How Does Task Scheduling Affect Engine Control Performance?

Paolo Pazzaglia, **Alessandro Biondi**, Marco Di Natale and Giorgio Buttazzo

Scuola Superiore Sant'Anna, Pisa, Italy

Sant'Anna
Scuola Universitaria Superiore Pisa

Retis
Real-Time Systems Laboratory

Real-time scheduling theory has provided a foundation for understanding and solving res
problems in systems that have real-time constraints. New fundamental results are needed t
trends in real-time systems design. RTSOPS 2016 encompasses all aspects that are rele
scheduling.

RTSOPS 2016 invites extended abstracts of open problems in areas such as, but not limited to:

- Single-, Multi- and Many-core scheduling;
- New models for real-time systems;
- Scheduling in cyber-physical systems;
- Mixed-criticality scheduling;
- Interactions between WCET (worst-case execution time) analysis and scheduling.

Instructions for authors

Extended abstracts (for either new open problems or status reports on previously-presented p
be written in English, and not exceed two A4 pages in length in single column, 10pt format, inc

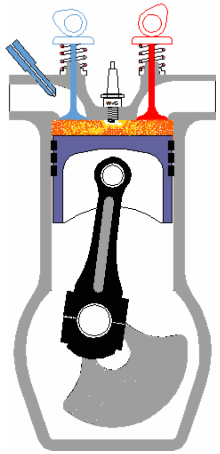
- Download the MS Word template here
- Download the LaTeX template here

INTRODUCTION

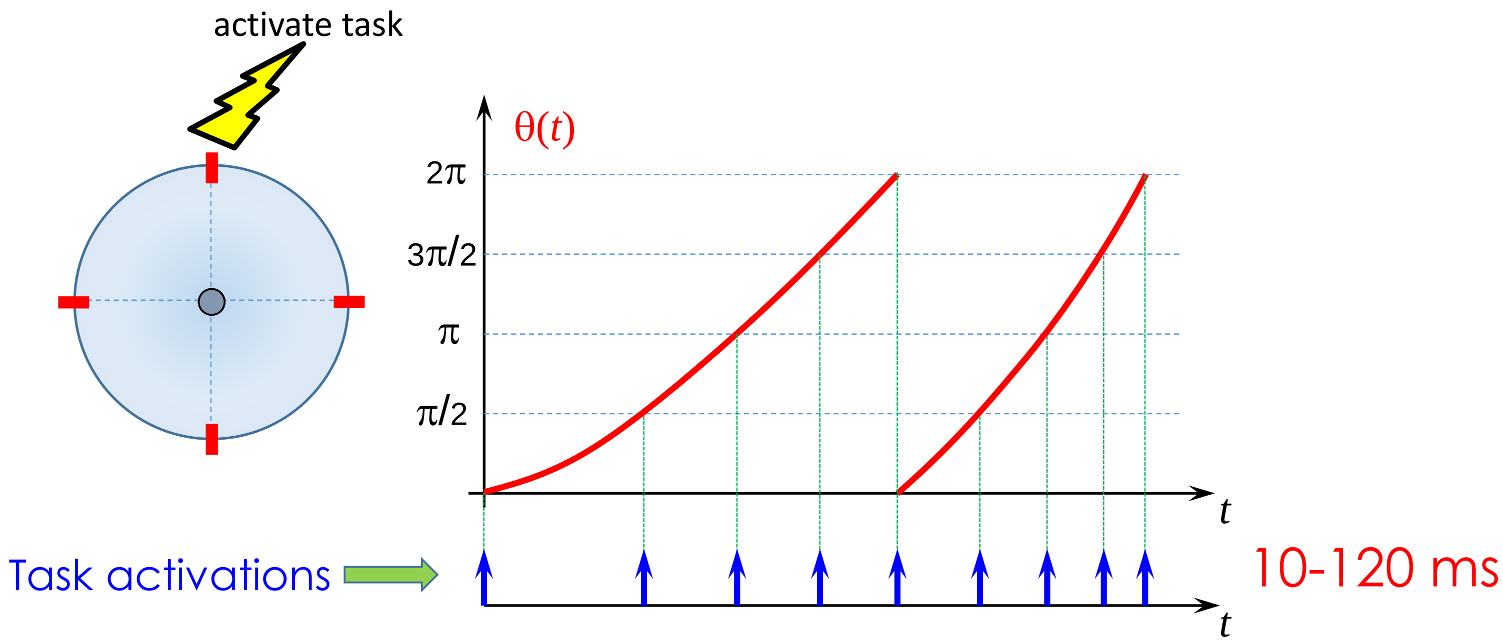
- **Engine control** is a very interesting and challenging **CPS** problem

- Scheduling plays a **key** role
- **Design constraints** (limited computational power)
- Timing significantly influences system **performance**
- Both time- and event-driven behavior

ENGINE CONTROL APPLICATIONS

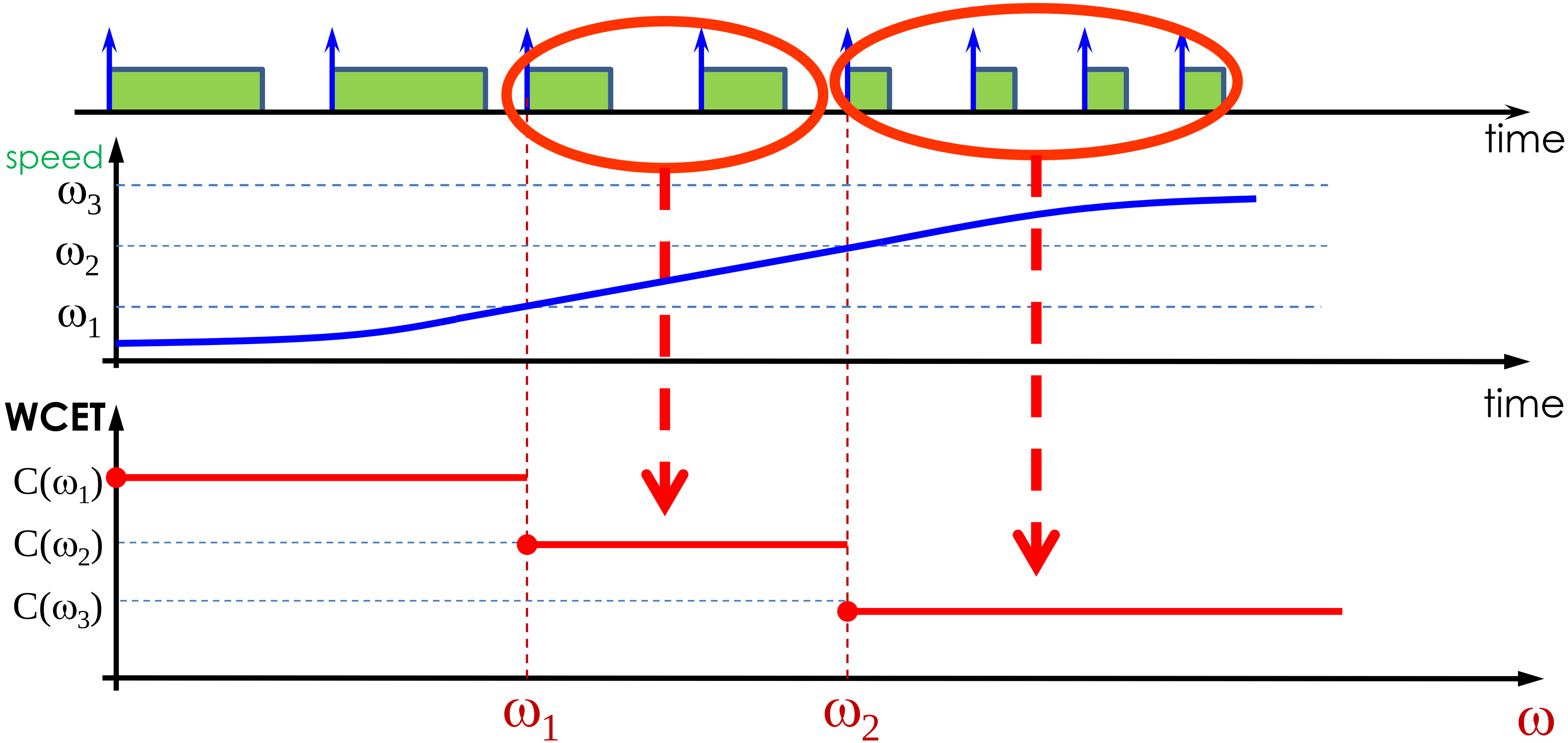


- **Engine control** applications include
  - **Periodic Tasks**, with fixed periods (1-500 ms)
  - **Angular Tasks**, linked to the rotation of the crankshaft

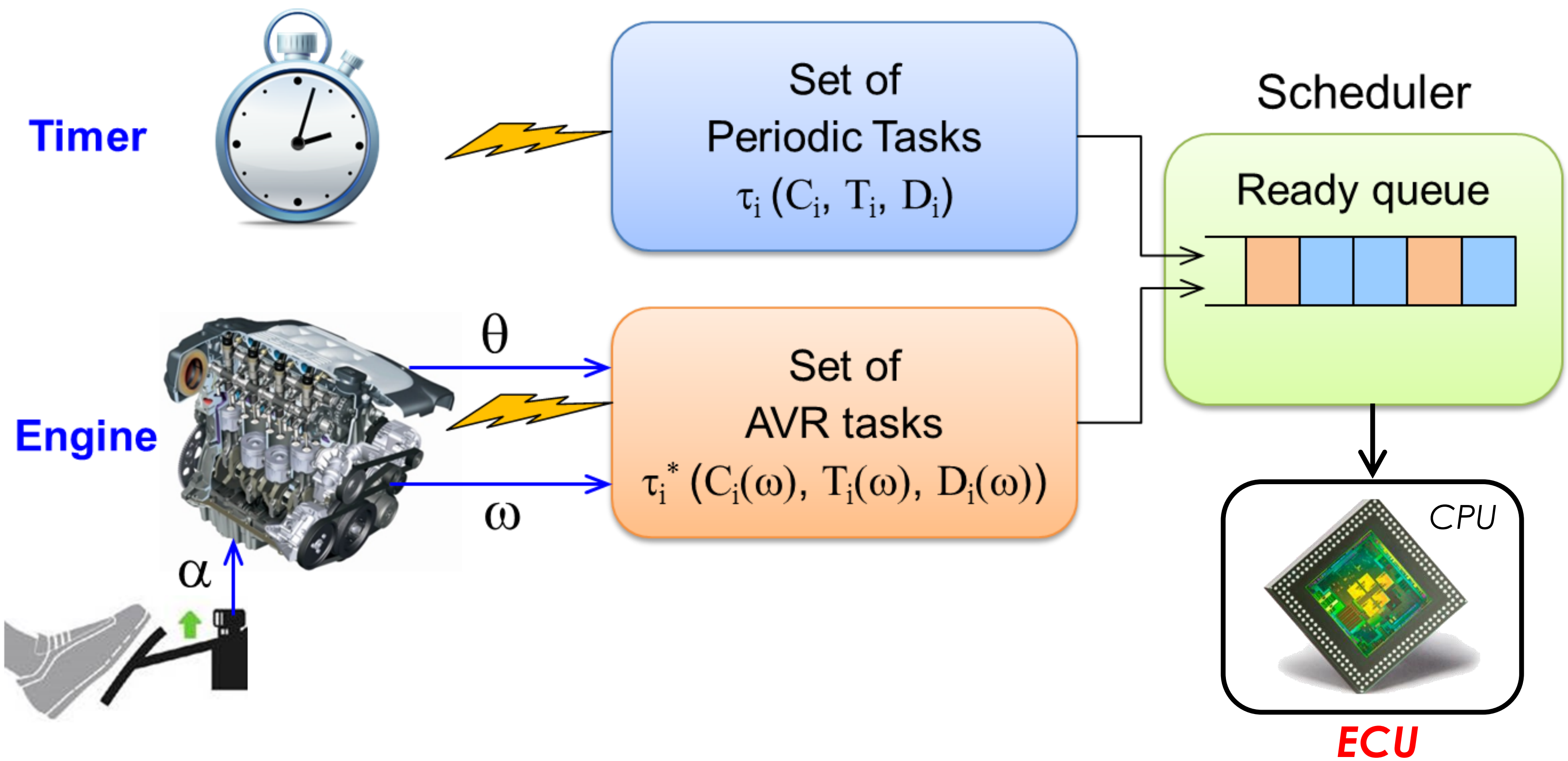


ADAPTIVE BEHAVIOUR

To prevent **overload** at high rates, *different control implementations* are used



ENGINE CONTROL APPLICATIONS



SCHEDULING PROBLEM

Periodic computational activities

Periodic Real-Time Tasks – Studied since *70's*

Engine-triggered computational activities

AVR Tasks – Studied only in the last years

Buttazzo et al. DATE14 *Davis et al.* RTAS14 *Biondi et al.* ICCPS15 *Guo and Baruah* ICCPS15

ALL THE SOLUTIONS FOR THE SCHEDULING PROBLEM ASSUMED HARD DEADLINES

Are engine control applications **hard real-time**?

THE (REAL) PROBLEM

- Engine control is **not** **hard real-time**

> **Deadline misses** can be **tolerated**

- <u>Informal **specifications**</u>
  - *"Deadline can be missed but not that many"*
  - *"Not that many consecutive deadline misses"*
  - *"Not that large maximum response-times"*
  - *"What matters is the engine performance"*
  - *"The system incurs in transient overloads"*

THE (REAL) PROBLEM

- The **objective** of the scheduling is **not** necessarily to *meet* al the **deadlines**.

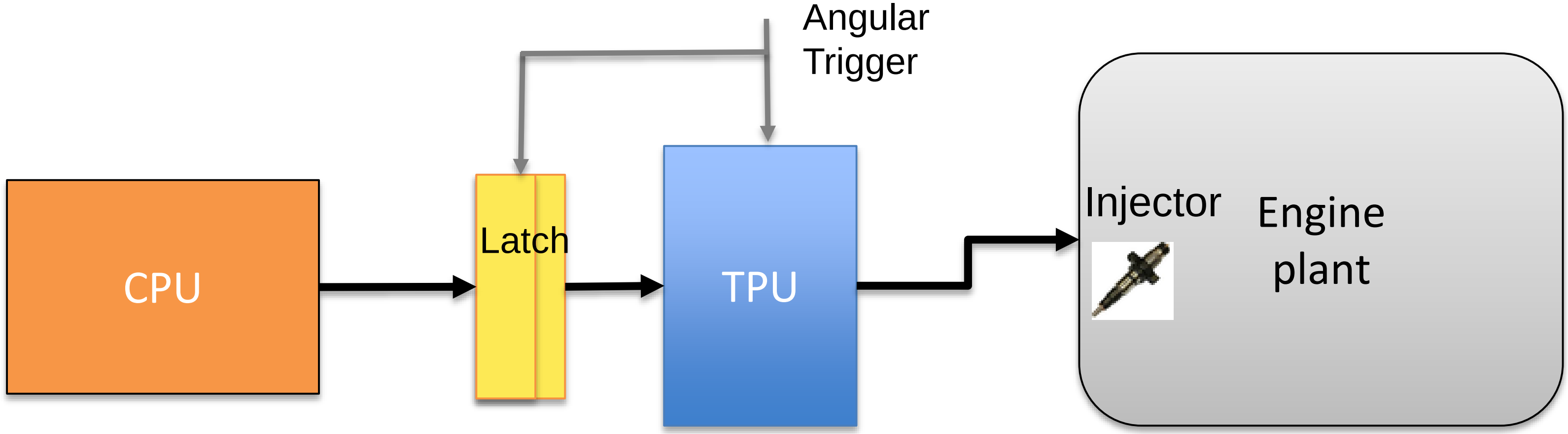
BUT

Maximize the **engine performance** given a set of computational constraints

Engine control is a complex **multi-criteria design optimization** problem
(*power, fuel efficiency, noise, emissions,…*)

EXAMPLE OF CHALLENGES

FUEL INJECTION



- **TPU** uses data produced from the **CPU** (injection angle, quantity of fuel, CR pressure…)

If **deadlines** are **missed** (on the **CPU**), the **TPU** uses **old data** for the *next injection*

EXAMPLE OF CHALLENGES

FUEL INJECTION

- **Deadline misses** can be **penalizing** if the conditions of the engine **changed** (too much) from previous cycles.
- The use of **old data** can produce **errors** in the **injection angle**.

Scheduling **errors**

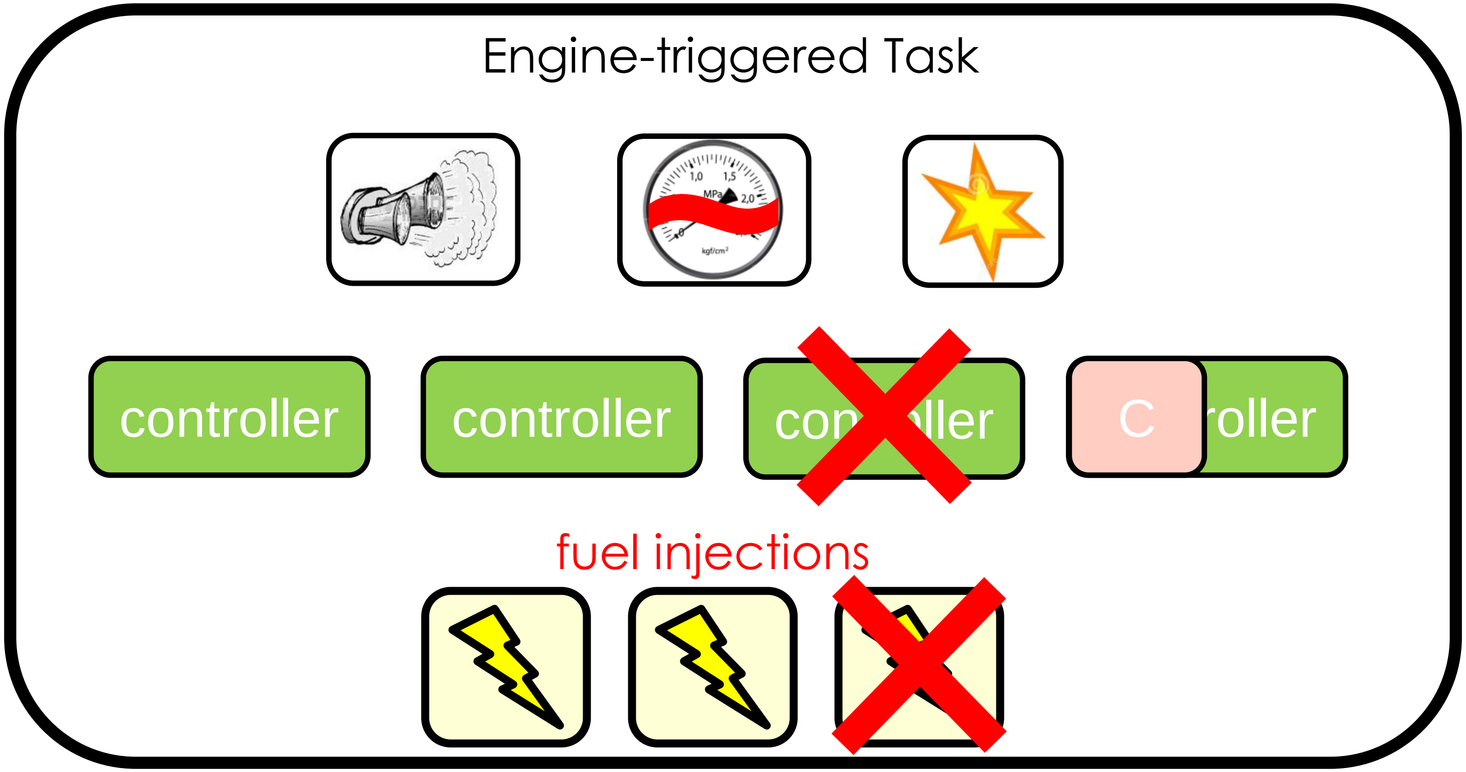
Inaccurate injection

Decreasing performance

EXAMPLE OF CHALLENGES

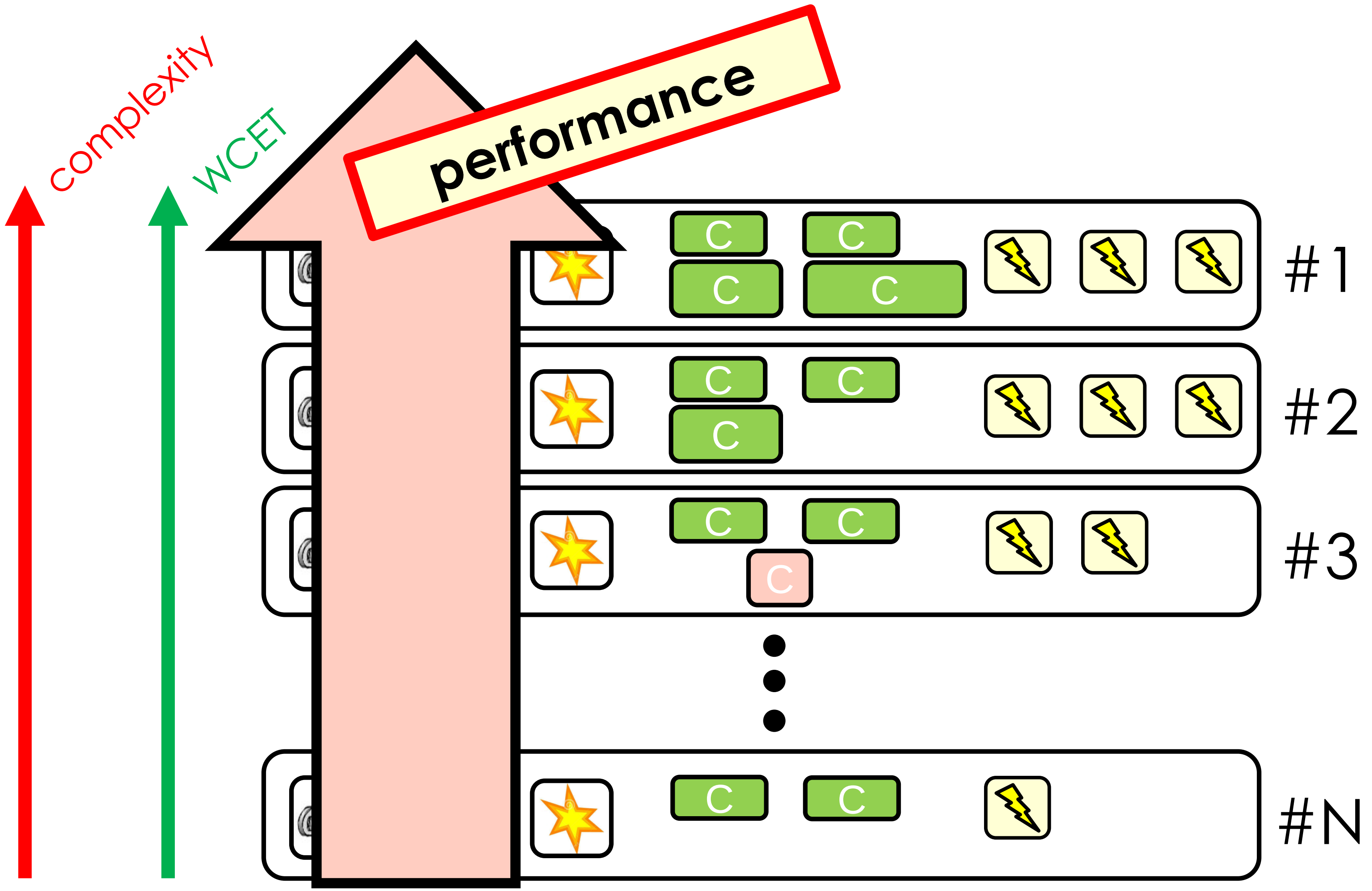
SWITCHING SPEEDS

- To prevent **overload** conditions, **different control implementations** are used depending on the **engine speed**



EXAMPLE OF CHALLENGES

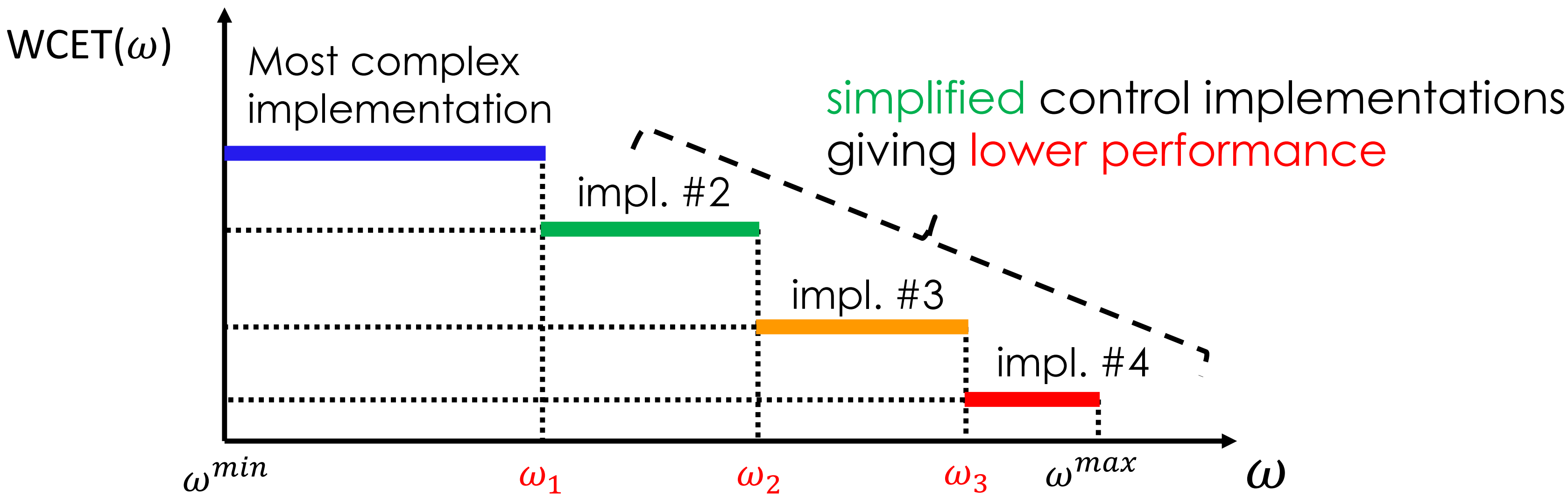
SWITCHING SPEEDS



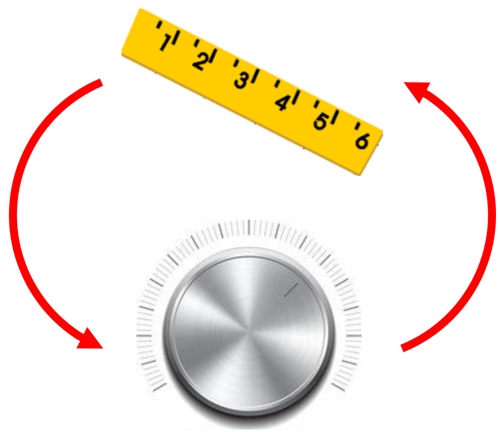
EXAMPLE OF CHALLENGES

SWITCHING SPEEDS

- Which is the **best** speed to **switch** control implementation?
- The problem has been *recently* attempted only under the assumption of hard deadlines…



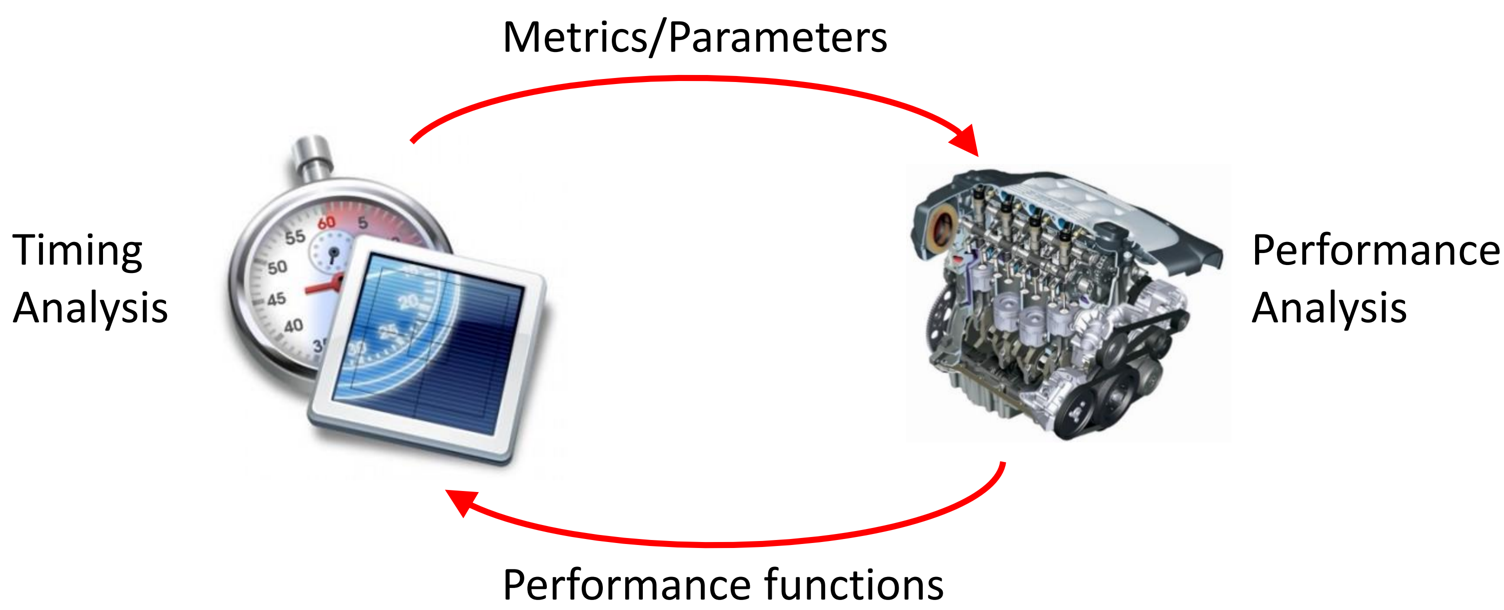
TODAY'S APPROACH: ITERATIONS BETWEEN TEST-BENCH AND TUNING



"Something" more systematic supported by a **model** and an **analysis** would be very useful…

CAN THE PROBLEM BE PARTITIONED?

- Is it possible to **separate** the **timing** (*scheduling*) problem from the **functional** (*performance*) analysis?



EXISTING APPROACHES

Firm real-time (e.g., m-k model)

- Still yes/no analysis;
- No way to express impact on performance.

Generalized response-time analysis

- Allows computing max. number of consecutive deadline misses;
- System state not considered;
- No way to express impact on performance.

Value-based scheduling

- Allows expressing performance as value functions;
- How to obtain (and define) value functions?
- Value should be dependent on the system state.

LIMITATIONS

- **None** of the existing **approaches** can be used **as it is**.
- Possibly a **combination** of such techniques will be required.

Large **lack** of **models** (and corresponding **analysis** techniques) to take into account the **system state** (and hence **performance**)

SCHEDULING AS DESIGN OPTIMIZATION

Scheduling in engine control *should* be a **design optimization** of **performance functions**

PROBLEMS

- Likely, **performance** cannot be expressed as a simple function of timing parameters.
- **Performance** is **not** independent from **past behavior**.
- **Multiple** performance indexes must be considered.

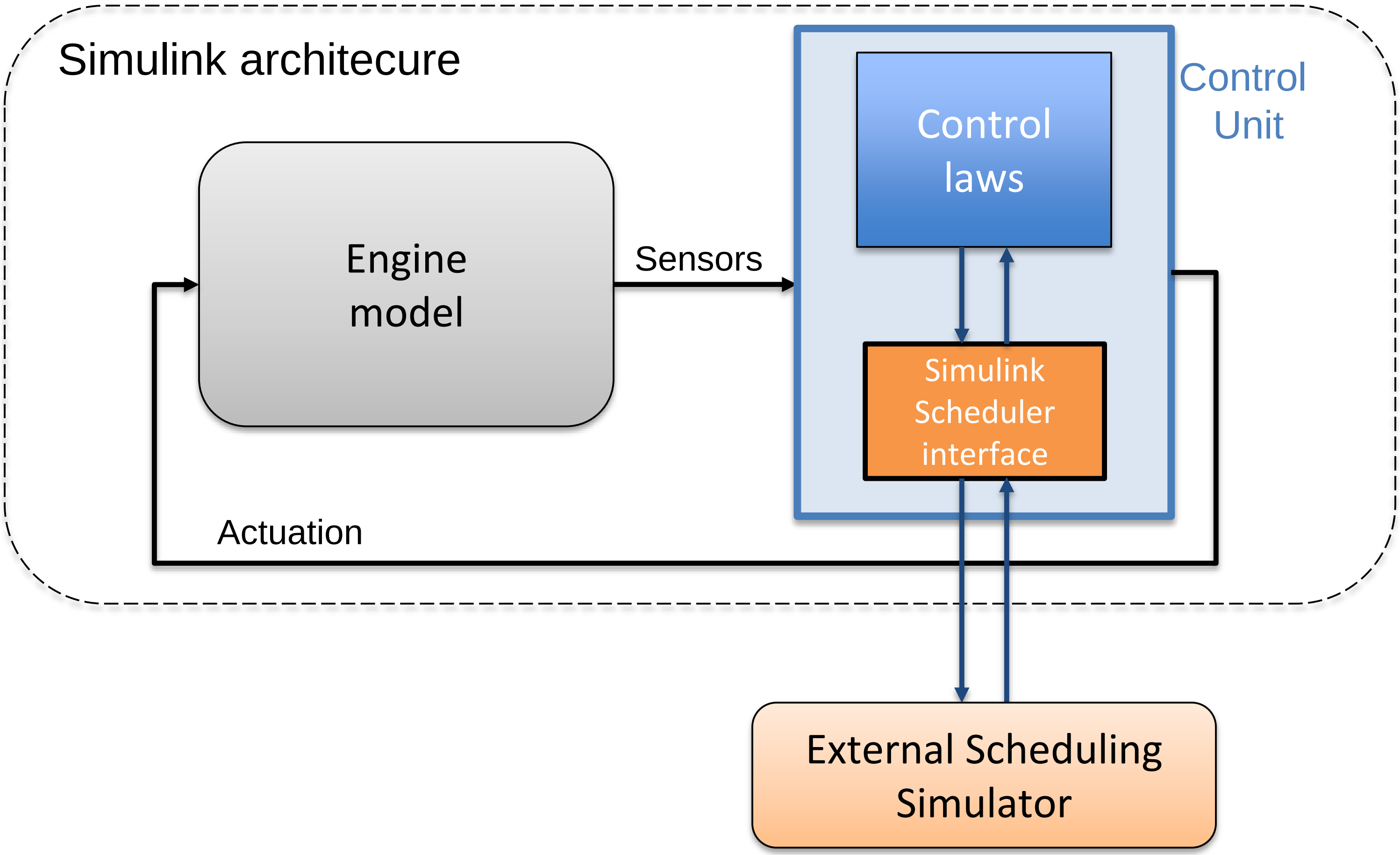
OUR (CURRENT) APPROACH

- Closed form **functional mapping** between **temporal** parameters and **performance** is possible for **simple** control systems.
- This approach becomes soon **prohibitively difficult** for a **realistic** CPS due to the intrinsic complexity of the system.

Our attempt

Use a **simulation framework**

OUR (CURRENT) APPROACH



Thank you!

Alessandro Biondi
alessandro.biondi@sssup.it