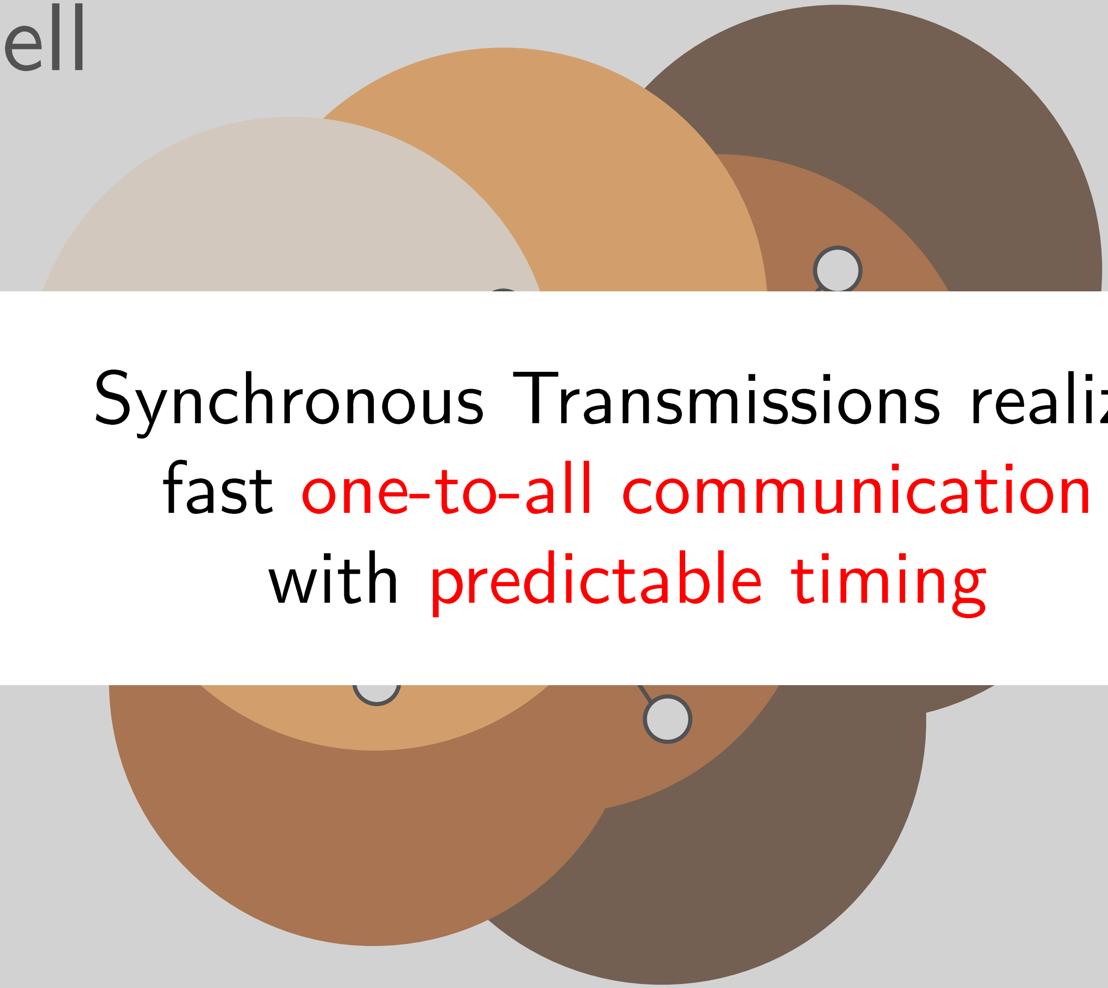
TTW: A Time-Triggered Wireless design for CPS
R. Jacob, L. Zhang, M. Zimmerling, J. Beutel, S. Chakraborty, L. Thiele, 2018

Synchronous Transmissions in a nutshell



Multi-hop broadcast based on flooding

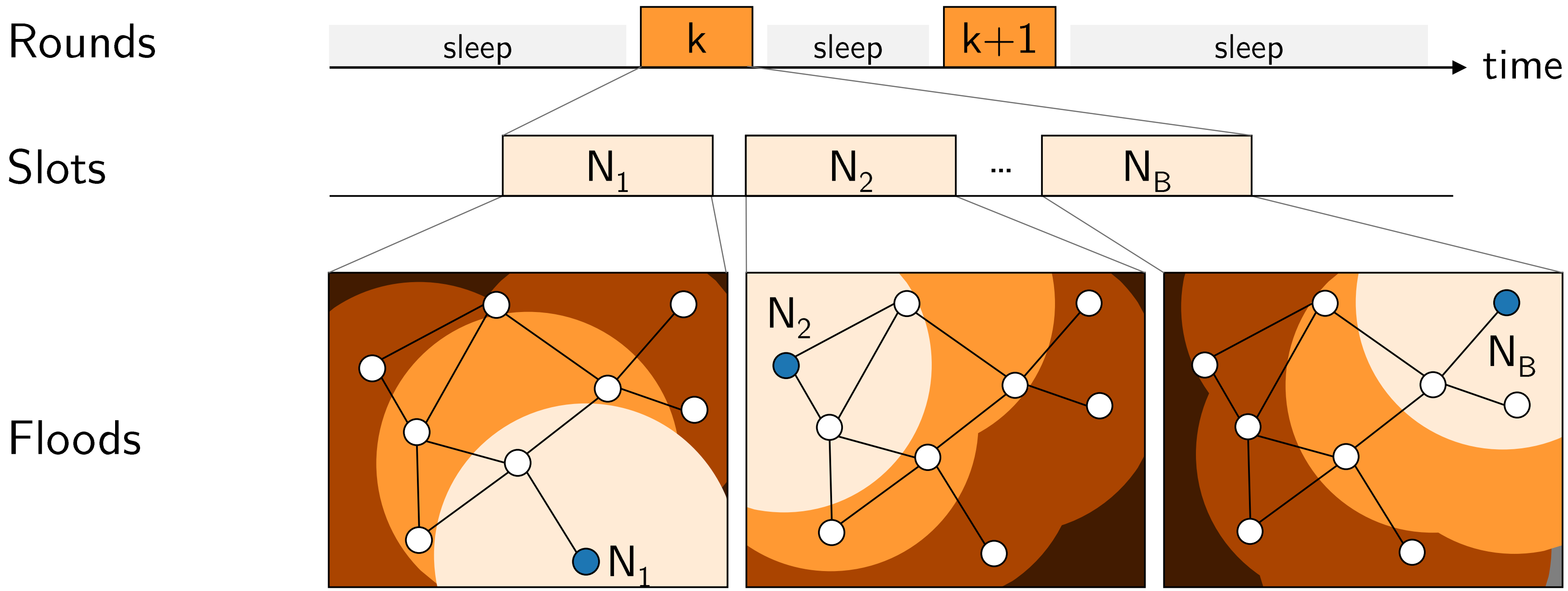
Synchronous Transmissions in a nutshell



Synchronous Transmissions realize fast one-to-all communication with predictable timing

Multi-hop broadcast based on flooding

Typical network stack based on Synchronous Transmissions



System must support

Primary goals

Predictability

End-to-end real-time guarantees

Adaptability

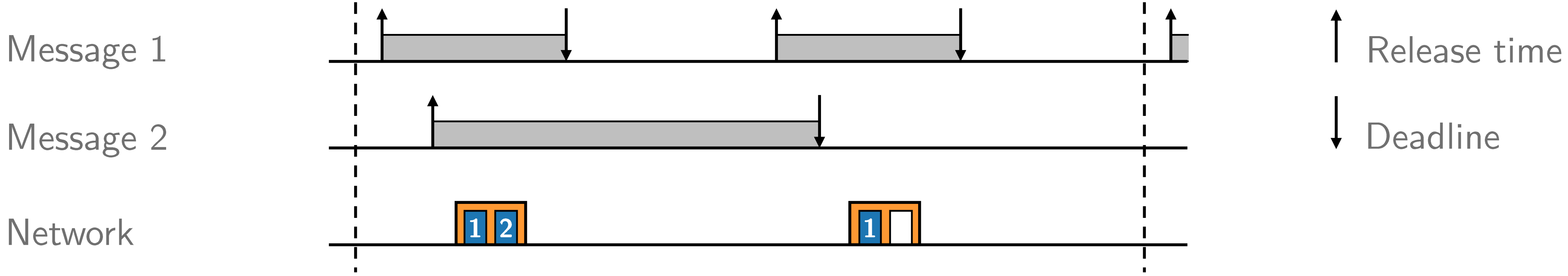
- Change in traffic demand
- Mobility

Secondary objectives

Efficiency

- Low latency
- Low energy consumption

Real-time scheduling allows to meet deadlines

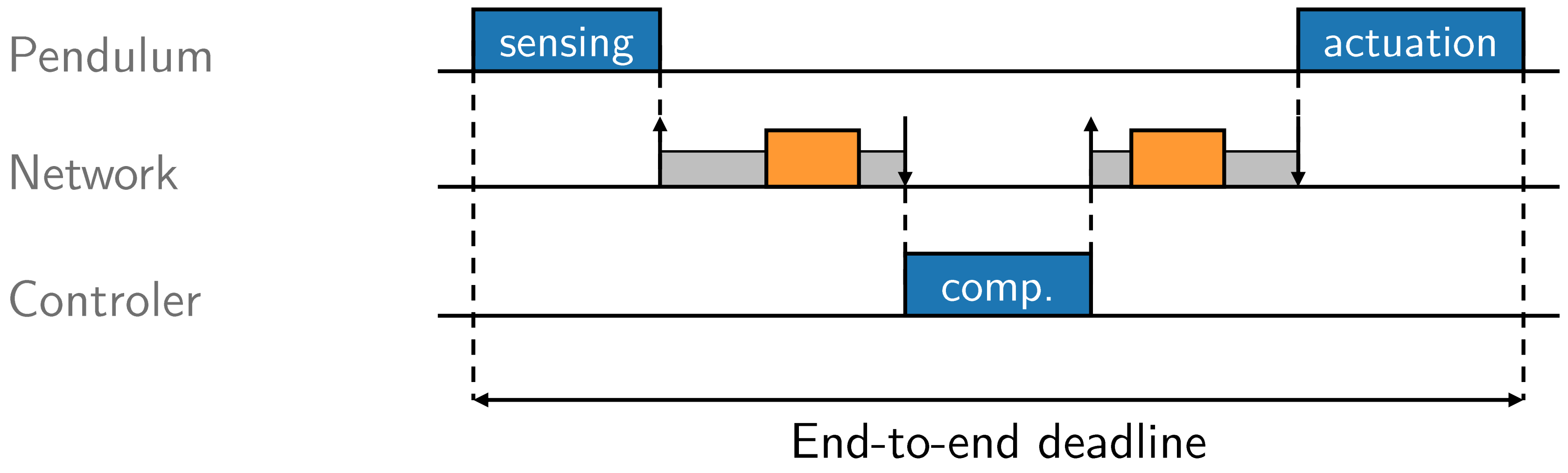


In practice

How do you get the message deadlines?

From the execution time of the tasks!

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



End-to-end guarantees requires 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
10	Slots per round

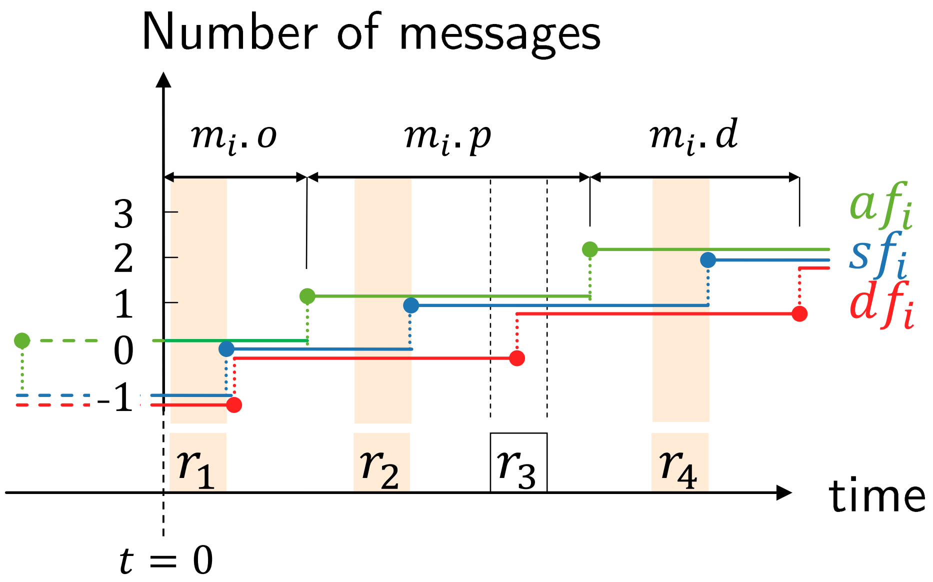
$\Rightarrow$ Reduction of $\sim$ 30% in radio-on time

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

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

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

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

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

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

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

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

System must support

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

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

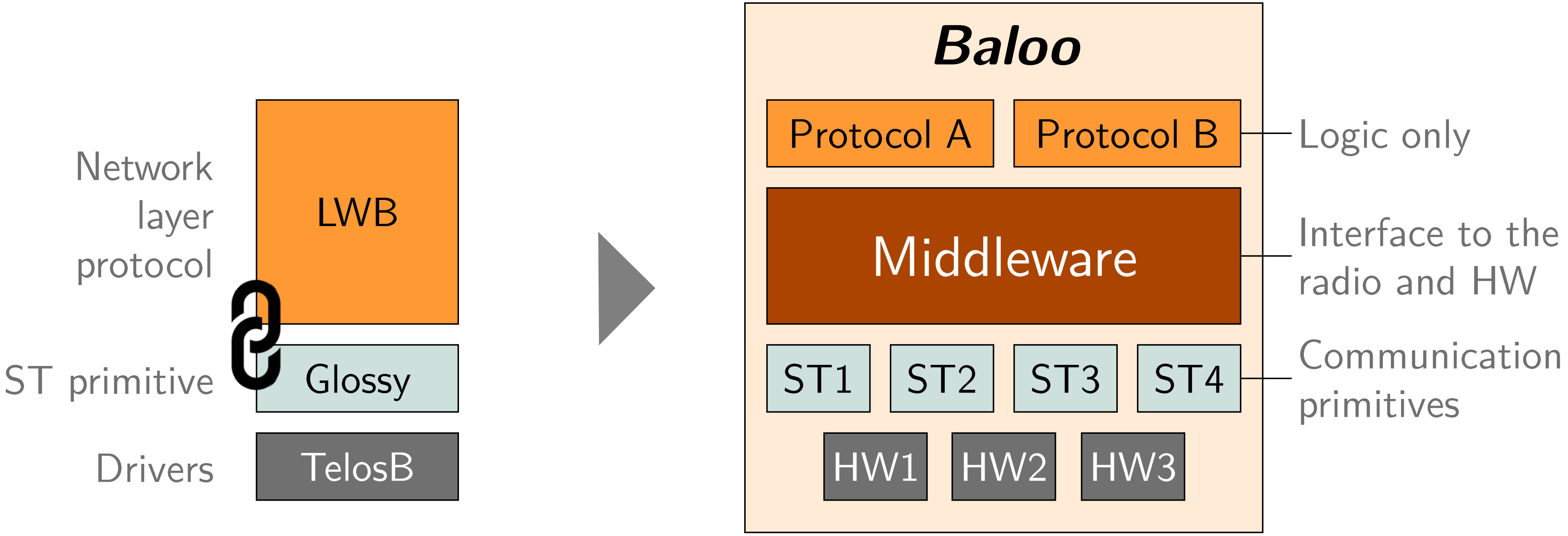
- Can we implement this?
- Does it really work?

Chapter 3

Synchronous Transmissions Made Easy: Design Your Network Stack with Baloo

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

Baloo facilitates the design of ST-based stacks by decoupling the protocol logic and the radio control



Monolithic stack design **Flexible framework**

Real-time System Designs	*DRP*	*TTW*	
Tools and Methods	*Baloo*		***TriScale***
	Synchronous Transmissions		Networking in General

Chapter 2

TriScale: A Framework Supporting Reproducible Networking Experiments

Towards a Methodology for Experimental Evaluation in Low-Power Wireless Networking
R. Jacob, C. Boano, U. Raza, M. Zimmerling, L. Thiele, 2019

TriScale: A Framework Supporting Reproducible Networking Evaluations
R. Jacob, M. Zimmerling, C. Boano, L. Vanbever, L. Thiele, 2019

What does reproducibility mean?

Reproducibility
(unformal)

An experiment is said reproducible if the "same results" are obtained when the experiment is reproduced under the "same conditions".

Reproducibility
(unformal)

An experiment is said reproducible if the “same results” are obtained when the experiment is reproduced under the “same conditions”.

- Real RF environment cannot be controlled.
- The RF environment affects performance of protocols in ways that are hard to predict.

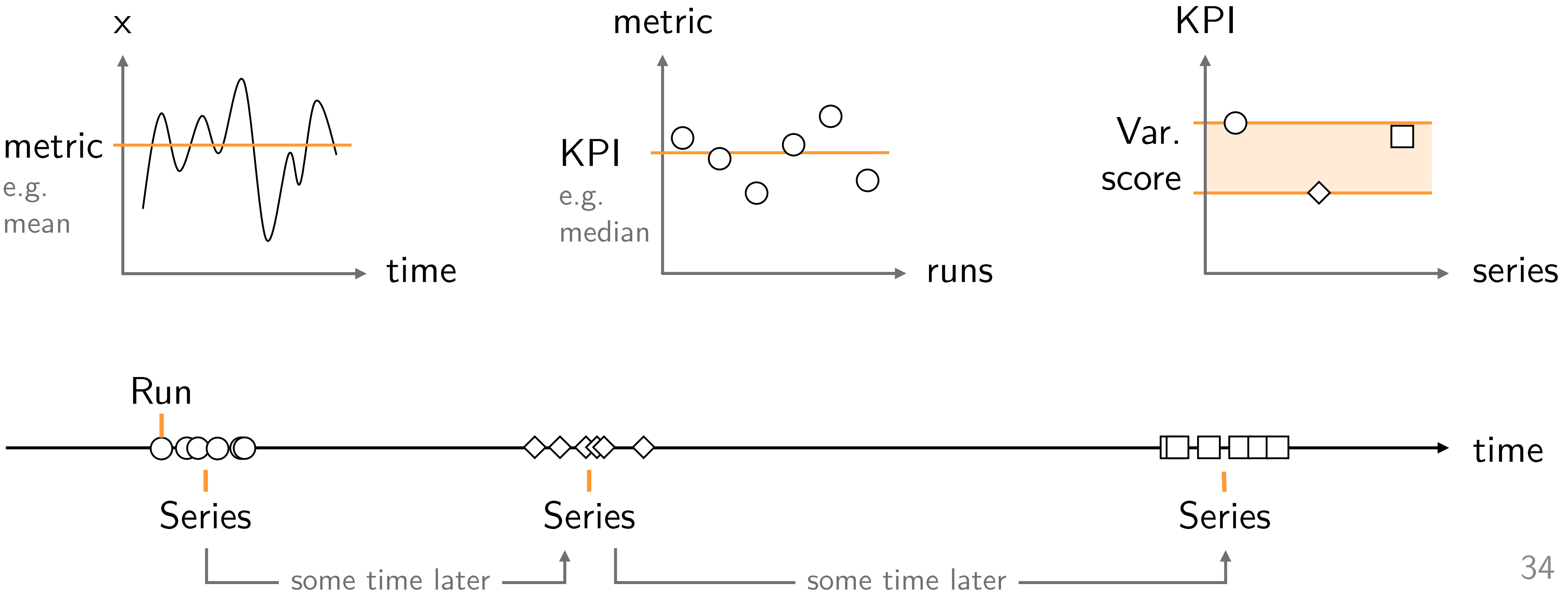
Reproducibility
(unformal)

An experiment is said reproducible
if the “same results” are obtained
when the experiment is reproduced
under the “same conditions”.

- Real RF environment cannot be controlled.
- The RF environment affects performance of protocols in ways that are hard to predict.

⇒ Performance variability is expected.

Performance varies along three different time scales



The four questions of experiment design

How long should a run be?

How many runs in a series?

What time span for a series?

How many series?

The four questions of experiment design

How long should a run be?

How many runs in a series?

What time span for a series?

How many series?

Objective

Find rationale answers to these questions

Quantify the trade-off between

- experiment effort
- confidence in the results

The four questions of experiment design

How long should a run be?

How many runs in a series?

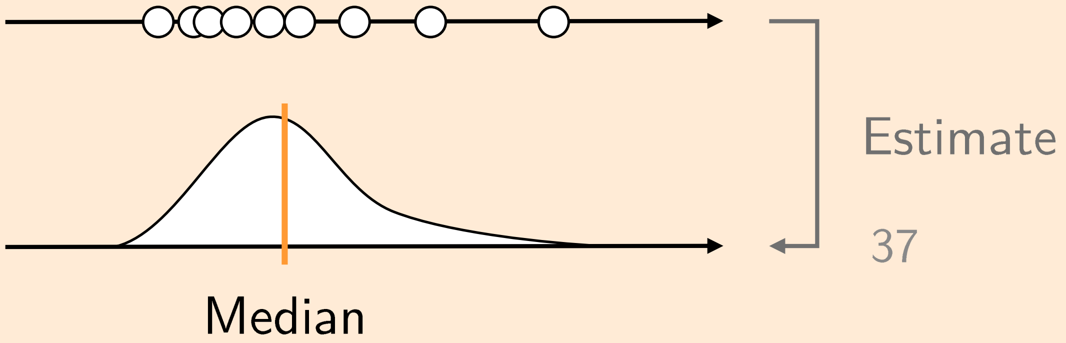
What time span for a series?

How many series?

Objective

The KPI would be the same shall more runs be performed

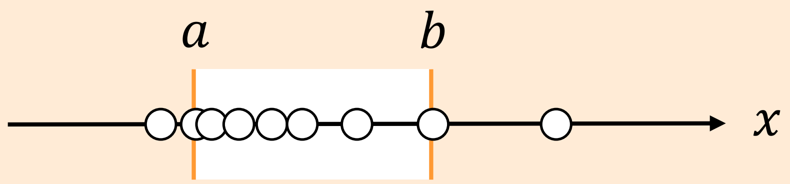
Based on a few runs, estimates the KPI value of the underlying distribution



We must use confidence intervals (CI)

Informally

Numerical interval
in which lies
the true value
(which you don't known)
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

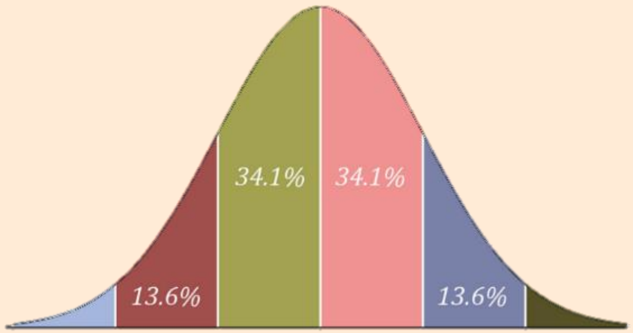
The probability that the true median of x is within $[a, b]$ is larger or equal to 95%.

We must use non-parametric statistics

Makes no assumption on
the nature of the distributions

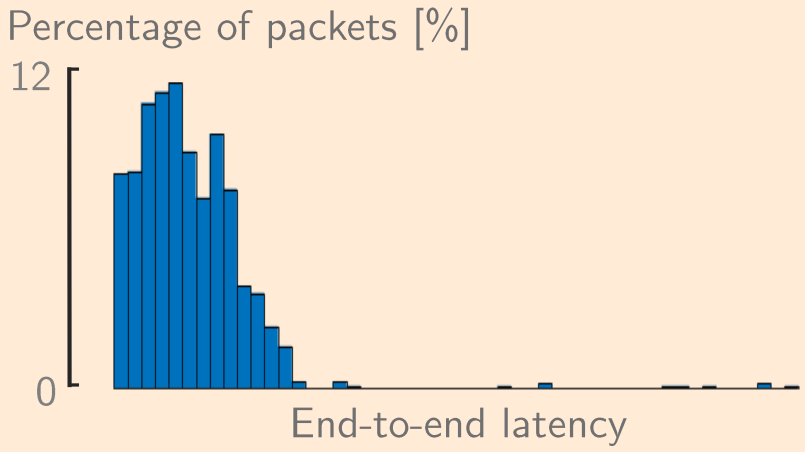
	Common KPIs
Tendency	Mean
Spread	Standard deviation
In computer science	Experimental data is almost never normally distributed

Estimate



Only valid for a
normal distribution!!

Normal



Not normal

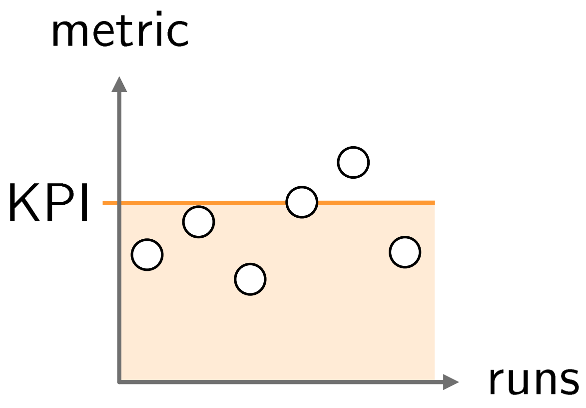
The Thompson's methods provides non-parametric CI for distribution percentiles

In TriScale

Key Performance Indicators (KPIs) are percentiles of the distribution of metric values

Compute confidence intervals using Thompson's method

- Define upper- and lower-bounds on the true percentile values for a certain confidence level
- The KPIs are defined as one such bounds



95%-CI for the median

The Thompson's methods provides non-parametric CI for distribution percentiles

Probability of any P_p to be between two consecutive samples follows the binomial distribution

$$\mathsf{P}\left\{x_k \leq P_p \leq x_{k+1}\right\} = \binom{N}{k} p^k (1-p)^{N-k}$$

Lower-bound

$$\mathsf{P}\left\{x_m \leq P_p\right\} =$$

Upper-bound

$$P\left\{x_{N-m+1} \geq P_{1-p}\right\} = 1 - \sum_{k=0}^{m-1} \binom{N}{k} p^k (1-p)^{N-k}$$

The Thompson's methods provides non-parametric CI for distribution percentiles

For any confidence c

For any percentile P_p

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

$c = 0{,}95$

$p = 0{,}01$

1-th percentile

$\Rightarrow \quad N \geq 299$

Minimal number of runs in a series

We might want to rethink our performance claims...

The four questions of experiment design

How long should a run be?

How many runs in a series?

What time span for a series?

How many series?

Find rationale answers
to these questions

Quantify the trade-off between

- experiment effort
- confidence in the results

Example – Performance evaluation of TTW
Is the model providing a safe and tight upper-bound?

Metric

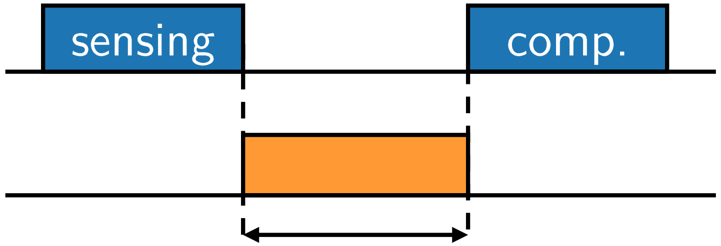
Maximum measured round time T_r among all nodes

KPI

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

KPI

95^{th} percentile

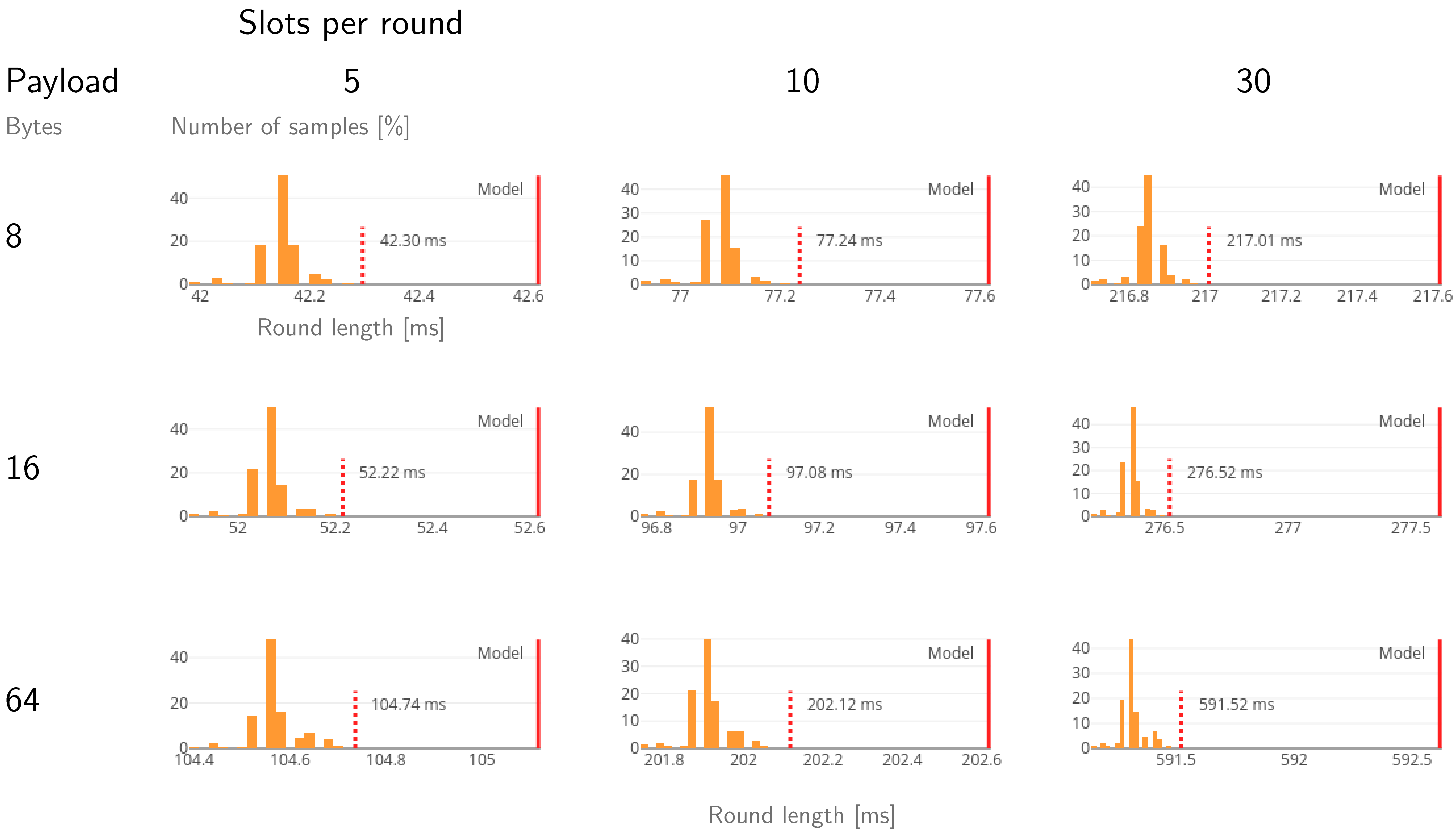
95% confidence level

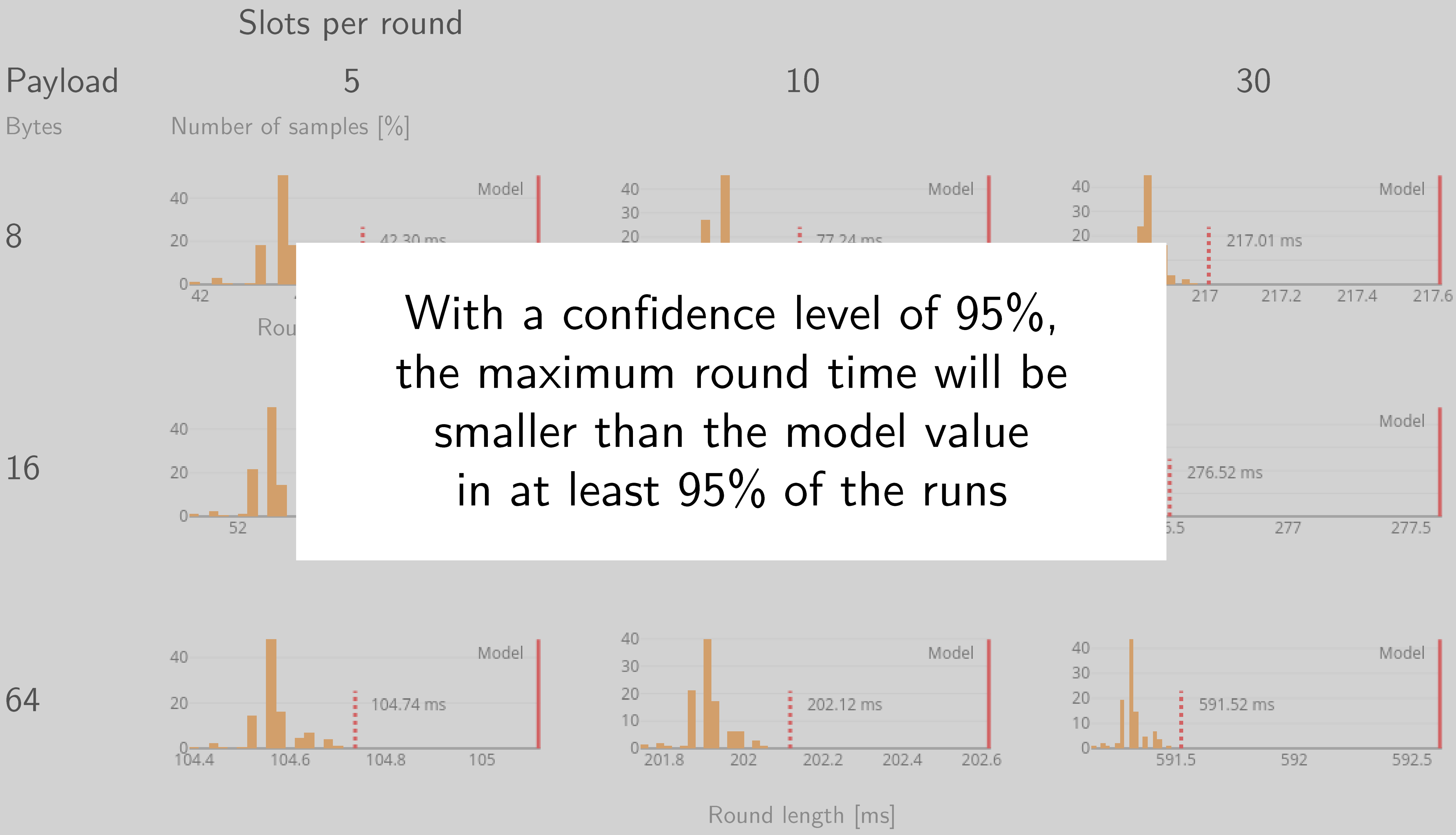
Aim to upper-bound the 95^{th} percentile of the maximum round time T_r with 95% confidence

Minimal number of runs in a series

$$c = 0{,}95 \quad p = 0{,}05 \quad \Rightarrow \quad N \geq 59$$

$\Rightarrow$ 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 maximum round time will be smaller than the model value in at least 95% of the runs

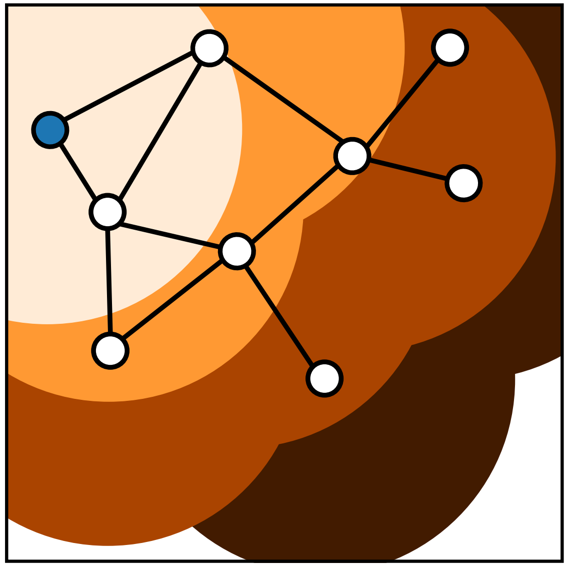
TriScale is a framework helping to design and analyze networking experiments

Implemented as	Python package
	Open source

Includes

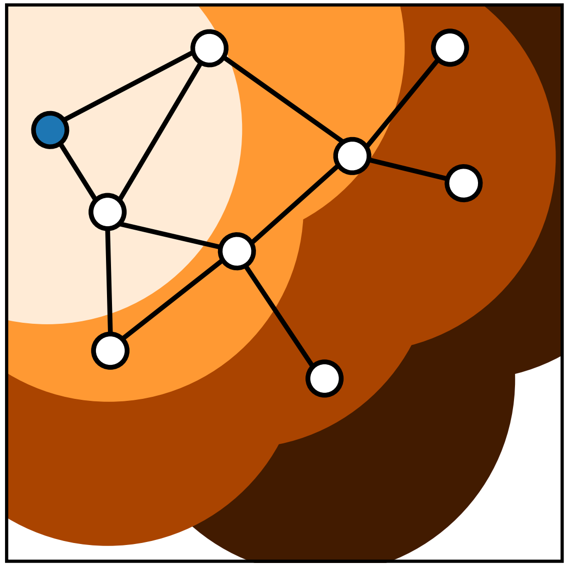
- Statistical methods and tests
- Experiment design helpers
- Automated data processing
- Dynamic visualizations

Leveraging Synchronous Transmissions for the Design of Real-time Wireless Cyber-Physical Systems



Real-time System Designs	***DRP*** Chapter 4 / ***TTW*** Chapter 5	
Tools and Methods	***Baloo*** Chapter 3	***TriScale*** Chapter 2
	Synchronous Transmissions	Networking in General

All documents, software, data are openly available



Links in the respective chapters

github.com/romain-jacob/doctoral-thesis

zenodo

DOI 10.5281/zenodo.3510184

Thank you all for your support!

Supervisor
Prof. Thiele

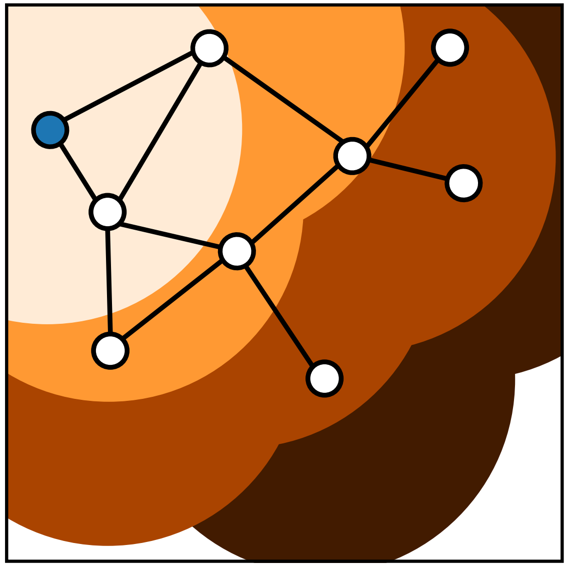
My wife
My friends
My parents

Co-authors
Jan Beutel
Reto Da Forno
Marco Zimmerling
Carlo Alberto Boano
Licong Zhang
Pengcheng Huang
Laurent Vanbever
Usman Raza
Samarjit Chakraborty
Jonas Bächli

TECies
Andreas Biri
Yun Cheng
Stefan Draskovic
Tonio Gsell
Xioxi He
Matthias Meyer
Philipp Miedl
Zhongnan Qu
Lukas Sigrist
Naomi Sticker
Roman Trüb
Balz Maag
Andres Gomez
Felix Sutton
Andreas Tretter
Georgia Giannopoulou
Roman Lim
Olga Saukh
Rehan Ahmed
Zimu Zhou

Master students
Jonas Bächli
Alex Dietmüller
Fabian Walter
Andreas Biri
Jonathan Candel
Antonios Koskinas
Jan Müller
Anna-Brit Schaper
Raphael Schnider

Leveraging Synchronous Transmissions for the Design of Real-time Wireless Cyber-Physical Systems



Romain Jacob

ETH Zurich

www.romainjacob.net

@RJacobPartner

The four questions of experiment design

How long should a run be? such that the metric would be the same shall the run be longer

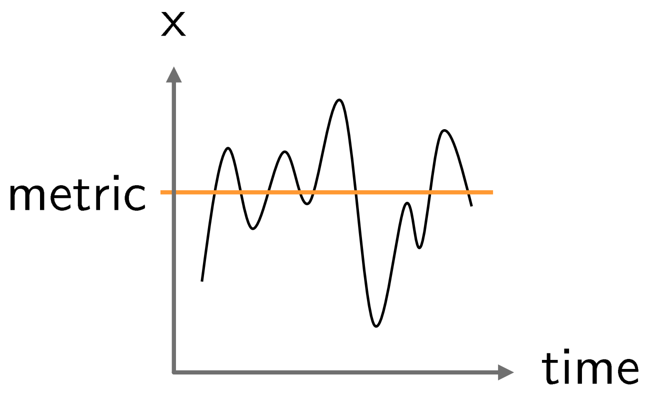
How many runs in a series?

What time span for a series?

How many series?

The Theil-Sen estimator performs robust and non-parametric linear regression

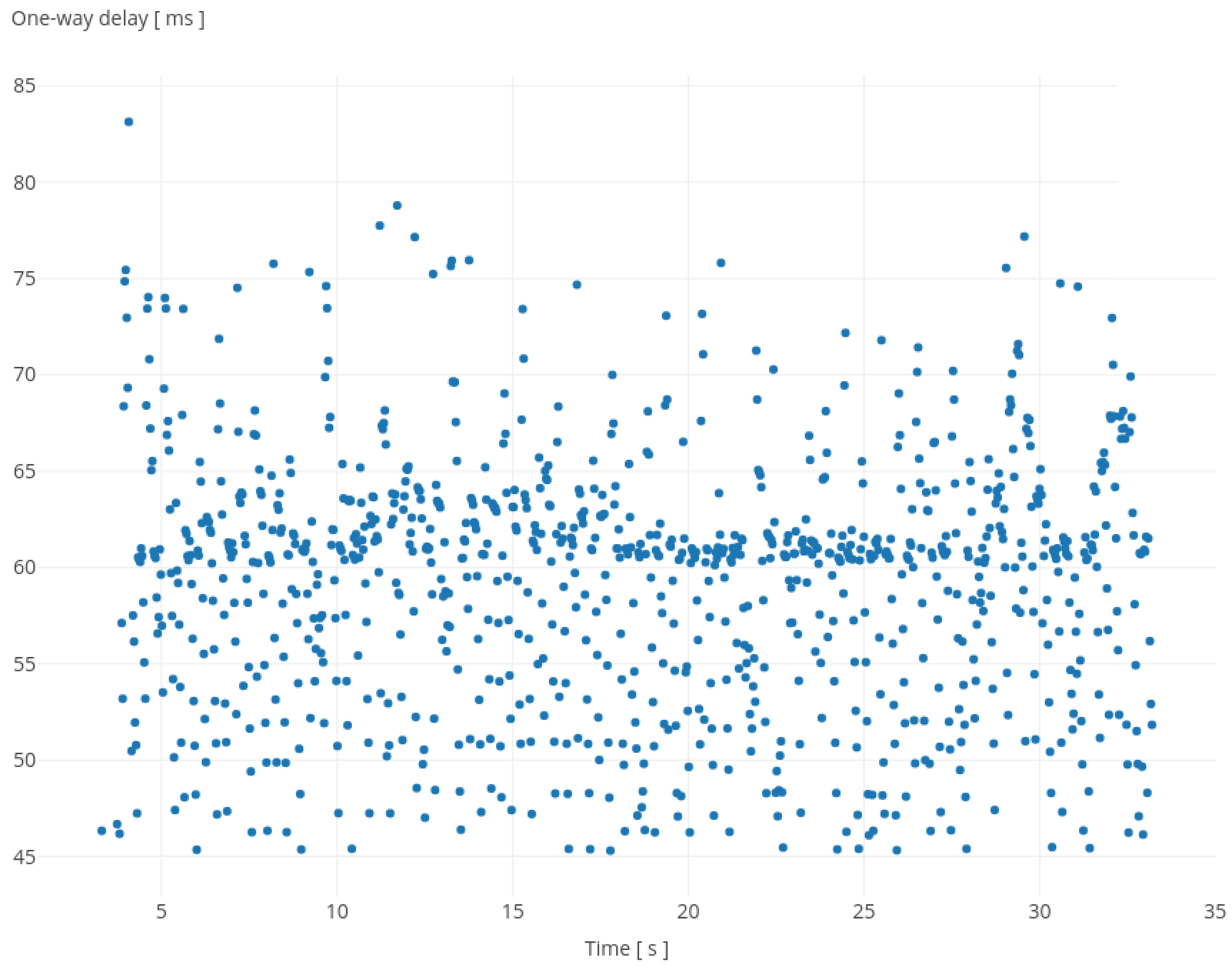
Computing a run's metric
is a regression problem

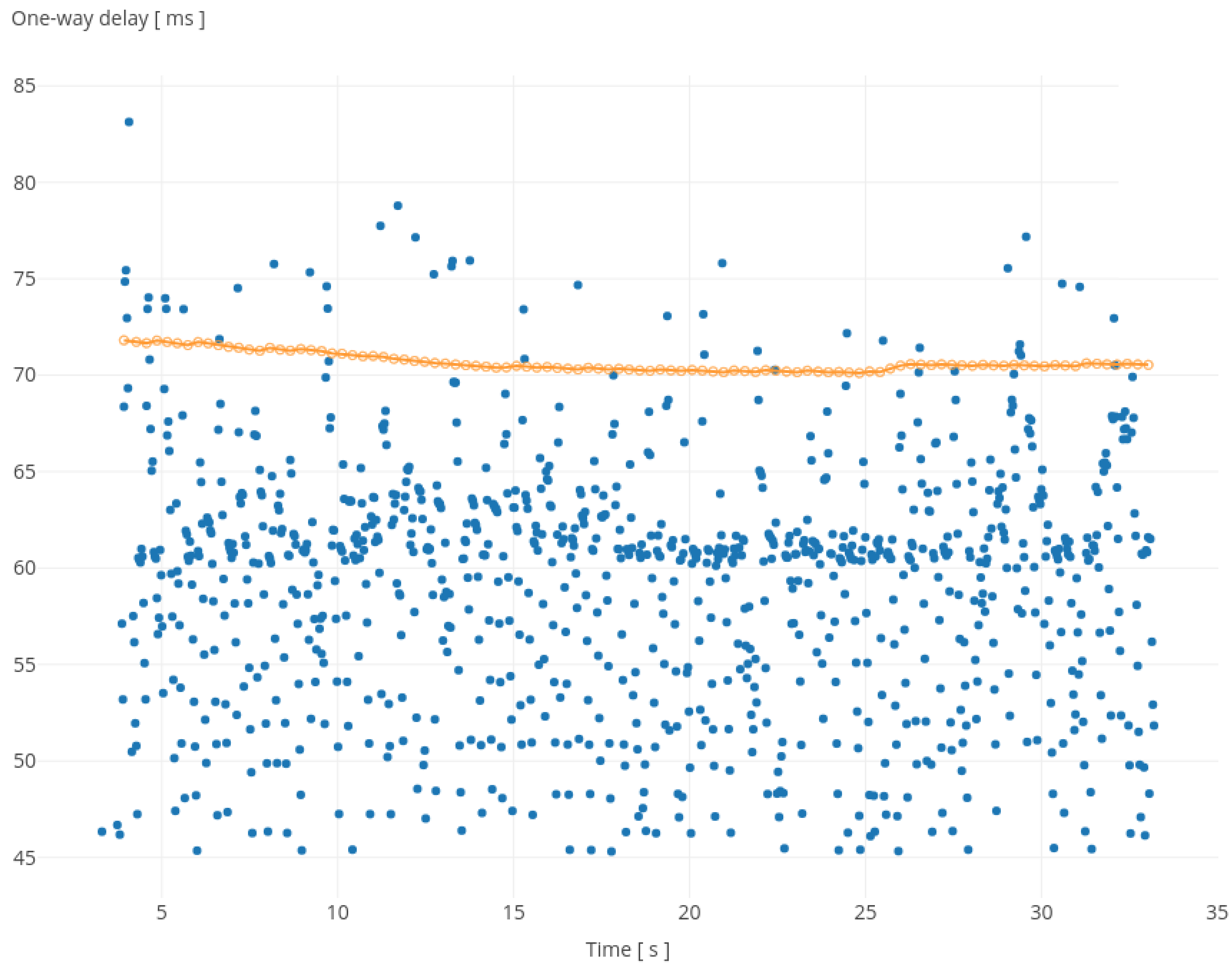


Non-parametric regression

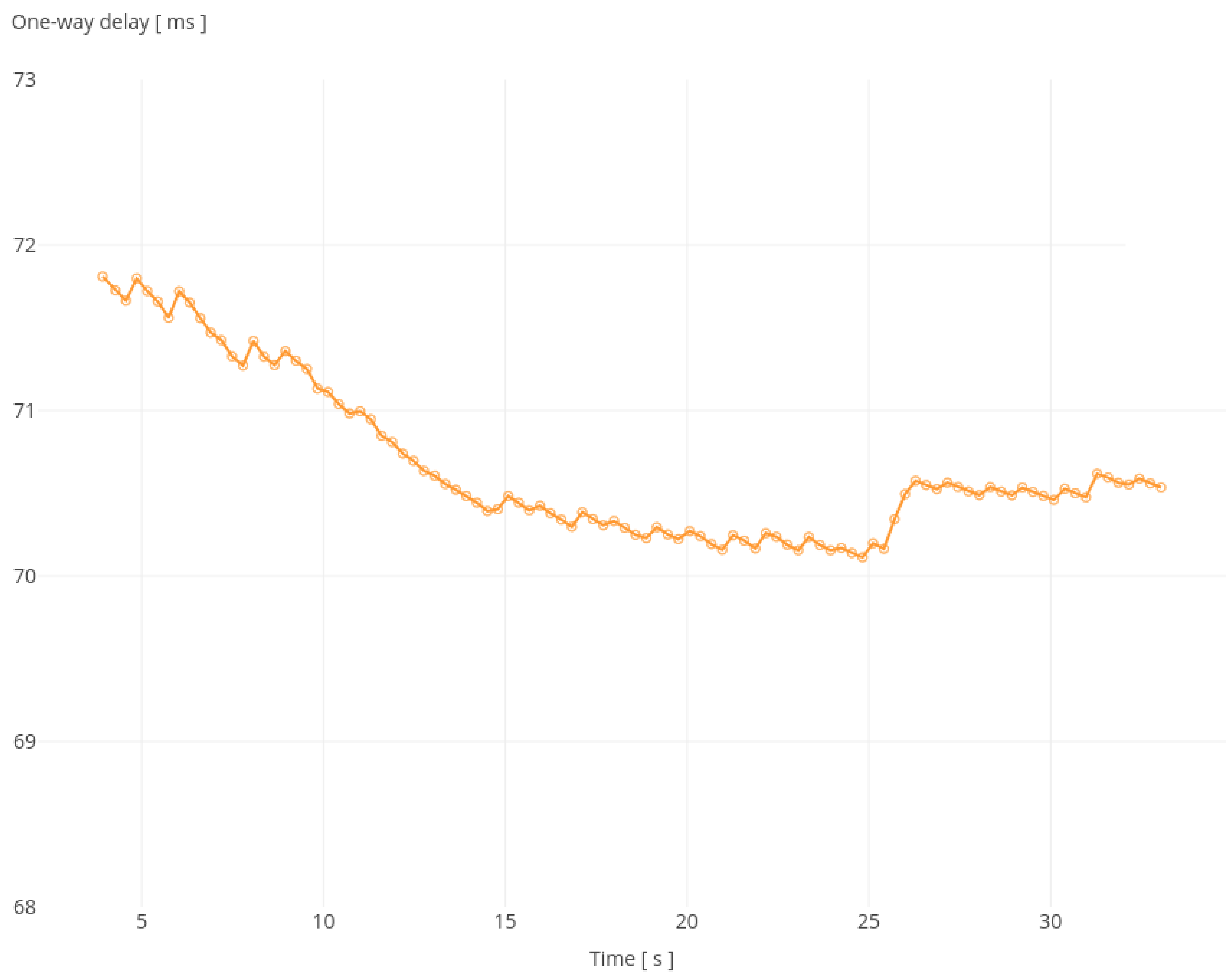
Theil-Sen estimator

- Slope is the median of all slopes between pairs of data points
- CI on the slope using the bootstrapping method


One-way delay [ms]
85
80
75
70
65
60
55
50
45
5
10
15
20
25
30
35
Time [s]

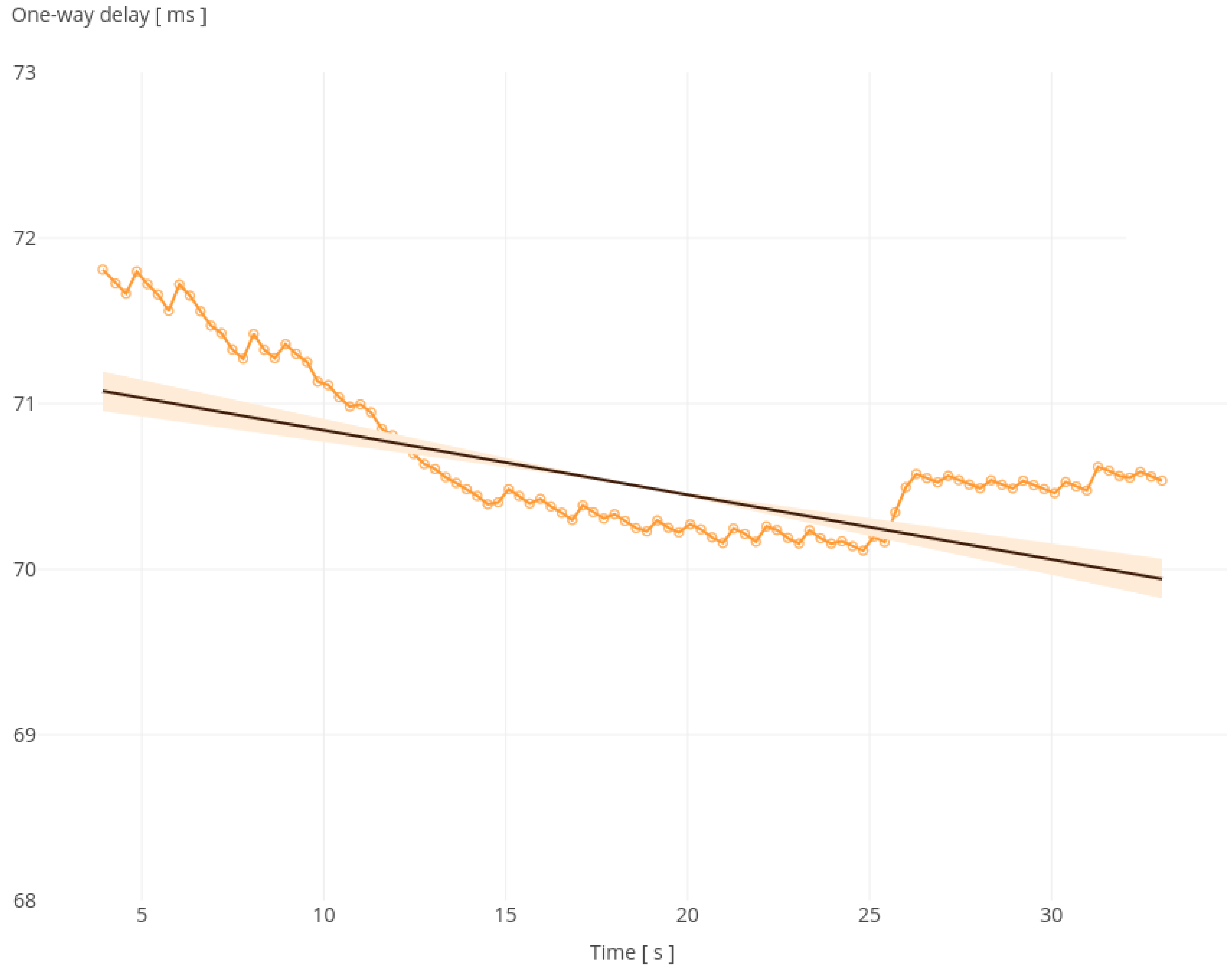


Compute the metric on a sliding window of data points

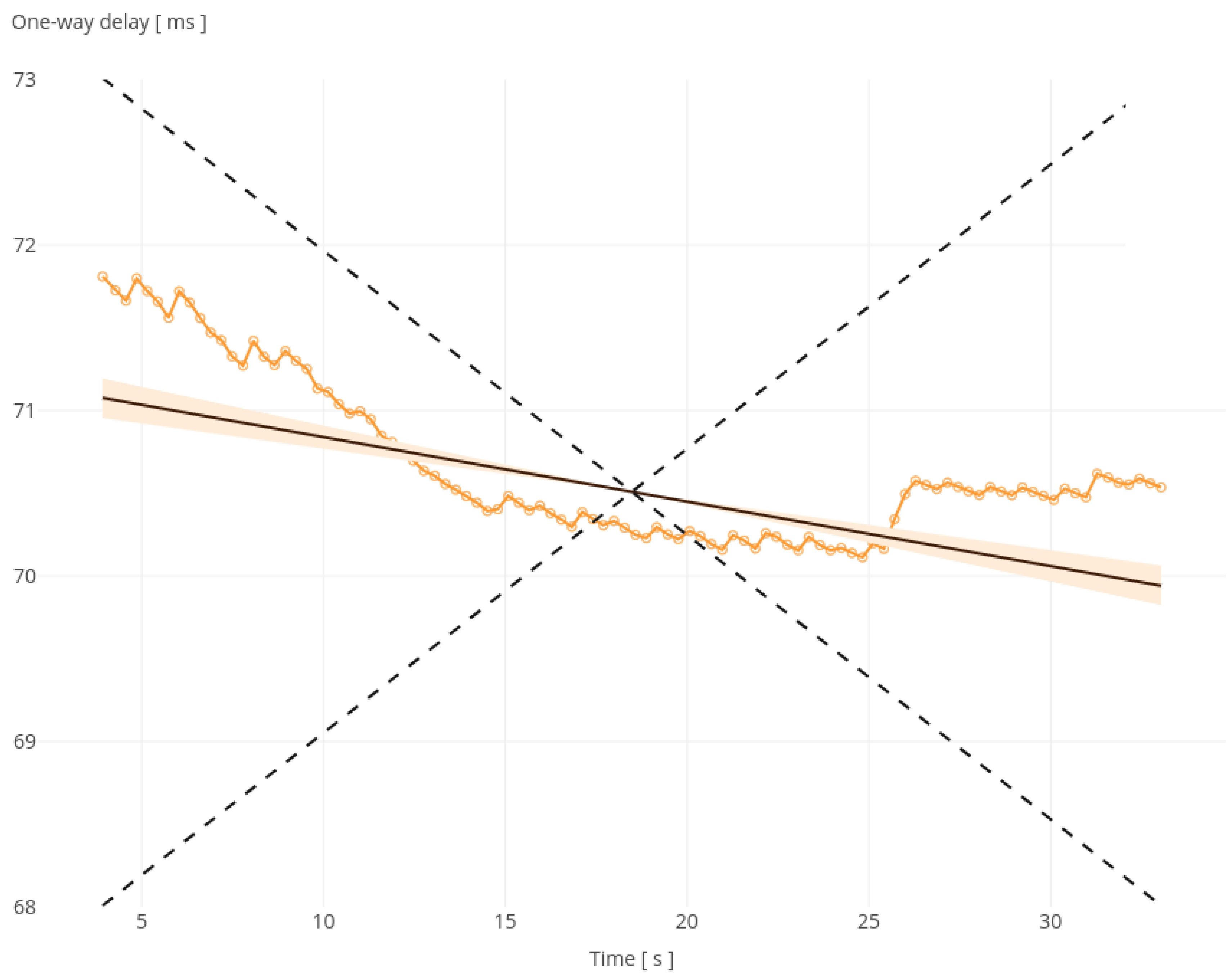


Compute the metric on a sliding window of data points

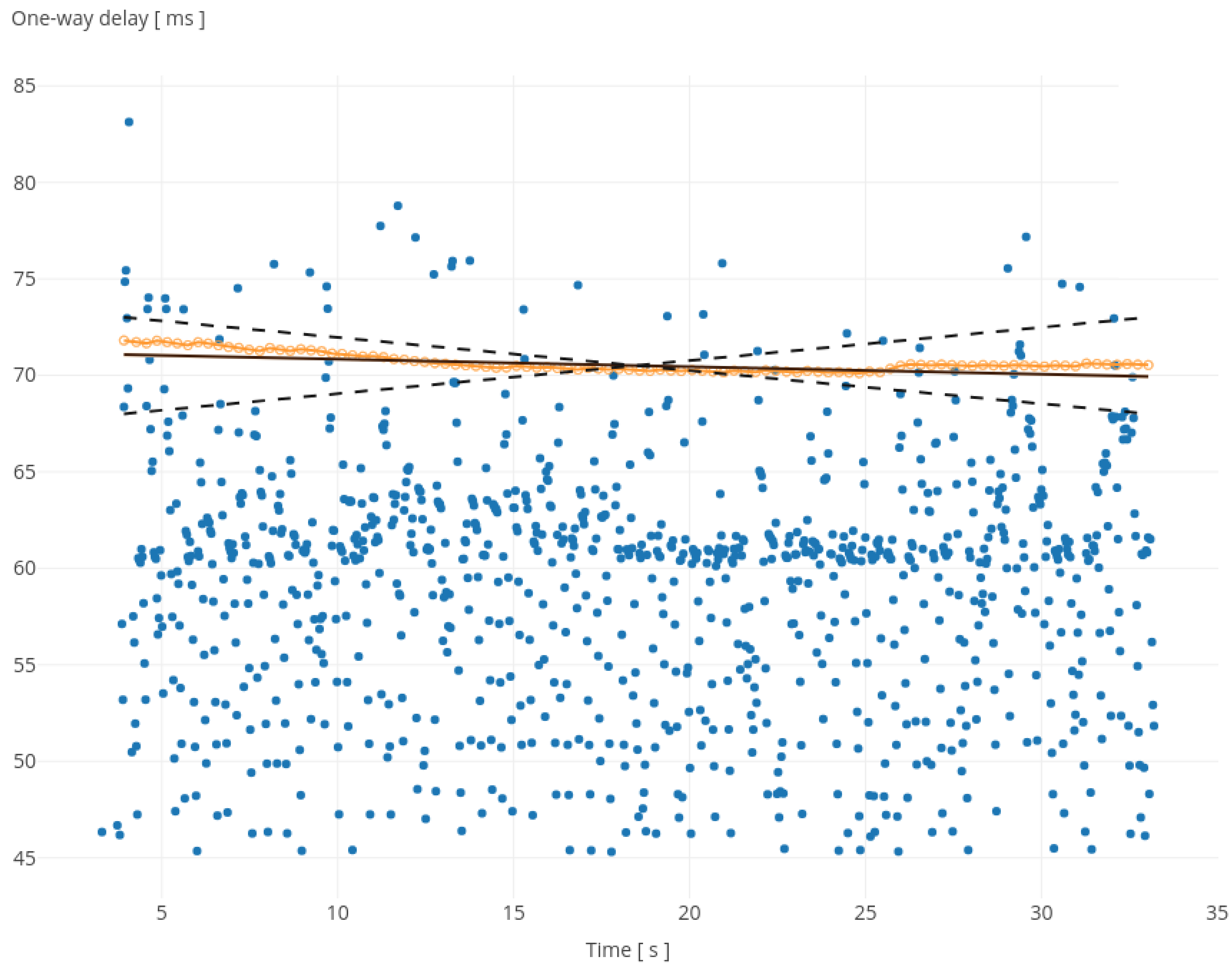
... zoom in



Compute the Theil-Sen slope and its 95%-CI



Compute the Theil-Sen slope and its 95%-CI which must fit in the tolerance interval +/– 5%



Take median of metric samples as run's metric

The application sets the requirements

Primary goals

Predictability

Adaptability

System must support

End-to-end real-time guarantees

- Change in traffic demand
- Mobility