A Co-Simulation Framework for Engine Control Applications

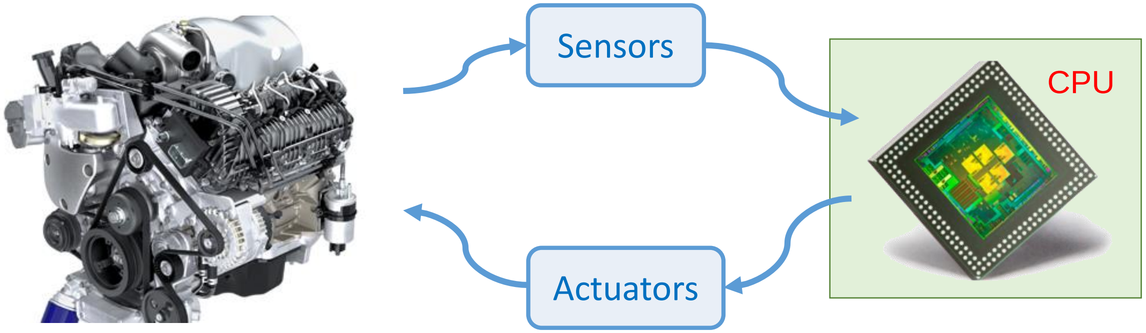
Paolo Pazzaglia, **Marco Di Natale, Giorgio Buttazzo**

Scuola Superiore Sant’Anna, Pisa

paolo.pazzaglia@santannapisa.it

Roma, September 8th, 2017

The Diesel engine control problem



- Challenging CPS problem
  - **Complex** physical components
  - High number of electronic control components
  - Periodic, aperiodic and **angular triggered** tasks

- Does **not need hard real time constraints** (resilient to deadline misses)
- …However perfomance sensitive to **jitter** and **delays**!

Objectives and framework

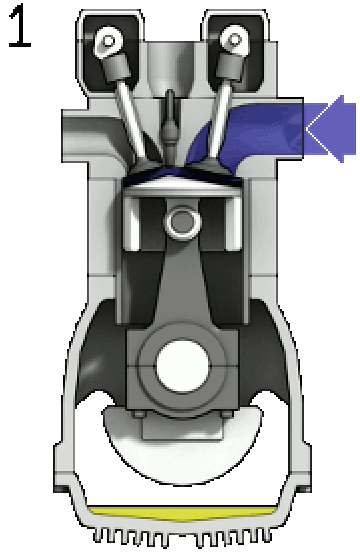
- Study the effects of scheduling policies and task design on **performance** of control applications
  - Evaluation with **simulation tools**
- Verify assumptions on the performance functions with respect to timing

Proposed solution:

- **Co-simulation framework** developed on Simulink with a **scheduling simulator** integrating:
  - Model of the engine
  - Model of the tasks and scheduler
  - Model of the functional controls

Injection problem in engine control

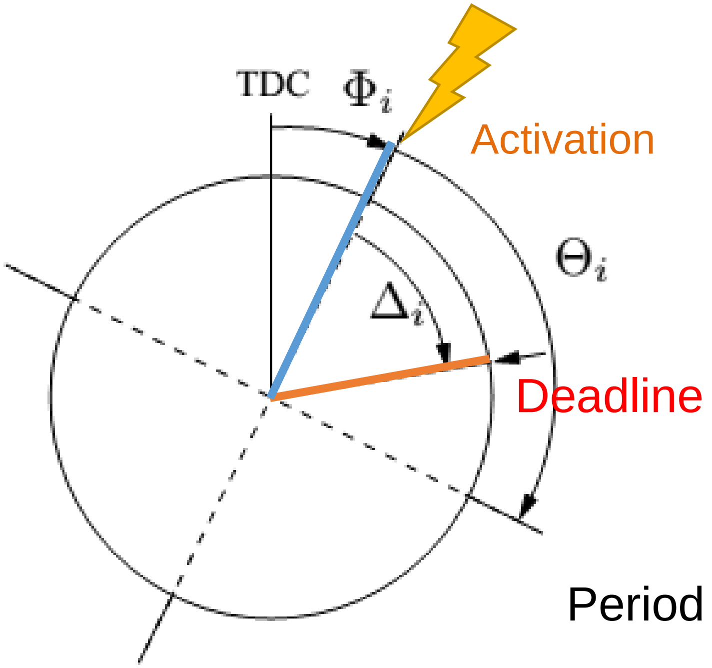
- **Fuel injection** is an example of task with temporal constraints
- It is the main component of control



Animation by Zephyris - Own work, CC BY-SA 3.0,
https://commons.wikimedia.org/w/index.php?curid=10896588

- Fuel quantity and timing vary with engine conditions
- Fuel injection must be **precise** to assure optimal combustion process
- Injection errors could compromise engine functionality

Angular task



- Task managing the fuel injection is an **angular task**:
- Angular tasks are activated at a specific crankshaft angle
- The **angular deadline** and **period** are fixed, but timing depends on engine speed

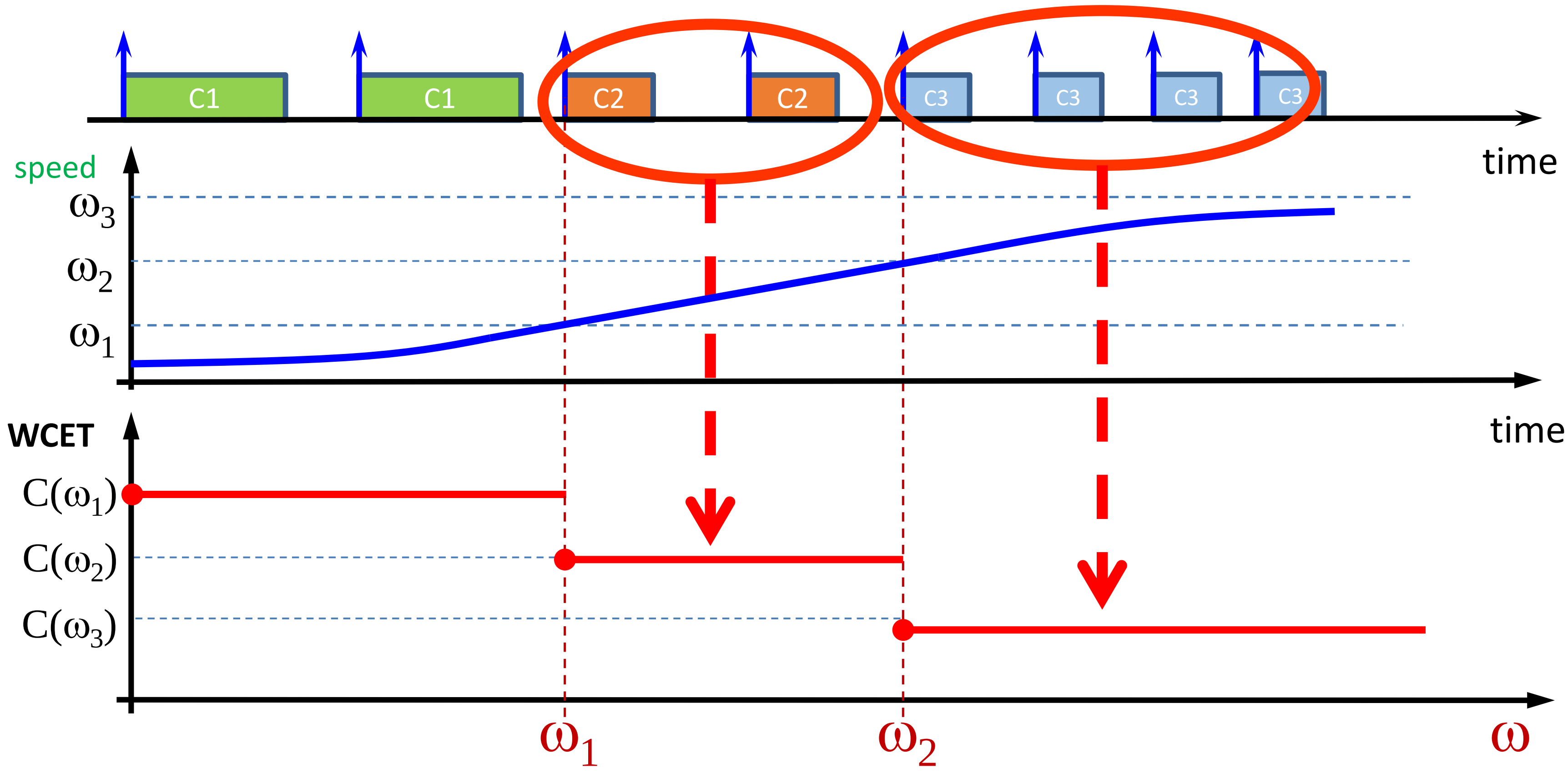
Activation rate depends on crankshaft speed:

↓

Potential **overload** at high speeds!

Adaptive Variable Rate

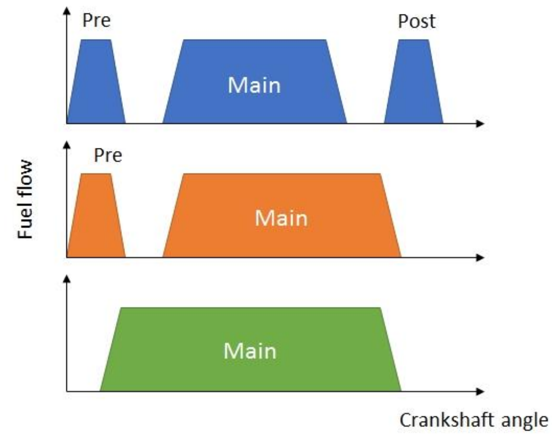
- Solution: Multiple control modes with WCET decreasing at high speeds: **Adaptive Variable Rate** (AVR) task
- Mode changes happen at particular **switching speeds**

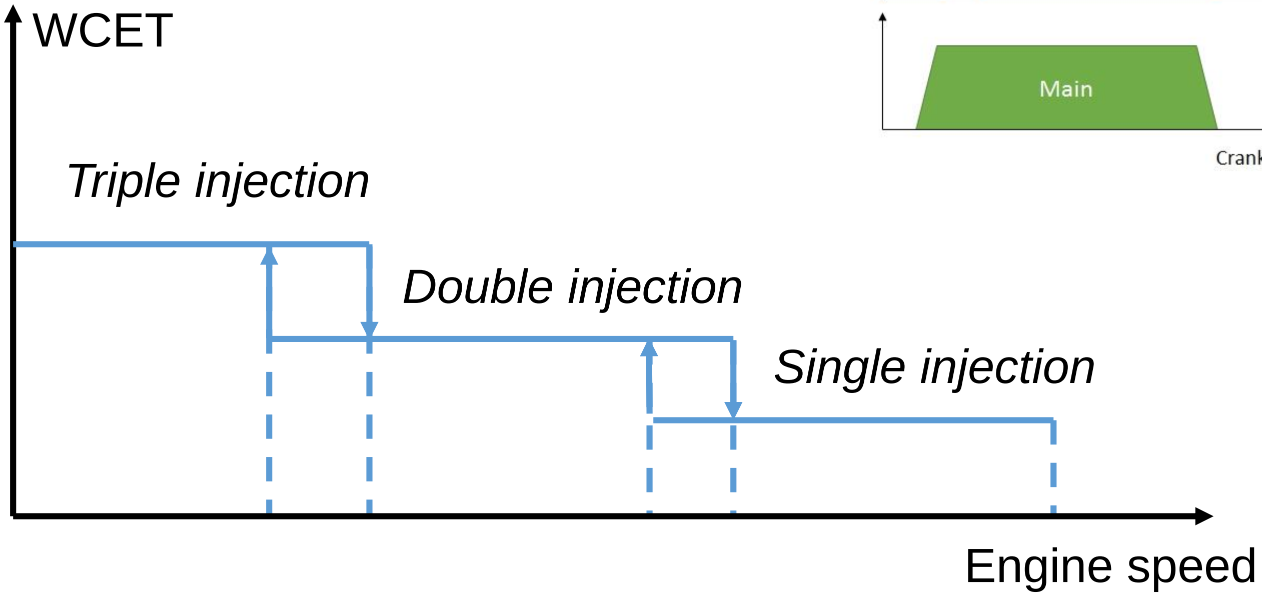


Adaptive Variable Rate

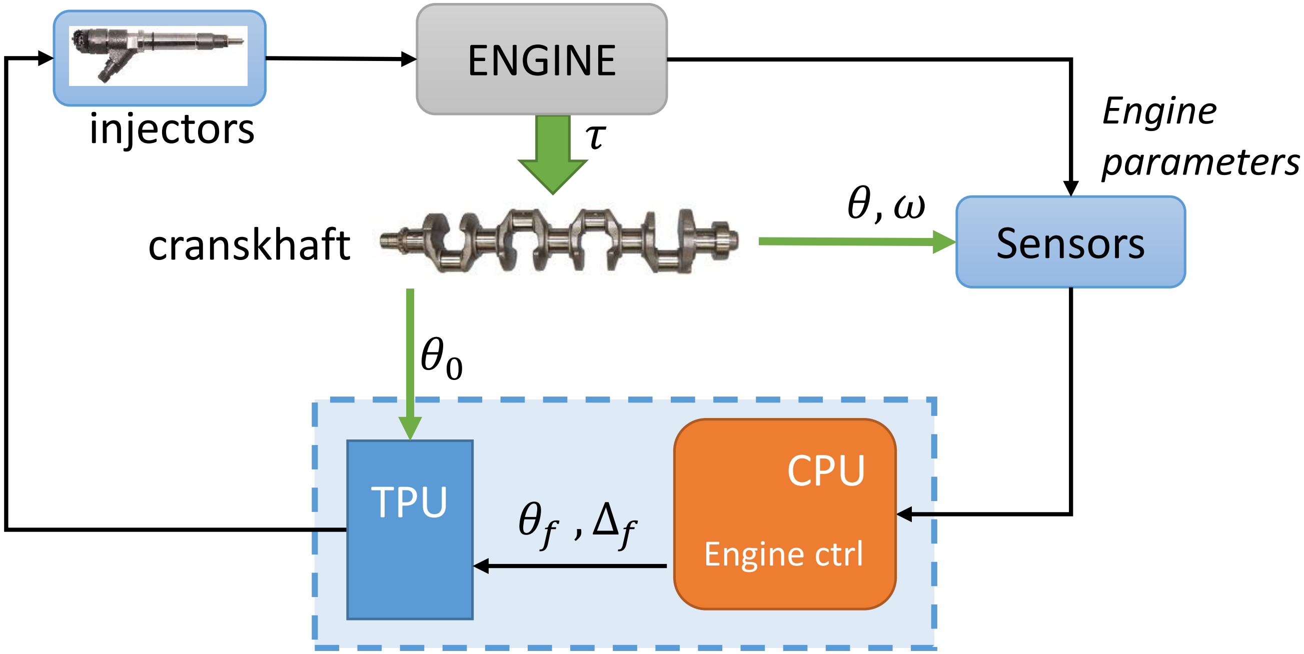
- For the purposes of this work we model mode changes only varying the number of injections

- **Multiple injections** help controlling combustion parameters





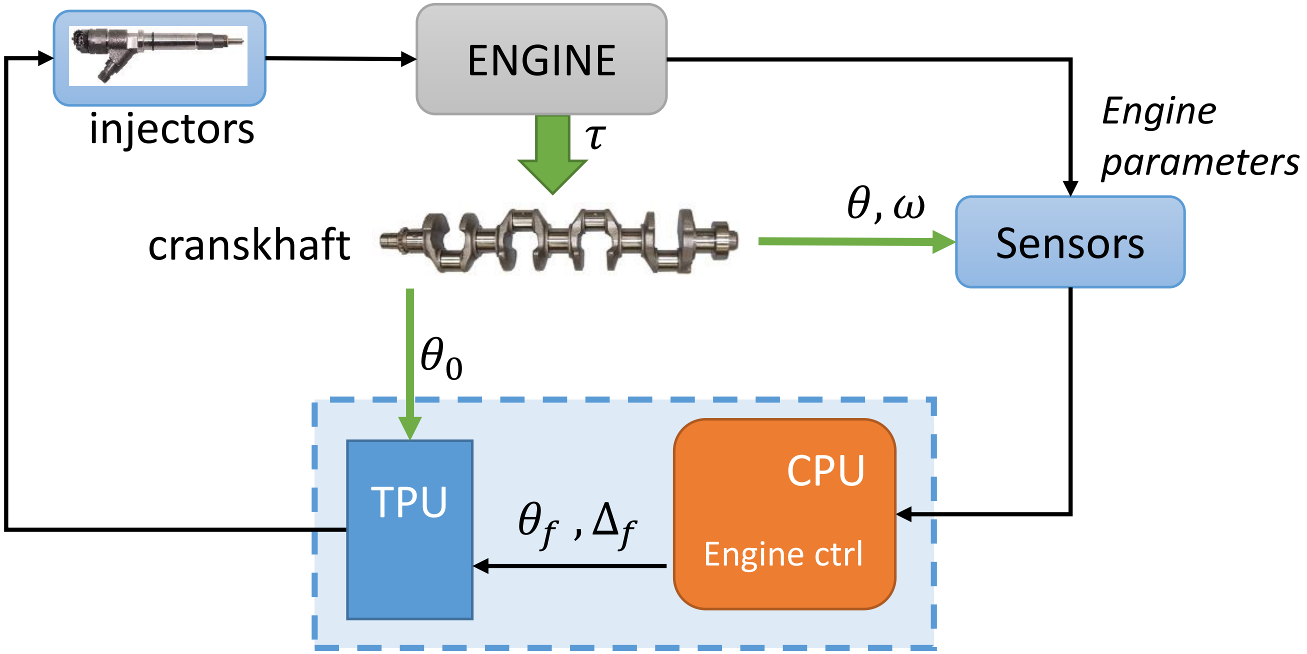
TPU and deadline misses



- The **Time Processing Unit** (TPU) is a co-microcontroller that handles the injection actuation in synchronous modality

- Missing a deadline on the control task means that the actuation is done with data of previous cycle

TPU and deadline misses



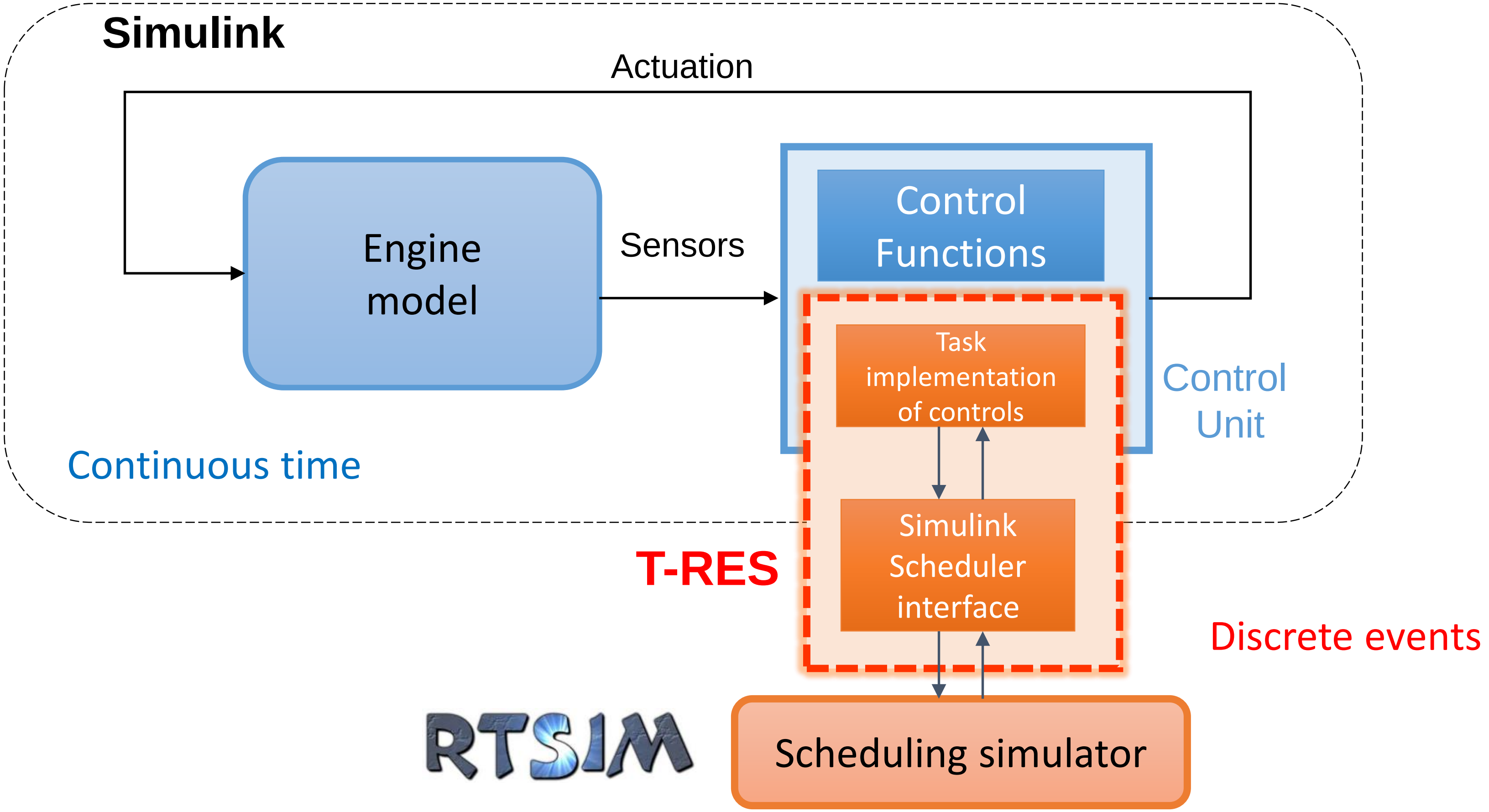
Deadline misses can be penalizing if the conditions of the system changed (too much) from the previous iteration!

Scheduling as design optimization

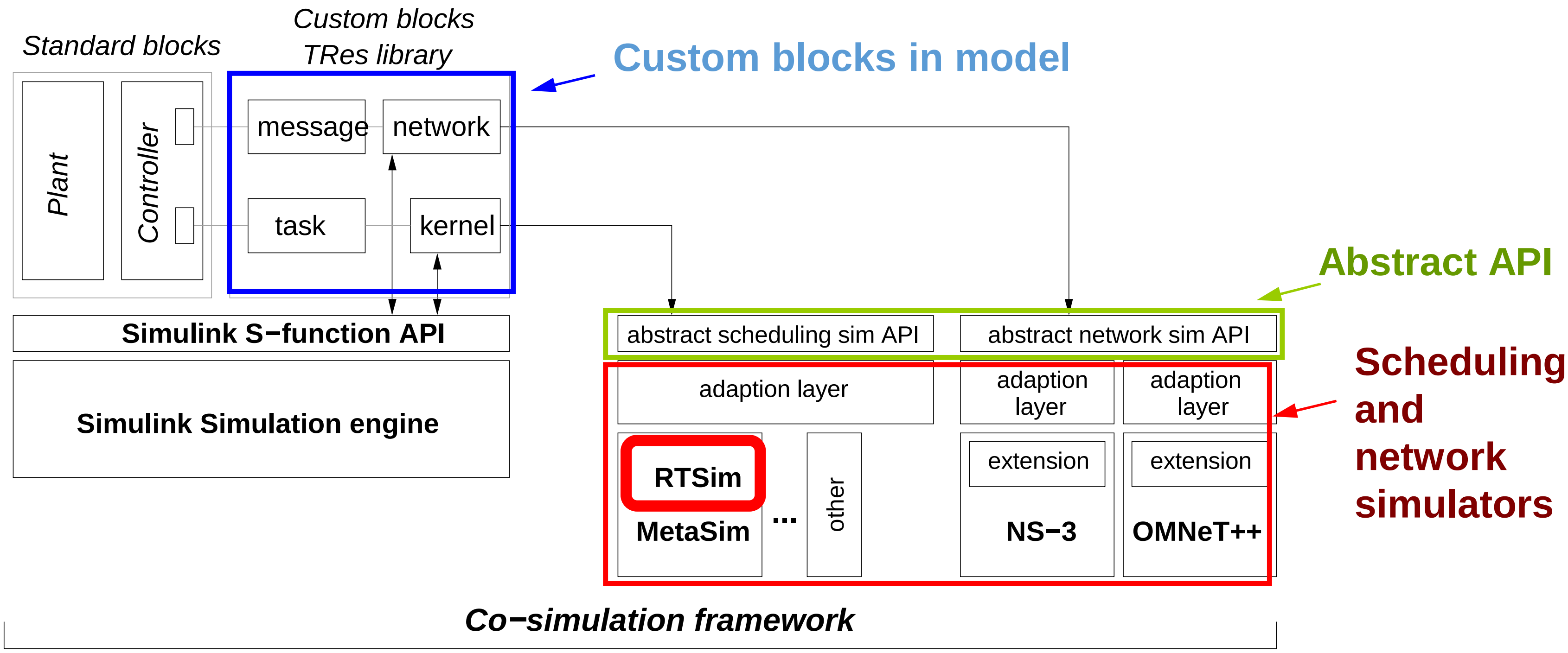
- Performance is strictly related to timing, but its sensitivity varies with status and its dynamics
- Performance functions not independent from past!
- Also, **multiple performance indexes** must be addressed (power, efficiency, emissions, noise, fuel consumption…)
- Scheduling in engine control problem should be a **design optimization** using performance functions

The co-simulation framework

A **co-simulation framework** to test different scheduling and control strategies and their impact on performance

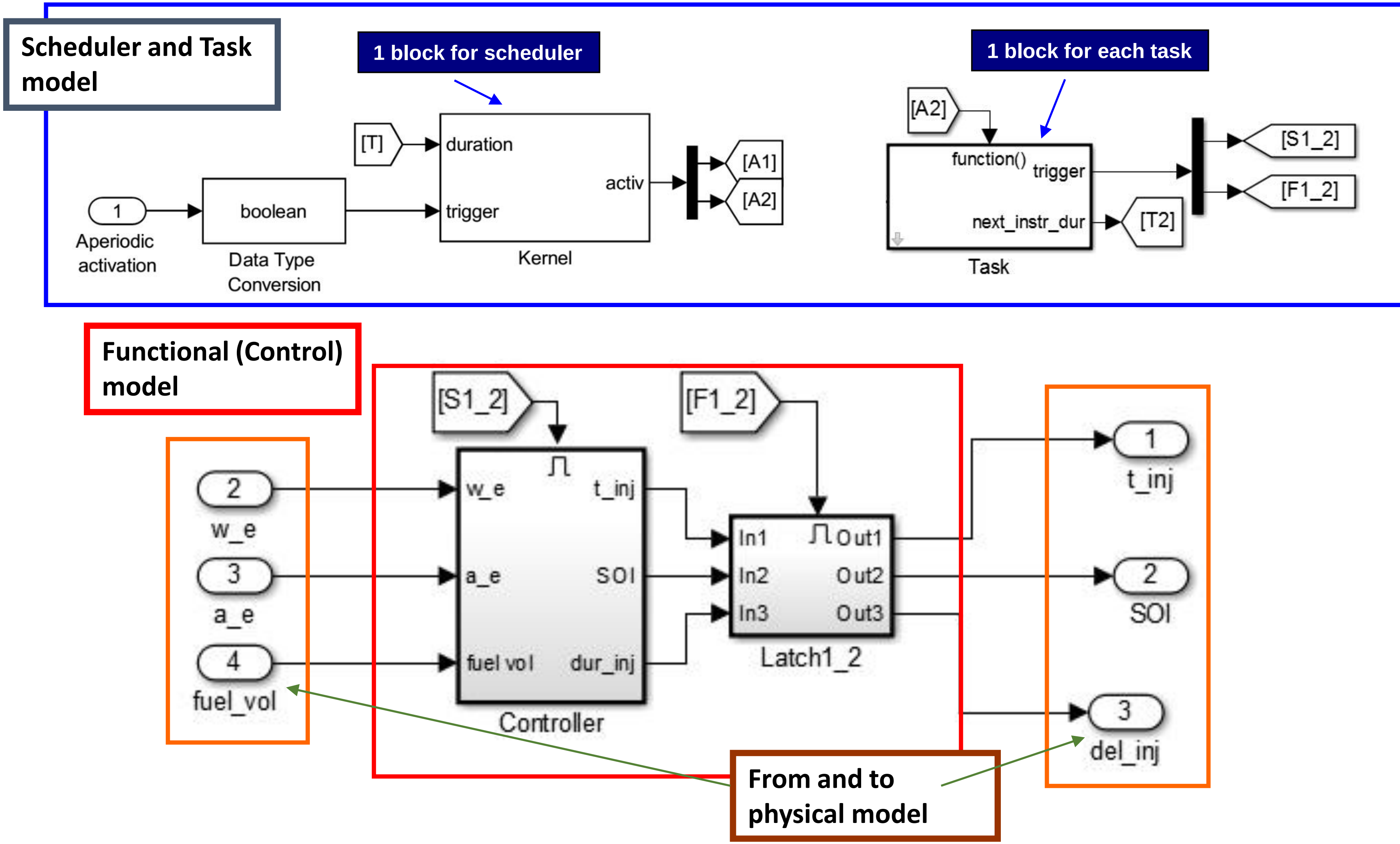


T-Res: a co-simulation framework

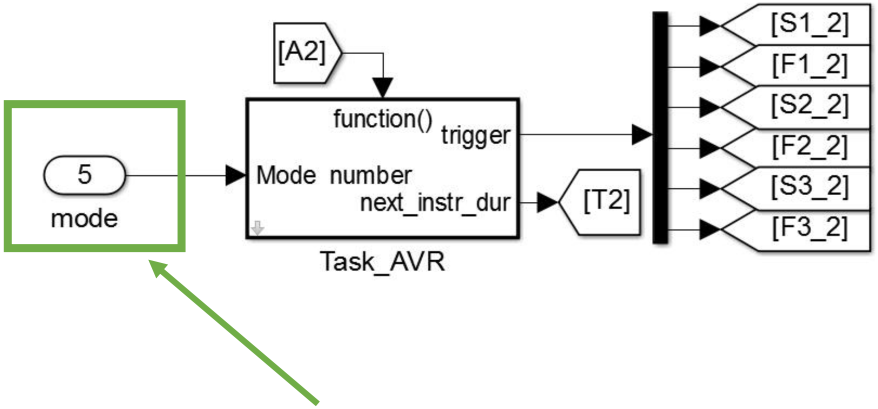


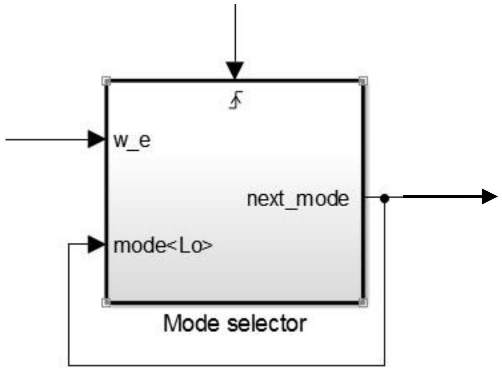
- **T-Res** manages activation, termination and preemption of tasks
- <u>Inserts scheduling delays in the simulation</u>

T-Res: Adding scheduling to Simulink



T-Res: Custom block for AVR tasks



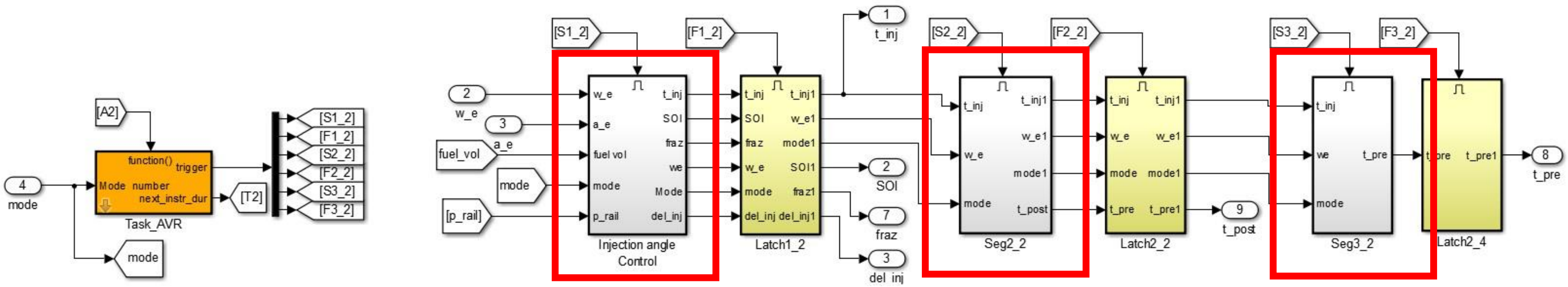


Mode selector

- Adding the "**active mode**" input
- Every mode is constructed as a sequence of instructions (segments), with different WCET
- The **deadline** of the AVR task is dynamically updated as the speed of the engine changes, and provided to RTSim

T-Res: AVR task implementation

- An example of the implementation of an AVR task in T-Res



Current setting:

<u>Three fuel control modes:</u>

1. Triple injection [0-1500]
2. Double injection [1200-3000]
3. Single injection [2800-v_max]

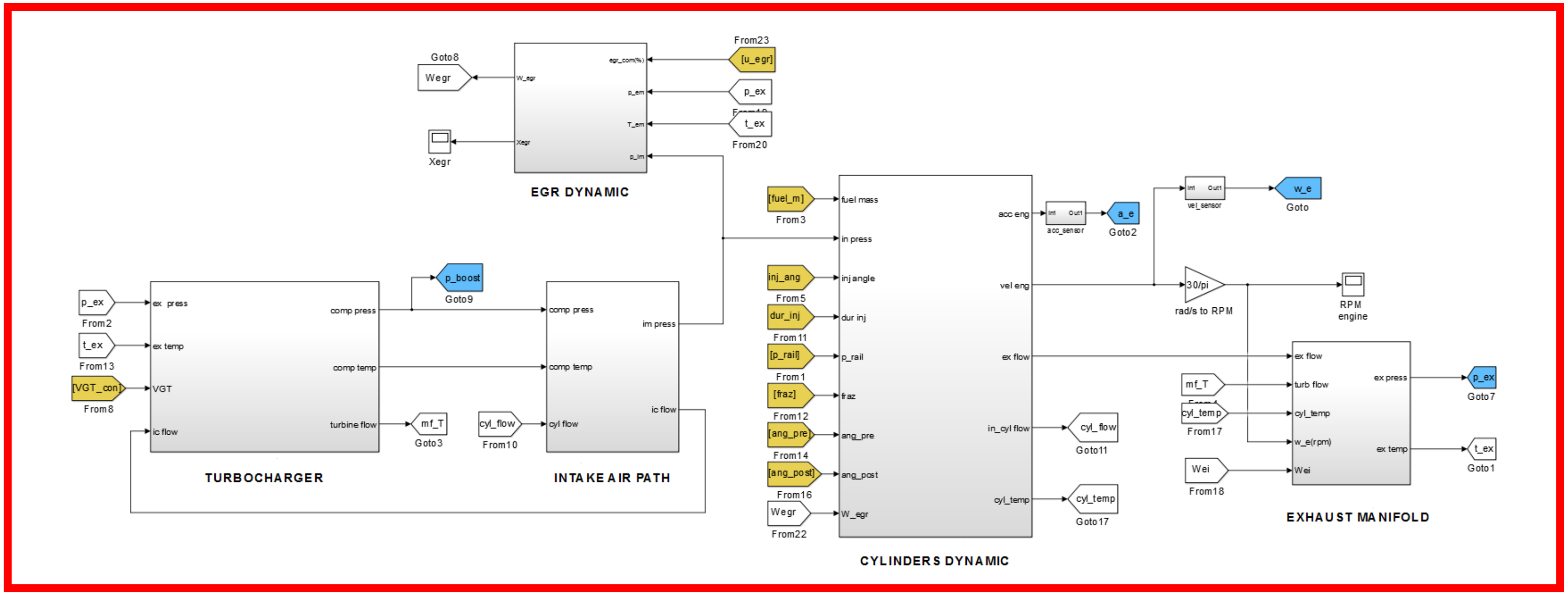
```
tAVR_instrs = {'fixed(.005)','fixed(.005)','fixed(.005)';...
               'fixed(.005)','fixed(.005)','fixed(0)';...
               'fixed(.005)','fixed(0)','fixed(0)'};

num_modes = 3;

v_plus = [1500, 3000, v_max];
v_minus = [0, 1200, 2800];
```

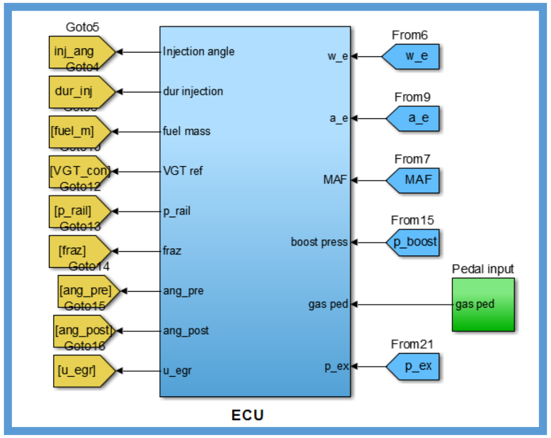
Matlab Simulink architecture

Simulink implementation: continuous+discrete simulation



Physical structure

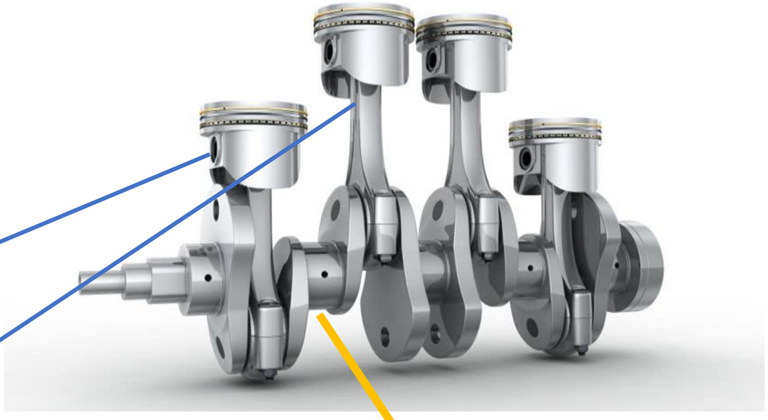
Control Unit reads data from sensors and computes actuation commands

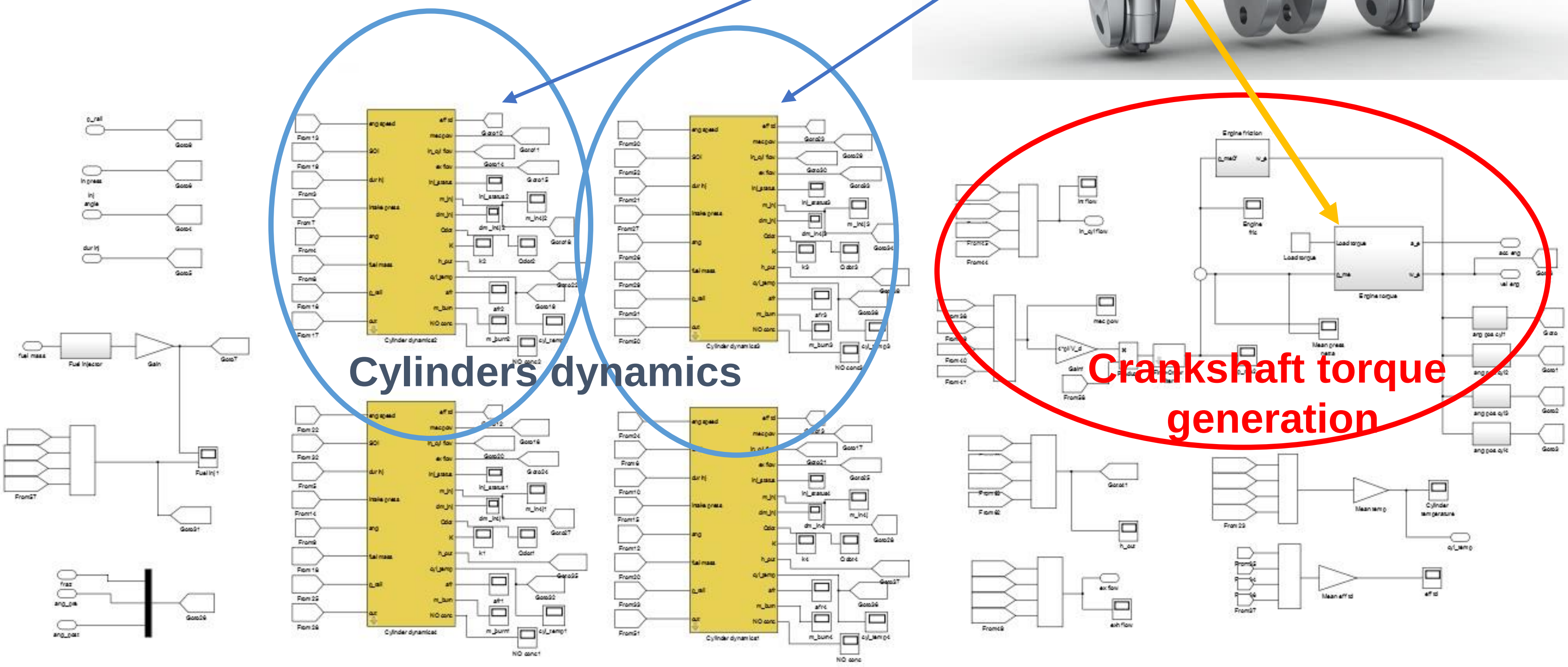


Control Unit

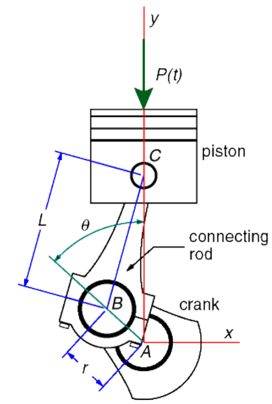
The engine model

- Engine dynamics
- Modeling multiple cylinders with a general **cylinder block**





In-cylinder dynamics



The cylinder block includes:

- **Mechanical model** of valves, crank-rod mechanism, torque generation and thermodynamic efficiency

$$\left(\frac{dm}{dt}\right)_x = \mu_{flow} A_{eff} \frac{p_0}{RT_0} \sqrt{\frac{2\gamma}{\gamma - 1}\left[\left(\frac{p_1}{p_0}\right)^{\frac{2}{\gamma}} - \left(\frac{p_1}{p_0}\right)^{\frac{\gamma+1}{\gamma}}\right]}$$

- **Injector** dynamics

$$\frac{dm_f}{d\alpha} = \frac{\mu_{inj}}{2\pi n_e} A_n \sqrt{2\rho_f (p_{rail} - p_{cyl})}$$

- **Heat Release model** of combustion

$$\dot{Q} = C_1 \left(m_f - \frac{Q}{LHV}\right) exp\left(C_2 \frac{\sqrt{k}}{\sqrt[3]{V_{cyl}}}\right)$$

- Semiempirical **emission models** of NOx and soot

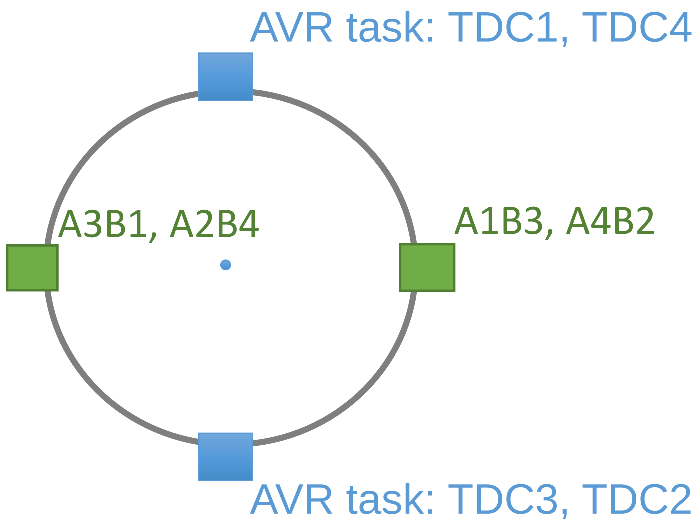
$$m_{NO_x} = \int_\alpha \dot{Q} C_1 \left(\frac{n_e}{2000}\right)^{C_2} \exp\left(\frac{C_3}{T_{ad}}\right) d\alpha$$

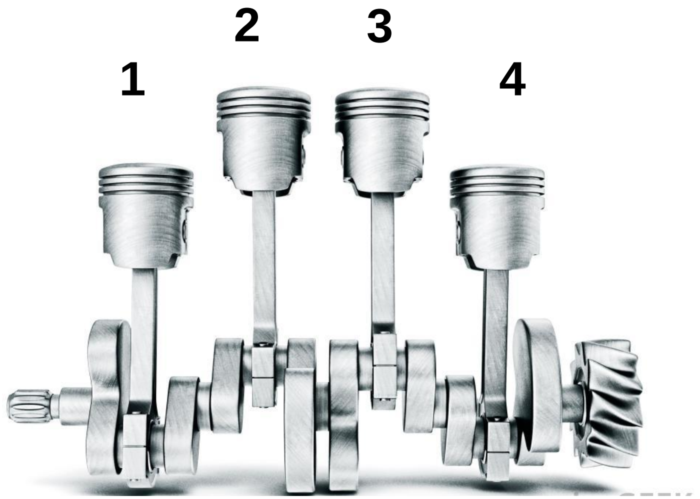
$$\frac{d[soot]}{dt} = \frac{d[soot_f]}{dt} - \frac{d[soot_o]}{dt}$$

Engine Control

- Injection angle control is formally split into two AVR tasks
- Tasks activation every half crankshaft rotation
- Phased by 90°

- Control done with static maps





Simulations

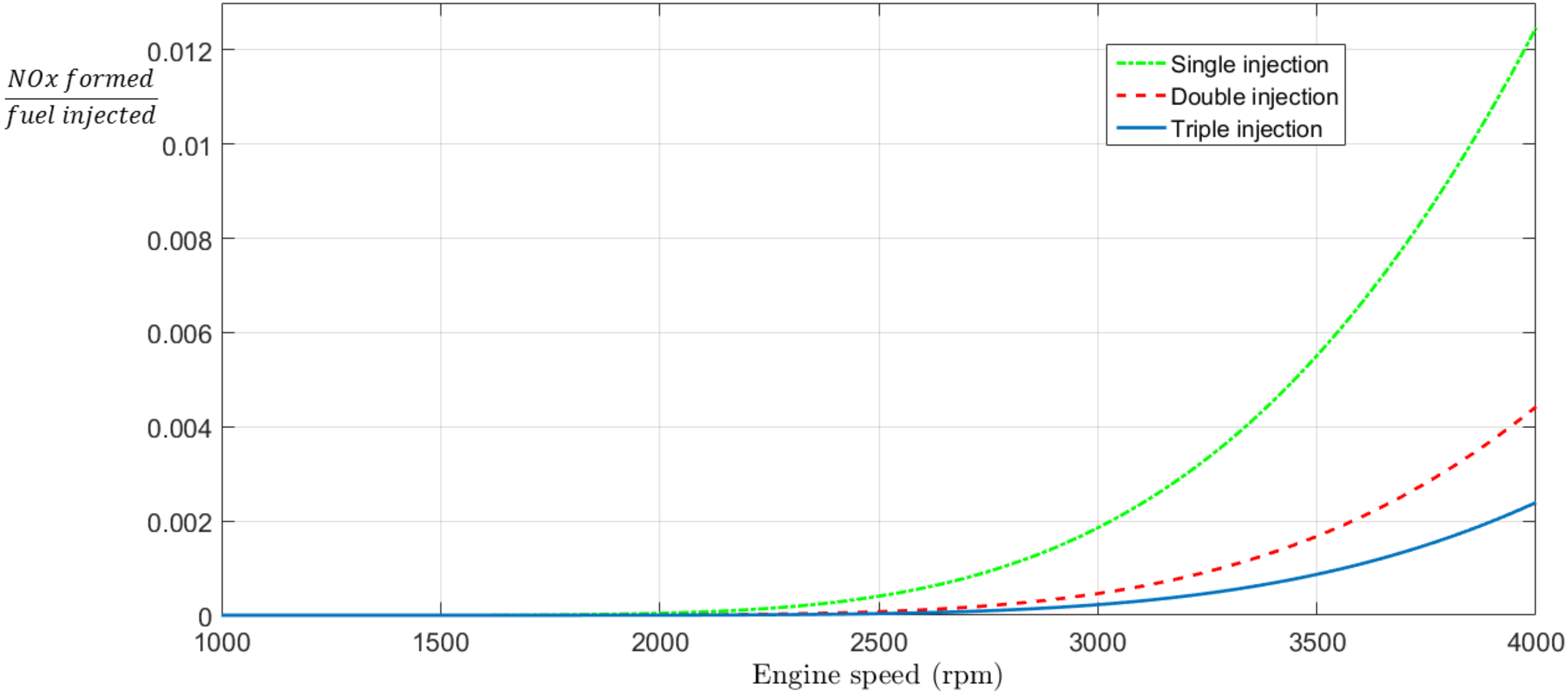
- Simulating specific patterns of input pedals:
  - Slow acceleration
  - Step acceleration

- Studying the performance index as a function of control modes and speed

- Showing how the scheduling delays result in errors in the angle/duration of the injection actuation and the corresponding loss in performance

Multiple injections and performance

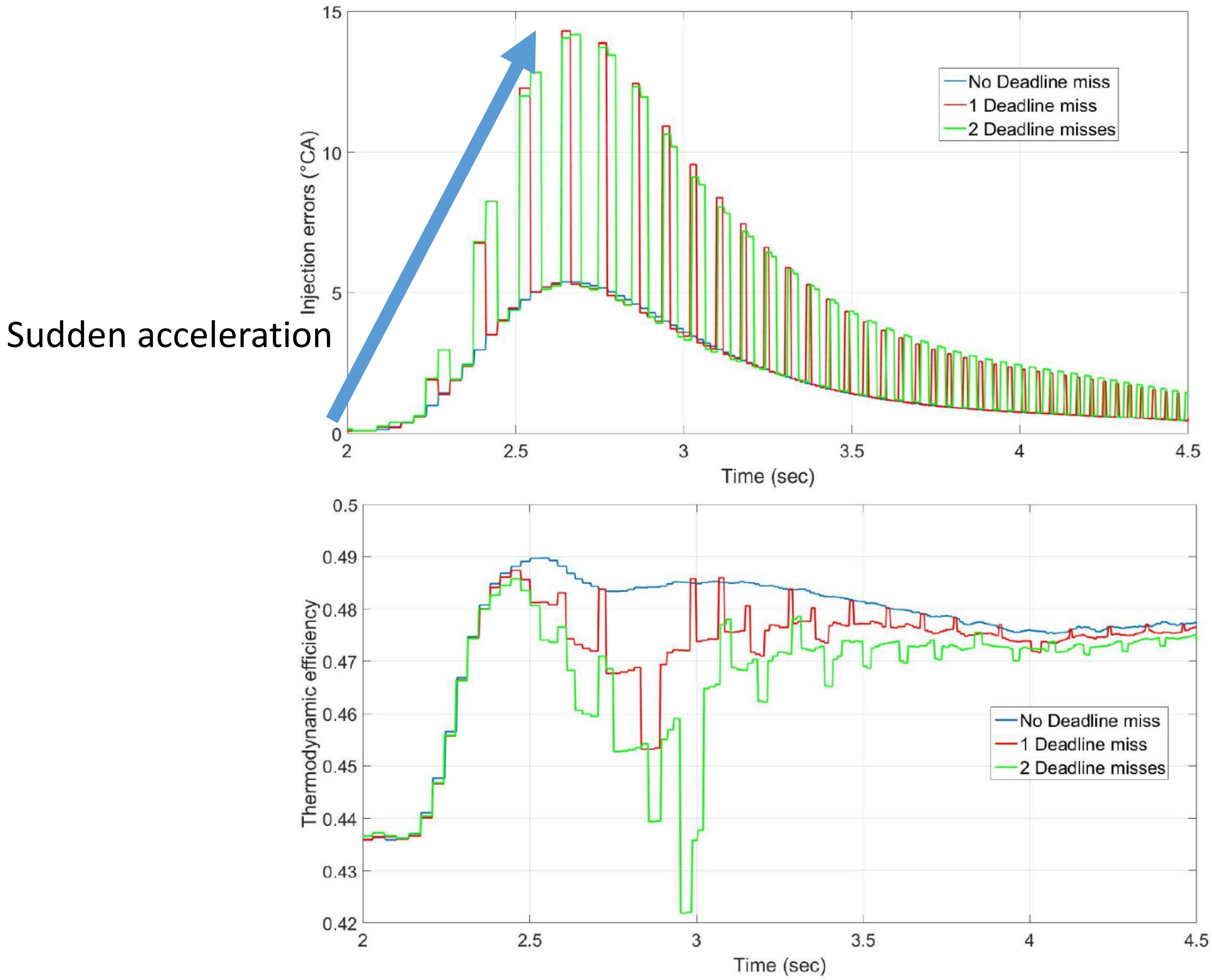
- Multiple injections reduce emissions:



Studying how timing impacts performance

- How thermodynamic efficiency changes with:
  - 1 deadline miss every two cycles
  - 2 deadline misses every two cycles

Studying how timing impacts performance



Conclusion

- Co-simulation framework of engine and control for obtaining more precise dependencies between timing and functionality
- Promising first results when considering multiple injections with respect to multiple performance metrics
- Need even better engine models!
- Need more accurate models of controls

Future work

- Better characterization of deadline miss impact on performance
- Integrate everything in a workflow for improving design of controller and scheduler
- Extend TRes framework for multicore support
- Include network model and memory access

Any questions?

Thank you!