25TH INTERNATIONAL CONFERENCE ON EMERGING TECHNOLOGIES AND FACTORY AUTOMATION
ETFA 2020
VIENNA, AUSTRIA
SEPTEMBER 08TH-11TH 2020

DTU TTTech

Mapping and Scheduling Automotive Applications on ADAS Platforms using Metaheuristics

Shane D. McLean[1], Silviu S. Craciunas[2], Emil Alexander Juul Hansen[1], Paul Pop[1]

[1]Technical University of Denmark, Kongens Lyngby, Denmark

[2]TTTech Computertechnik AG, Vienna, Austria

Autonomous vehicles/Assisted driving

TTTech

Safety – less accidents

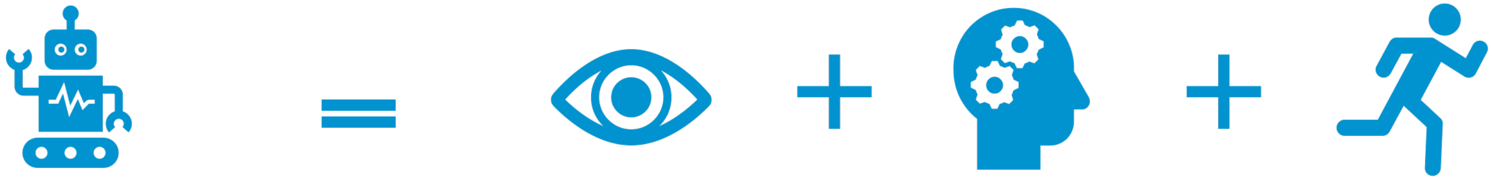
Reduced Emissions due to lower fuel consumptions

Better Health due to less stress through traffic jams

Increased highway capacity

An autonomous vehicle can be understood as a robot

TTTech



Robot = Sense = See + Think = + Act

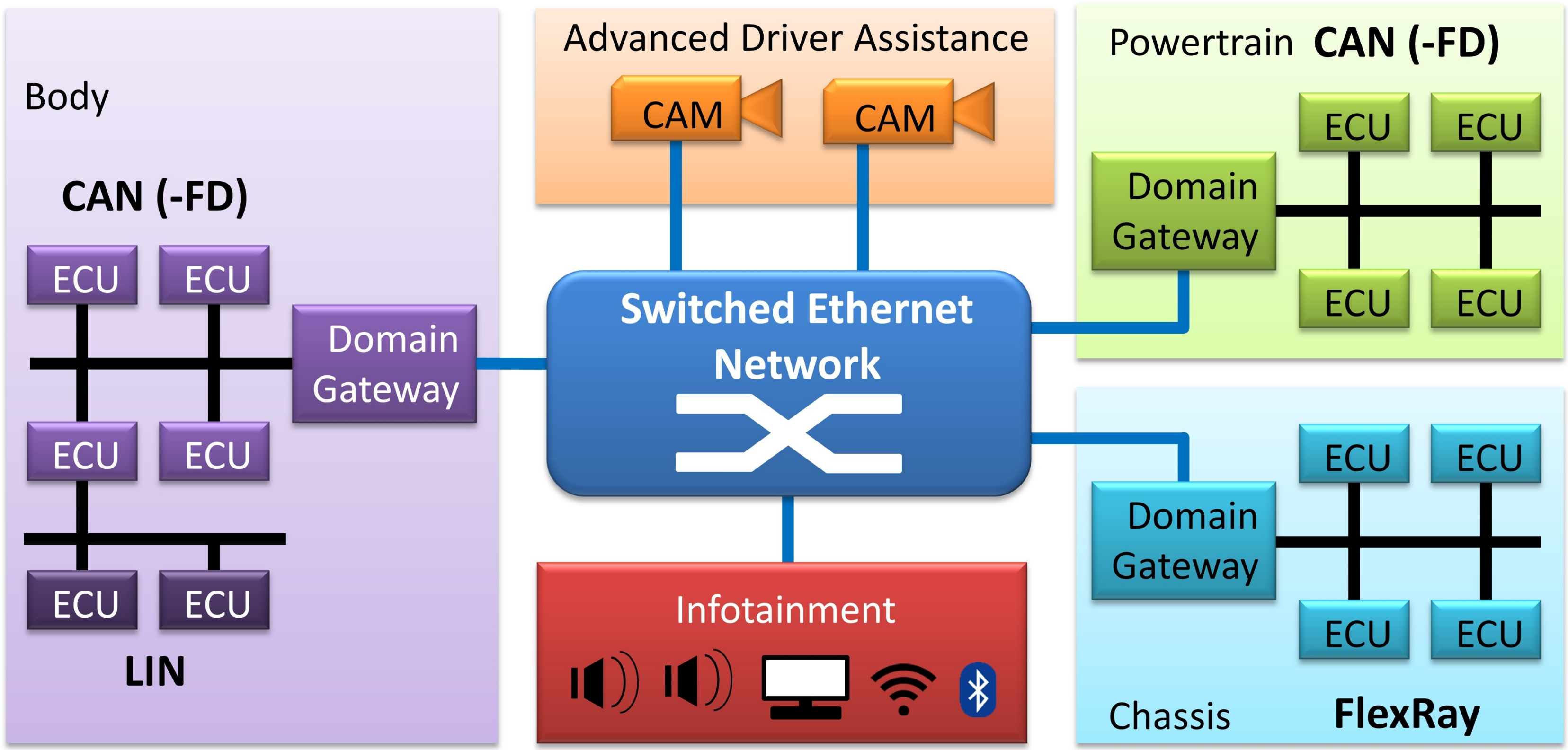
Sensors

Compute

Brake Actuators
Steering Actuators
Electric Drivetrain

Modern autonomous vehicles

Traditional approach based on distributed ECUs and separated domains, interconnected through different technologies (ETH, CAN, FlexRay)



Source: Ernst et al. - Ethernet as Future Automotive Communication Backbone

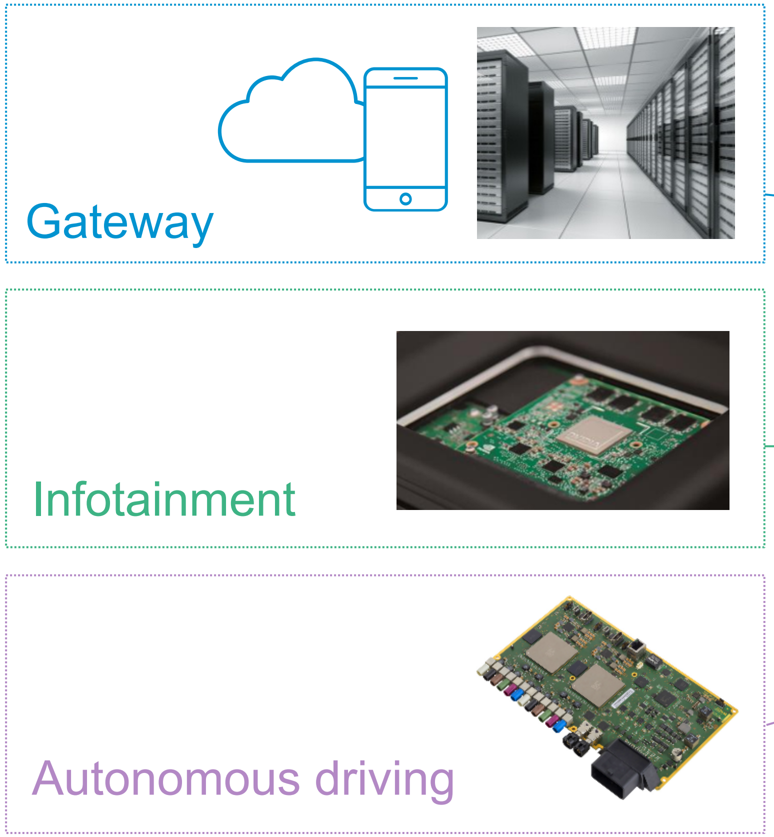
Modern autonomous vehicles – issues

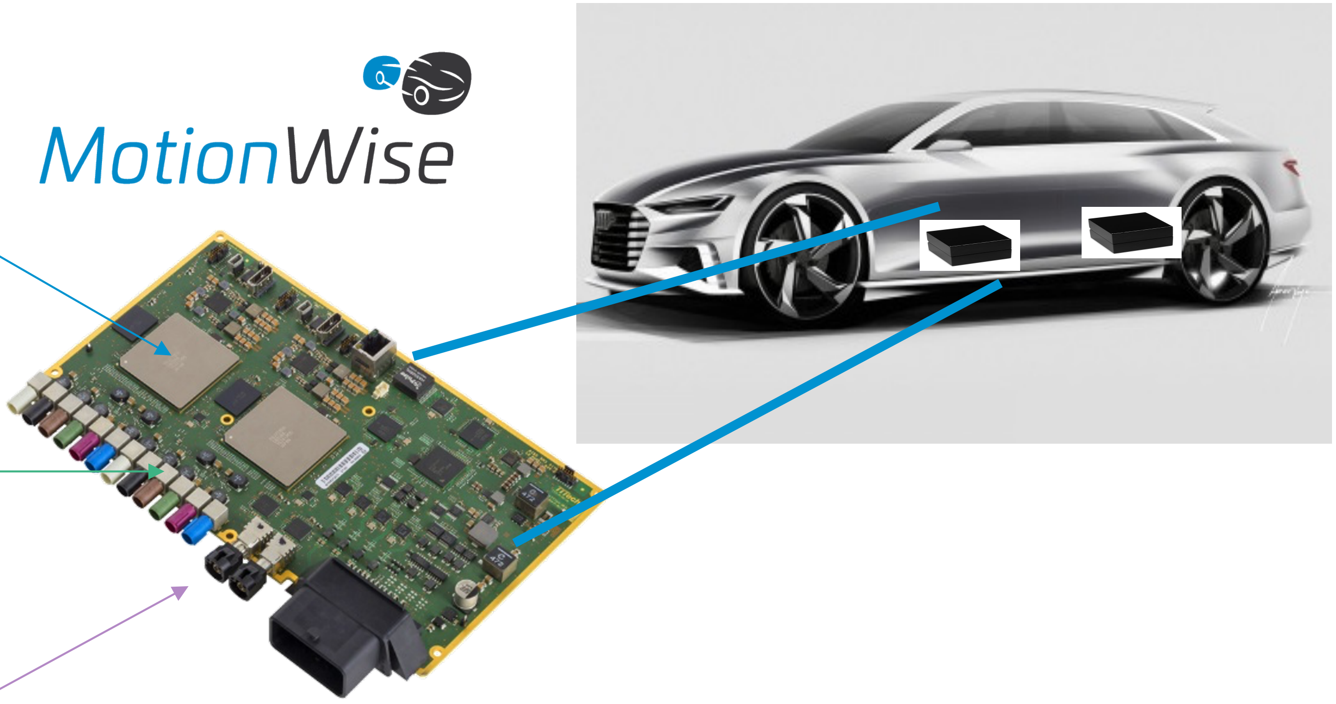
TTTech

- Rapid growth of software functionality and the necessary compute performance cannot be addressed with current electronics architecture and ECUs
- Too many ECU's with too little processing power and memory
- Limitation of the domain concept (development cost, replication of basic software functions, sources of failure, maintenance cost)
- Fail-operational requirement for level 4 autonomous driving:
  - The domain concept is not sustainable for L4/L5 autonomous driving.
  - Autonomous driving functions require the integration of cross-domain information and functions.

From distributed, separate ECUs ⟫ In-car Computing Platform

ICCP - Integrated platform





Integrated platform:
- from hardware to software
- from distributed to centralized

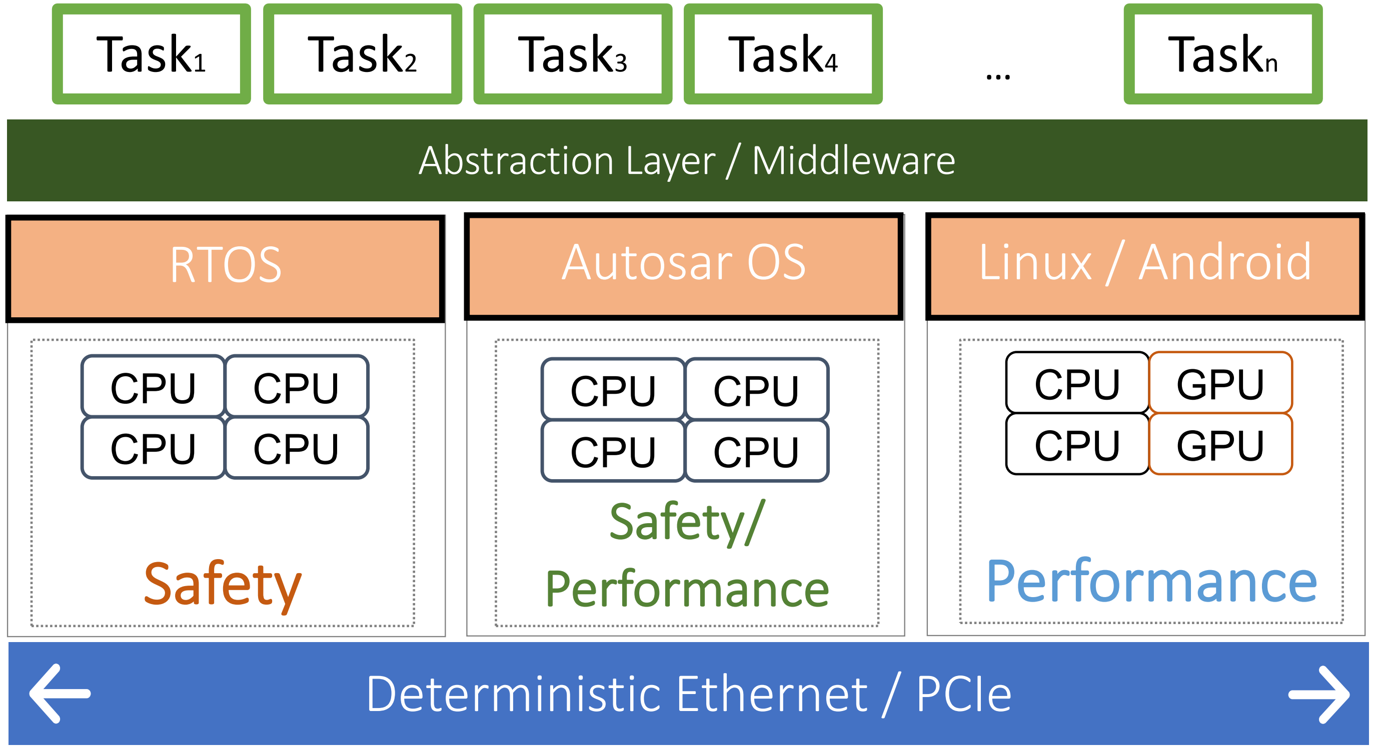
RazorMotion

Processing Resources:
1x Renesas RH850P/1H-C (ASIL D MCU with lockstep cores @ 240MHz)
2x Renesas R-Car H3 (ASIL B SoC with 4x Cortex A57, 4x Cortex A53, 1x Cortex R7, 1x IMP-X5, 1x IMG GX6650 GPU)

Video Interfaces:
12 x camera inputs (GMSL) incl. remote supply (PoC) 2 x display outputs (FPD-Link III)

Communication Interfaces:
4x OABR 100BASE-T1 2 x FlexRay (A/B channel) – wakeup capable 2 x HS-CAN – wakeup capable 4 x CAN-FD 2 x LIN I/O Interfaces 2 x analog/digital inputs 2 x high side outputs 1 x sensor supply output (5V)

Integrated Platform



Different multi-core CPUs:

- process the information arriving from a variety of sensors (radar, ultrasonic sensors, cameras, LIDAR, etc.)
- run control loops
- run other utility functions (lane keeping/changing assistant, emergency braking, logging, etc.)

Heterogeneous multi-core multi-SoC platform featuring a variety of CPUs and GPUs running at different speeds, which are interconnected through either a deterministic Ethernet backbone (TSN) or through PCIe

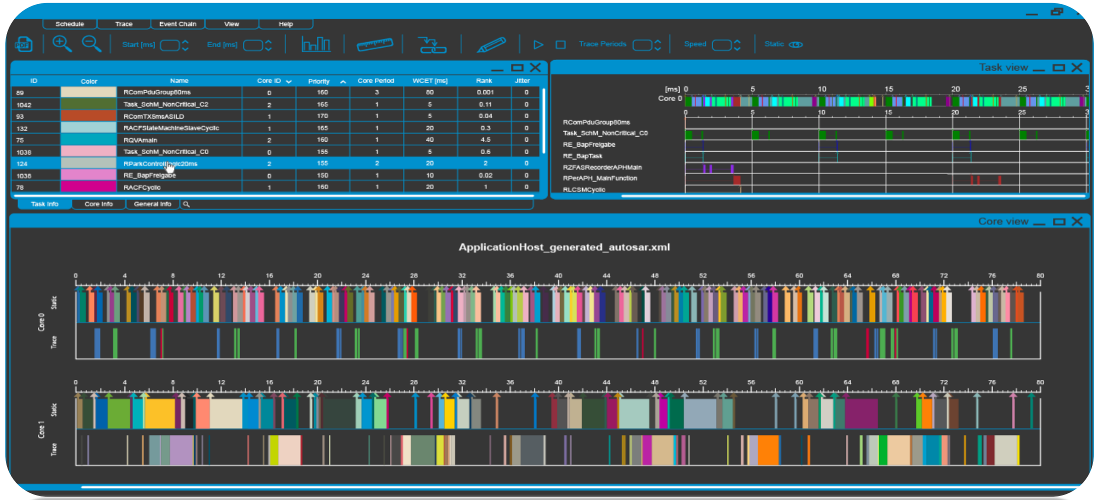
Integrated platform scheduling problem

Periodic hard real-time tasks with (WCET, Period) definition

- Are pre-assigned to CPUs (WCET is already scaled to speed)
- Can be pre-assigned to core, if not assigned, assignment will be part of the allocation problem
- Can have deadline, activation, jitter constraints
- Preemption is allowed, migration is not allowed

NP-complete

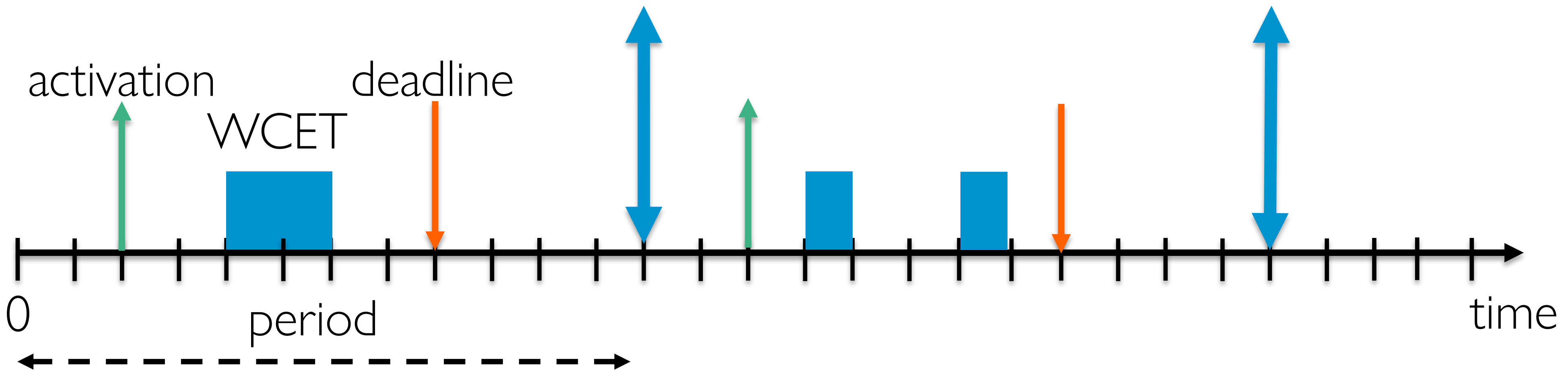
Result of scheduling is a static table which determines the exact timely behavior of tasks



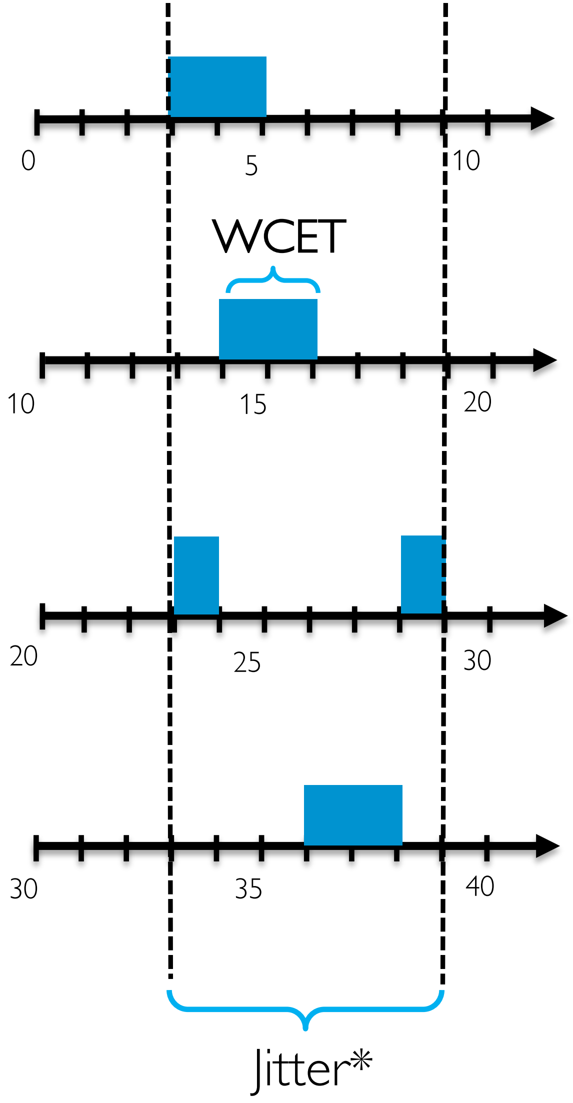
Different dimensions to the allocation problem:

- Assignment of tasks to cores/CPUs
- Scheduling of tasks
- Real-time requirements are met end-to-end

Real-time requirements – activation, deadline, period



Real-time requirements – jitter

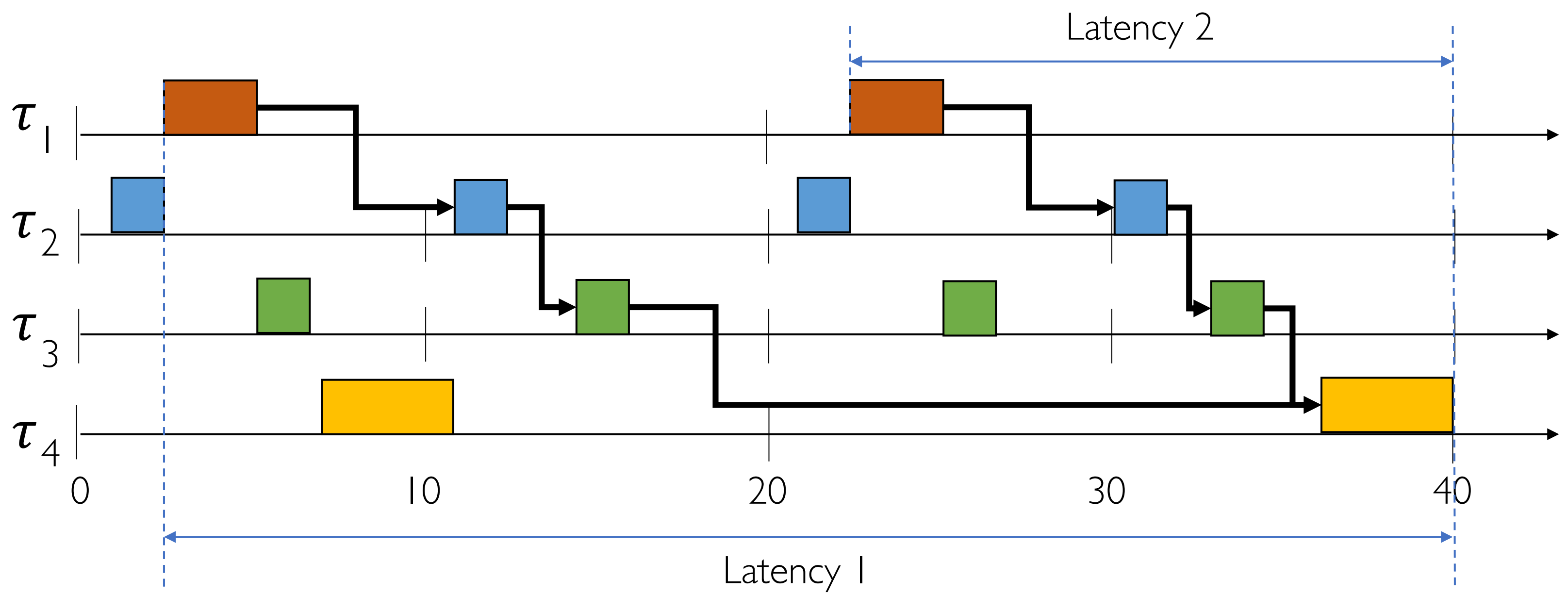


Jitter = Jitter* - WCET

Real-time requirements – dependency chains

Characteristic of automotive software – **Cause-effect chains**:

- provide additional timing and dependency requirements on the execution of tasks
- can span across multiple activation patterns
- include multiple tasks, even the same task multiple times
- have priorities and end-to-end latencies
- include communication latencies



SA algorithm

Simulated Annealing (SA)-based metaheuristic approach which uses an **EDF-based** heuristic to solve the task scheduling problem.

The scheduling heuristic allows task preemption by simulating an Earliest Deadline First (EDF) scheduling policy parameterized by task offsets and local deadlines decided by SA

Algorithm 1 SimulatedAnnealing($\mathcal{A}, \Gamma, s_0, t_s, cr, i$)

$t \leftarrow t_s$
$s \leftarrow$ ScheduleSynthesis($\mathcal{A}, \Gamma, s_0$)
$s^* \leftarrow s$
while *timeleft* **do**
  while $t > 1.0$ **do**
    for $k \leftarrow 1$ to i **do**
      $s' \leftarrow$ GenerateNeighbor($\mathcal{A}, \Gamma, s$)
      if Cost(s') $<$Cost(s) **then**
        $s \leftarrow s'$
        if Cost(s') $<$Cost(s^*) **then**
          $s^* \leftarrow s'$
        end if
      else if $exp(\frac{Cost(s)-Cost(s')}{t}) > random[0,1]$ **then**
        $s \leftarrow s'$
      end if
      $t \leftarrow t \cdot (1 - cr)$
    end for
  end while
end while
return s^*

EDF simulation

TTTech

EDF is an optimal online scheduling algorithm which at each time instant prioritizes the task with the earliest deadline

We can use it to generate a static schedule table – simulate EDF until 2*Hyperperiod + max_offset

Schedulability test: $\forall t_1 \in \Phi^\sigma, \forall t_2 \in \Delta^\sigma, t_1 < t_2 :$

$$\sum_{\tau_i \in \Gamma^\sigma} C_i \times \left(\left\lfloor \frac{t_2 - \phi_i - D_i}{T_i} \right\rfloor - \left\lceil \frac{t_1 - \phi_i}{T_i} \right\rceil + 1 \right)_0 \leq t_2 - t_1,$$

where

$$\Phi^\sigma \stackrel{def}{=} \{a_{i,j} = \phi_i + j \times T_i | \tau_i \in \Gamma^\sigma, j \geq 0, a_{i,j} \leq \lambda^\sigma\},$$

$$\Delta^\sigma \stackrel{def}{=} \{d_{i,j} = a_{i,j} + D_i | \tau_i \in \Gamma^\sigma, j \geq 0, d_{i,j} \leq \lambda^\sigma\},$$

$$\lambda^\sigma = max(\{\phi_i | \tau_i \in \Gamma^\sigma\}) + 2 \times lcm(\{T_i | \tau_i \in \Gamma^\sigma\}).$$

Two knobs to play around with: **offset** and **deadline** of each task

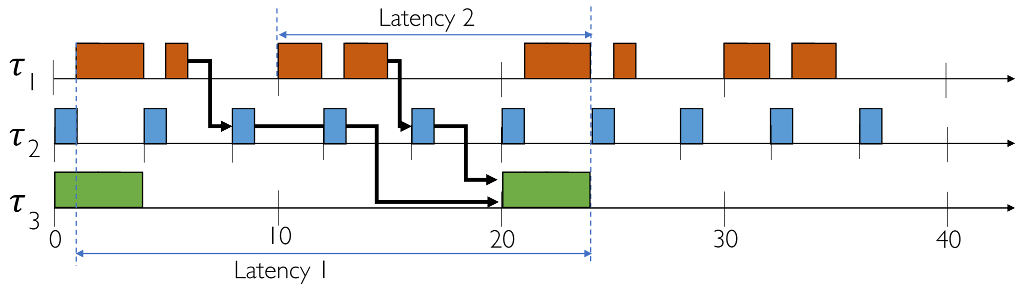
Simulated Annealing + EDF simulation

TTTech

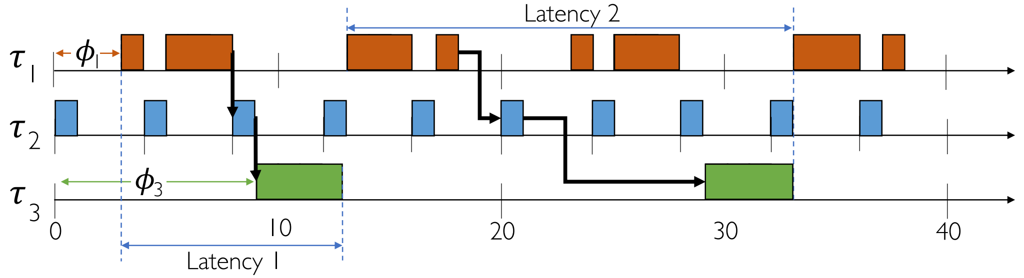
Simple Algorithm:

Generate initial candidate:

- task offsets = 0
- task deadlines = Period
- task to core assignment based on best-fit/first-fit (load balancing)



(a) End-to-end task chain latencies **not** satisfied



Simulated Annealing Loop:

- EDF schedulability test/EDF simulator
- Generate new candidate through performing design transformations
  - SwapTask, AdjustOffset and AdjustDeadline
- Evaluate a solution based on the cost metric

$$Cost(s) = \begin{cases} \dfrac{\sum_{\aleph_i \in \mathcal{L}_\aleph} \frac{l_i}{L_i} \cdot p_i}{|\mathcal{L}_\aleph|} \cdot w_1 & \text{if } \chi(s) = \text{true} \\ w_1 + \rho_\aleph + \rho_D + \rho_J & \text{if } \chi(s) = \text{false} \end{cases}$$

Experimental evaluation

Five test cases, ranging from 100% to 500% in scale, i.e., for ADAS1x100% the application contains 151 tasks and 31 chains using a model of the architecture

A test case is a scenario consisting of 30 synthetically generated task sets, with each undergoing 30 trials (900 trials for each algorithm)

EVALUATION RESULTS ON SYNTHETIC TEST CASES

Test case	**Time**	**Greedy**							**SA**							**GA**						
		Chains			**Jitter**			**Sched.**	**Chains**			**Jitter**			**Sched.**	**Chains**			**Jitter**			**Sched.**
		Min	*Avg*	*Max*	*Min*	*Avg*	*Max*		*Min*	*Avg*	*Max*	*Min*	*Avg*	*Max*		*Min*	*Avg*	*Max*	*Min*	*Avg*	*Max*	
ADAS1x100%	1 hour	0.97	0.98	1.00	0.58	0.61	0.68	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
ADAS1x200%	2 hours	0.97	0.99	1.00	0.55	0.67	0.75	1.00	0.98	1.00	1.00	0.94	1.00	1.00	1.00	0.98	1.00	1.00	0.71	0.95	1.00	1.00
ADAS1x300%	3 hours	0.97	0.99	1.00	0.52	0.64	0.72	1.00	0.97	0.99	1.00	0.70	0.87	1.00	1.00	0.97	0.99	1.00	0.70	0.88	1.00	1.00
ADAS1x400%	4 hours	0.97	0.97	0.98	0.52	0.64	0.73	1.00	0.97	0.99	1.00	0.69	0.80	0.88	1.00	0.94	0.99	1.00	0.70	0.81	0.92	1.00
ADAS1x500%	5 hours	0.97	0.98	0.98	0.51	0.62	0.70	1.00	0.95	0.98	0.99	0.63	0.78	0.86	1.00	0.95	0.98	1.00	0.64	0.79	0.87	1.00

Real-world test-cases with 151 tasks and 31 chains

EVALUATION RESULTS ON REALISTIC TEST CASES

Test case	Time	**Greedy**			**SA**						
		Chains	**Jitter**	**Sched.**	**Chains**			**Jitter**			**Sched.**
					Min	Avg	Max	Min	Avg	Max	
ADAS1	3.20	0.81	0.37	1.00	0.97	0.99	1.00	0.95	0.99	1.00	1.00
ADAS2	6.40	0.65	0.21	1.00	0.94	0.99	1.00	0.84	0.99	1.0	1.00
ADAS3	13.20	0.48	0.21	1.00	0.84	0.99	1.00	0.74	0.97	1.0	1.00

Vienna, Austria (Headquarters)	USA	Japan	China
Phone +43 1 585 34 34-0 office@tttech.com	Phone +1 978 933 7979 usa@tttech.com	Phone +81 52 485 5898 office@tttech.jp	Phone +86 21 501529250 china@tttech.com

www.tttech.com