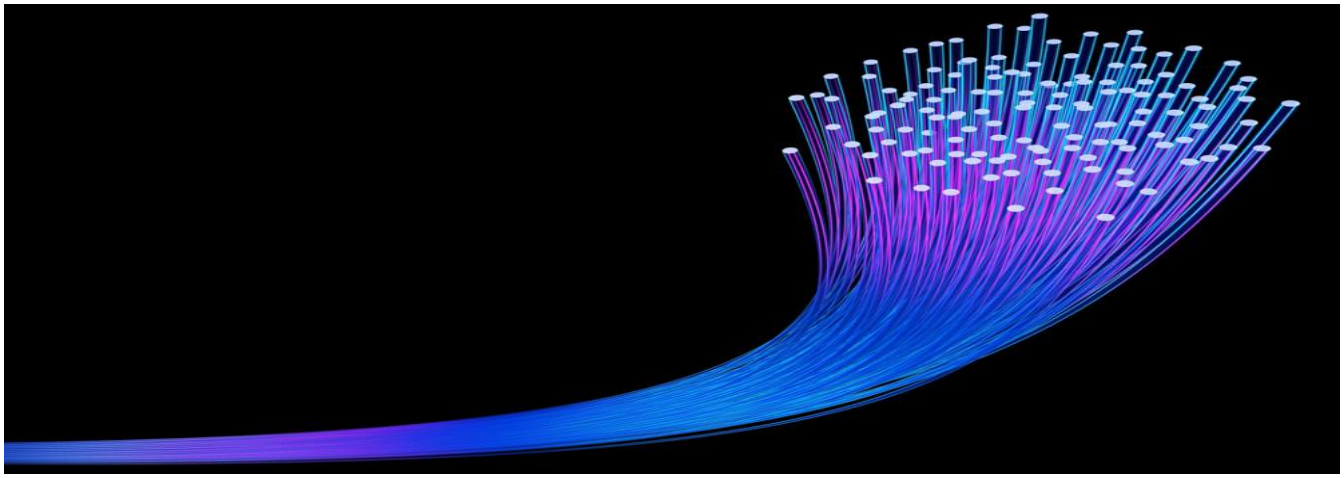
DreamCloud



Hard Real-time Guarantee of Automotive Applications during Mode Changes

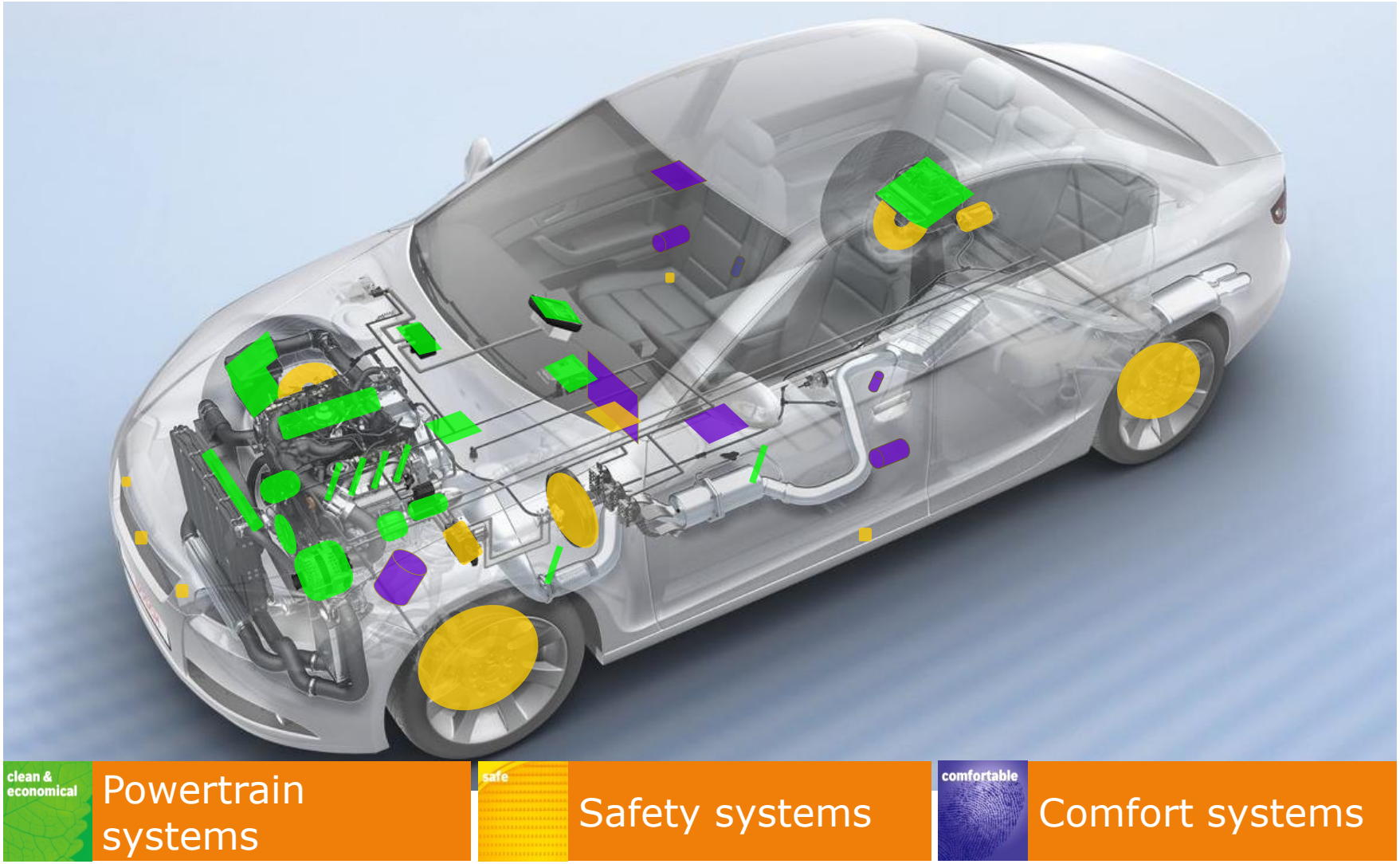
P. Dziurzanski, A.K. Singh, L.S. Indrusiak
University of York

B. Saballus
Robert Bosch GmbH

Overview

- Automotive technology: an overview
- Modes in Engine Control Units (ECU)
- DemoCar use case
- Design flow
- Conclusion

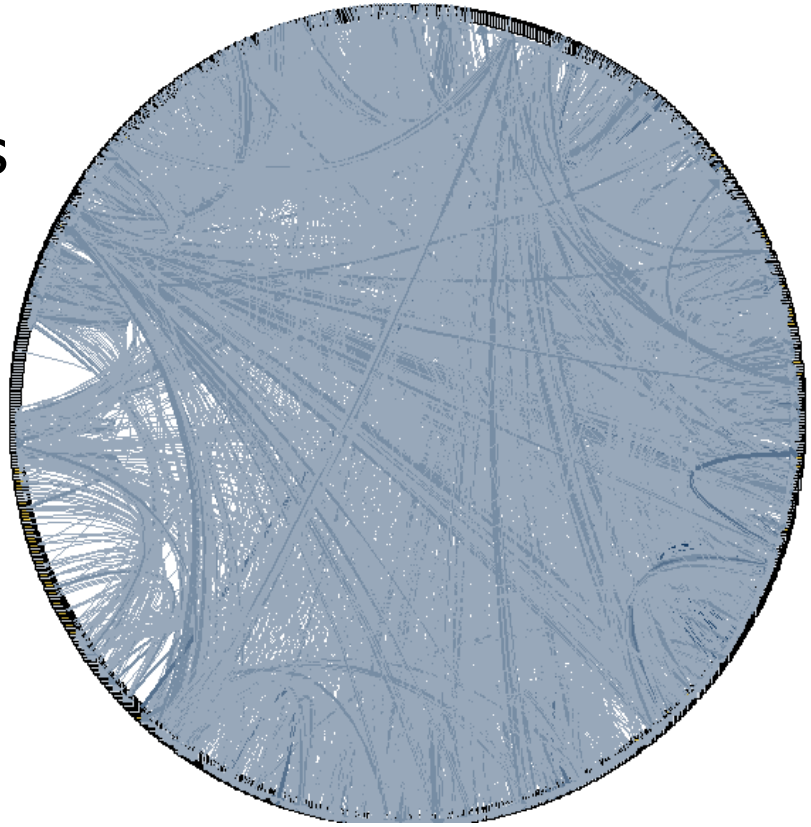
Automotive technology: an overview



Source: Robert Bosch GmbH, Future mobility: automated, connected, electric, Belgian Insurance Conference, 2014

Engine control system

- 800 – 1600 SW components
- 1000 – 1500 runnables (basic executable entities) in about 10 tasks
- 3000 – 6000 sub functions
- 7000 – 15000 messages
- hard real-time and safety requirements



Source: Robert Bosch GmbH

Current vs future trends in cars

DreamCloud

2015

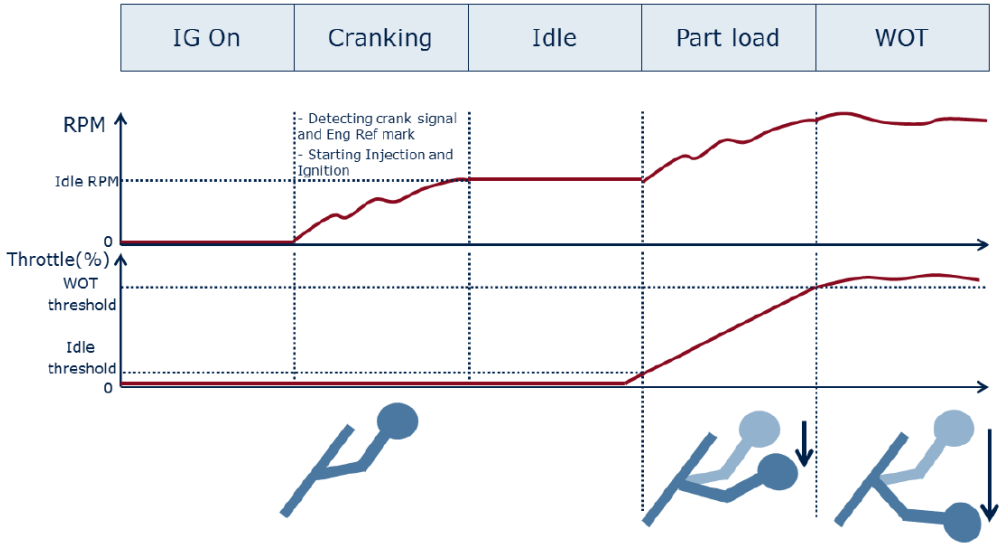
- different HW units for different functionality
- more than 100 ECUs in a premium car
- static distribution
- bus-based

Future

- multiple applications per HW unit
- dynamic distribution
- Network on Chips

Limited number of operating modes

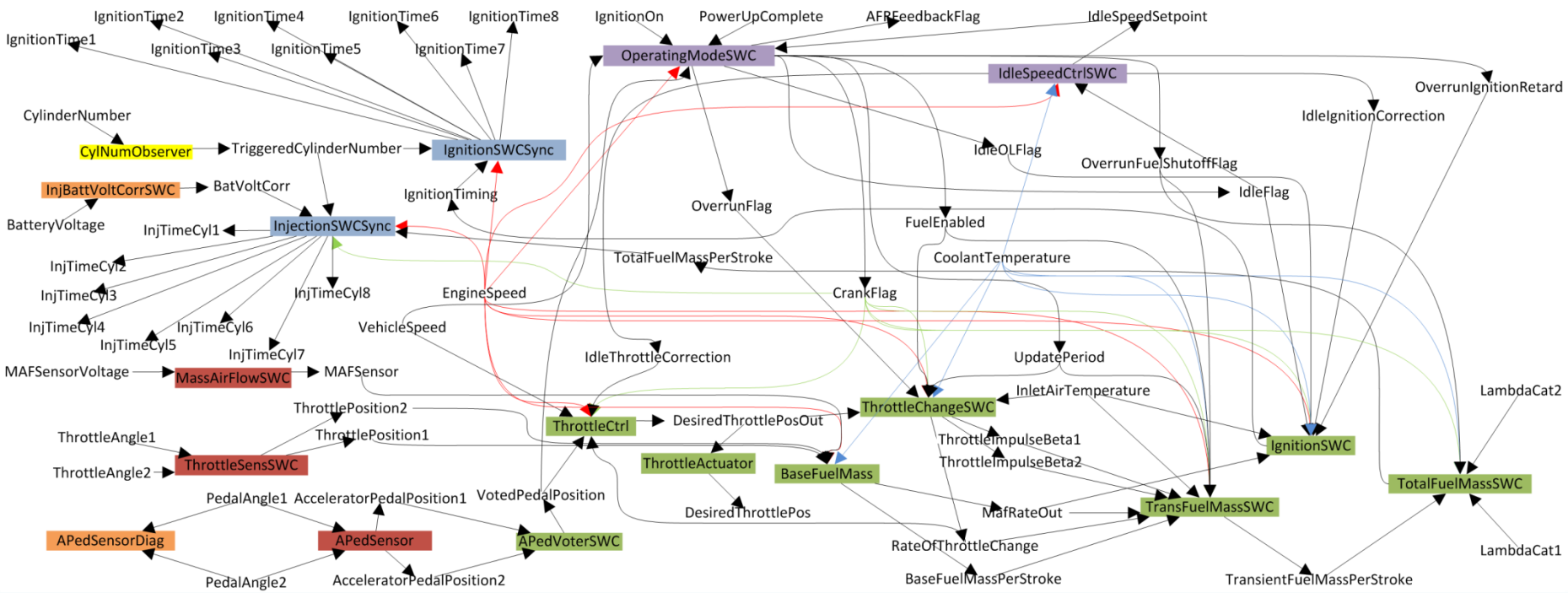
- **Ignition key-On**: engine off state
- **Cranking**: engine starting state
- **Idle**: a throttle valve is not opened
- **Part load**: a throttle valve is partially opened
- **Wide open throttle**: a throttle valve is wide open



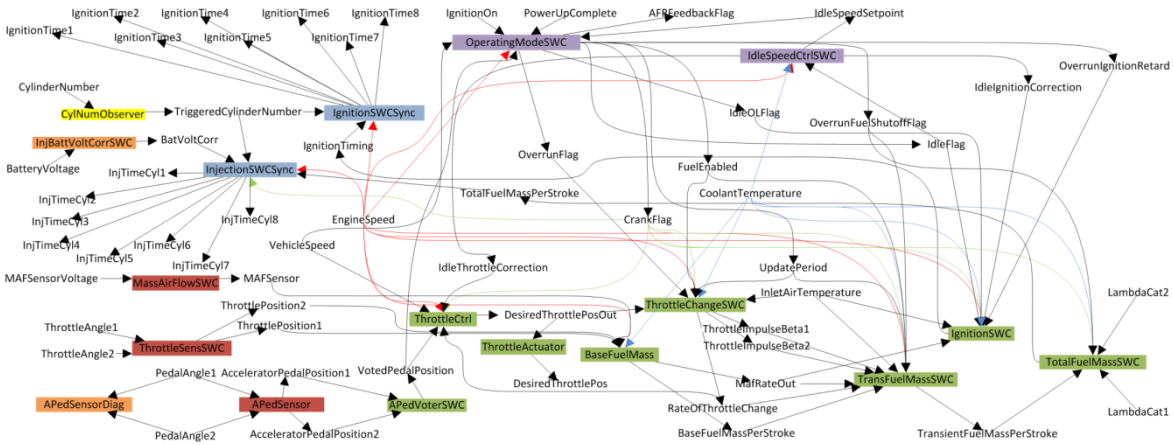
Source: J.S. Park et al., Mode-Dynamic Task Allocation and Scheduling for an Engine Management Real-Time System Using a Multicore Microcontroller, SAE Int. J. Passeng. Cars – Electron. Electr. Syst. 7(1):133-140, 2014.

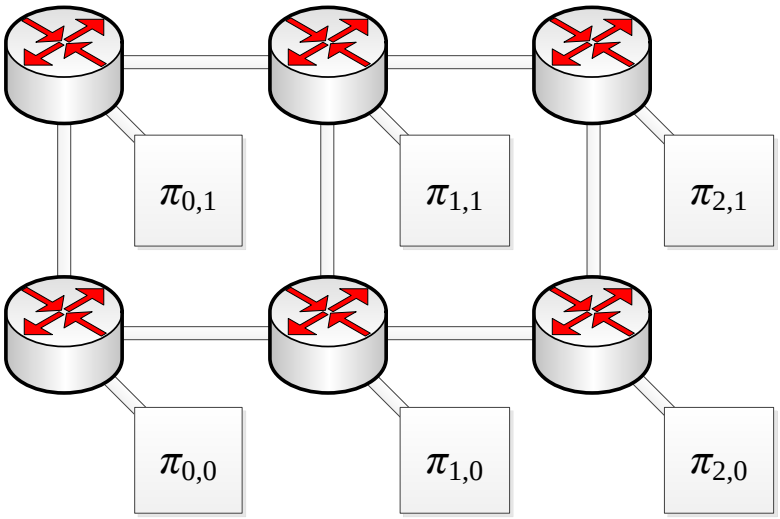
DemoCar- Minimal functionality gasoline Engine Control Unit

- 18 runnables
- 6 tasks
- 61 labels (data elements that can be read or written by runnables)



Problem formulation

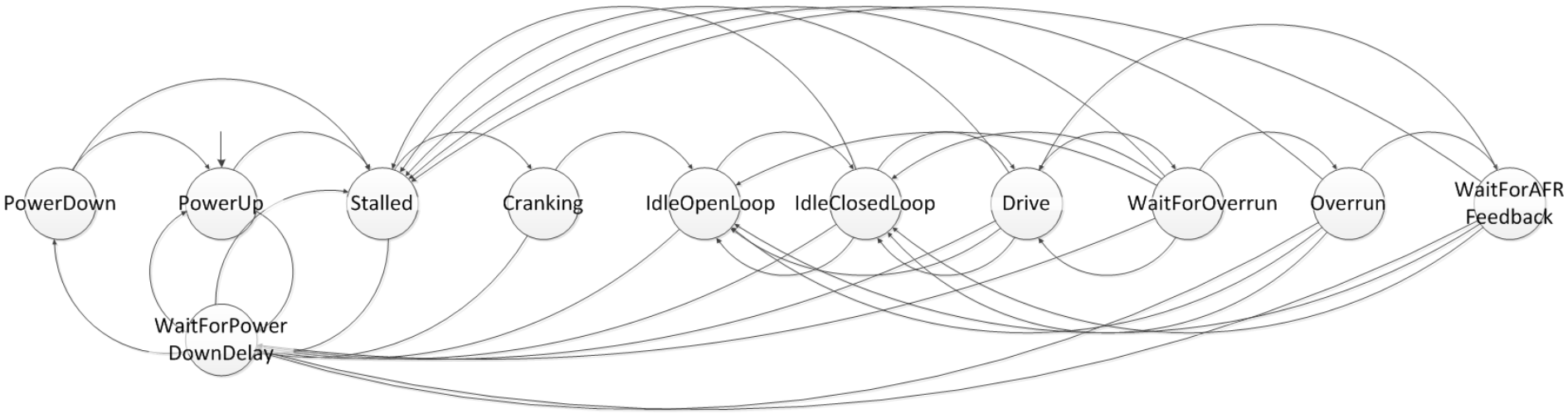




Goals:

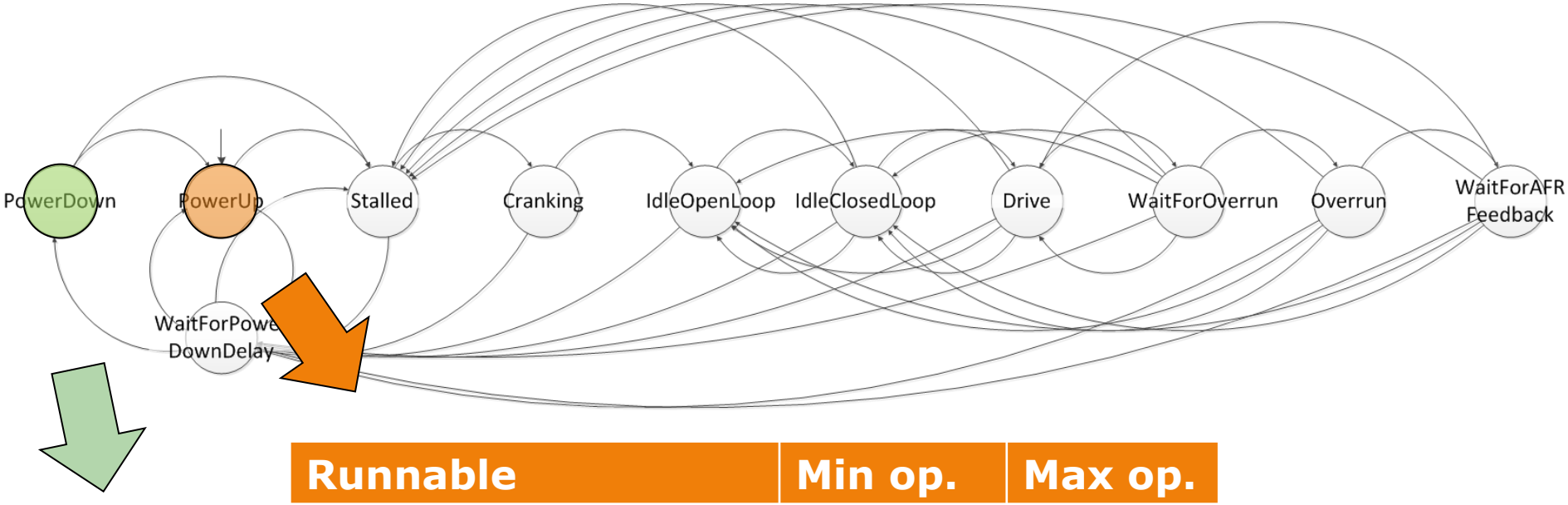
- no deadline violation
- minimal #used resources
- minimal energy dissipation

Finite State Machine describing mode changes in DemoCar use case



The current state is stored in a certain variable.

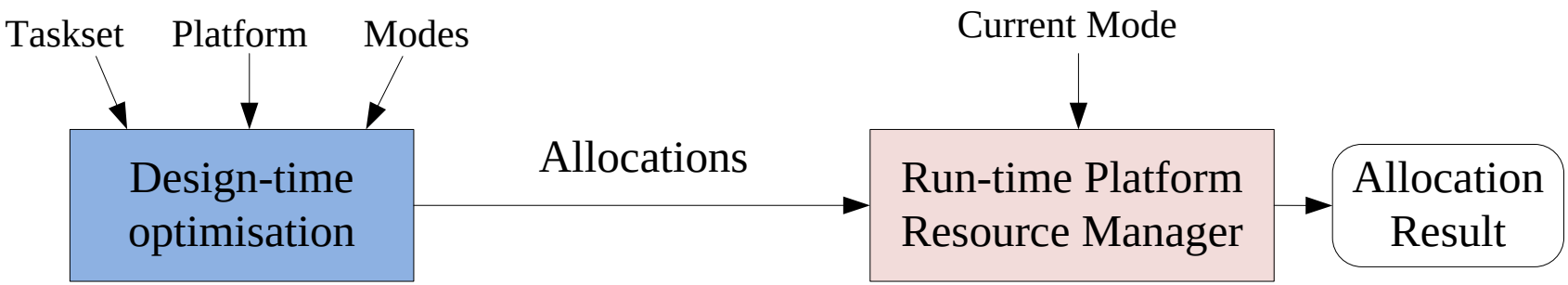
Runnables execution time in different modes



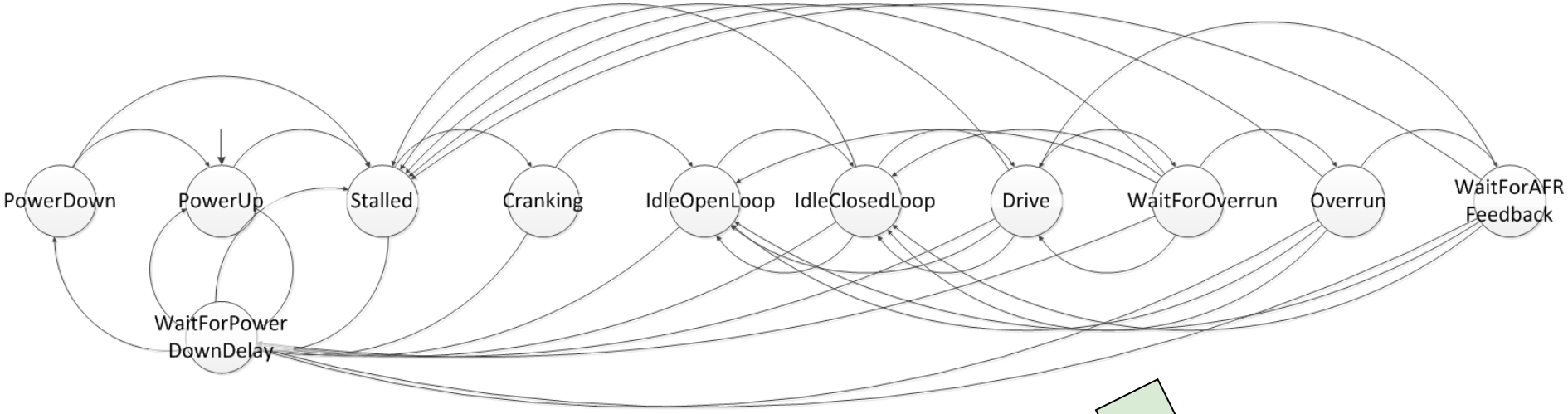
Runnable	Min op.	Max op.
CylNumObserver	134	345
InstructionsDeviation	3728	5921 !

Runnable	Min op.	Max op.
CylNumObserver	245	543 !
InstructionsDeviation	728	921

Steps of the proposed dynamic resource allocation method

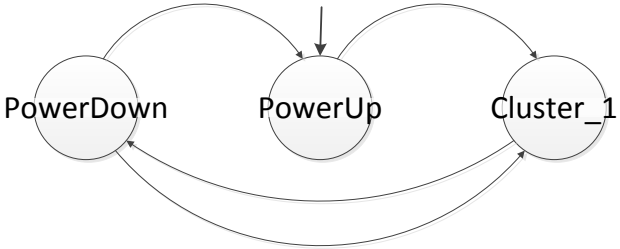


Mode detecting / clustering

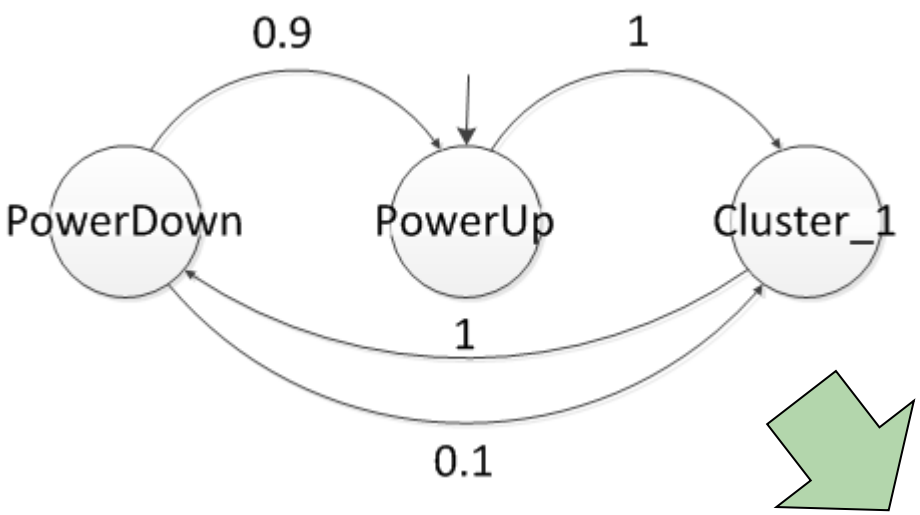


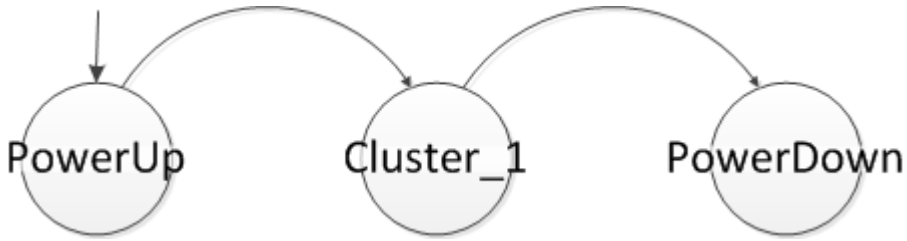
Reasons:

- Runnables have similar runtime and resource consumption in neighbouring modes.
- Transition is required to be done immediately.



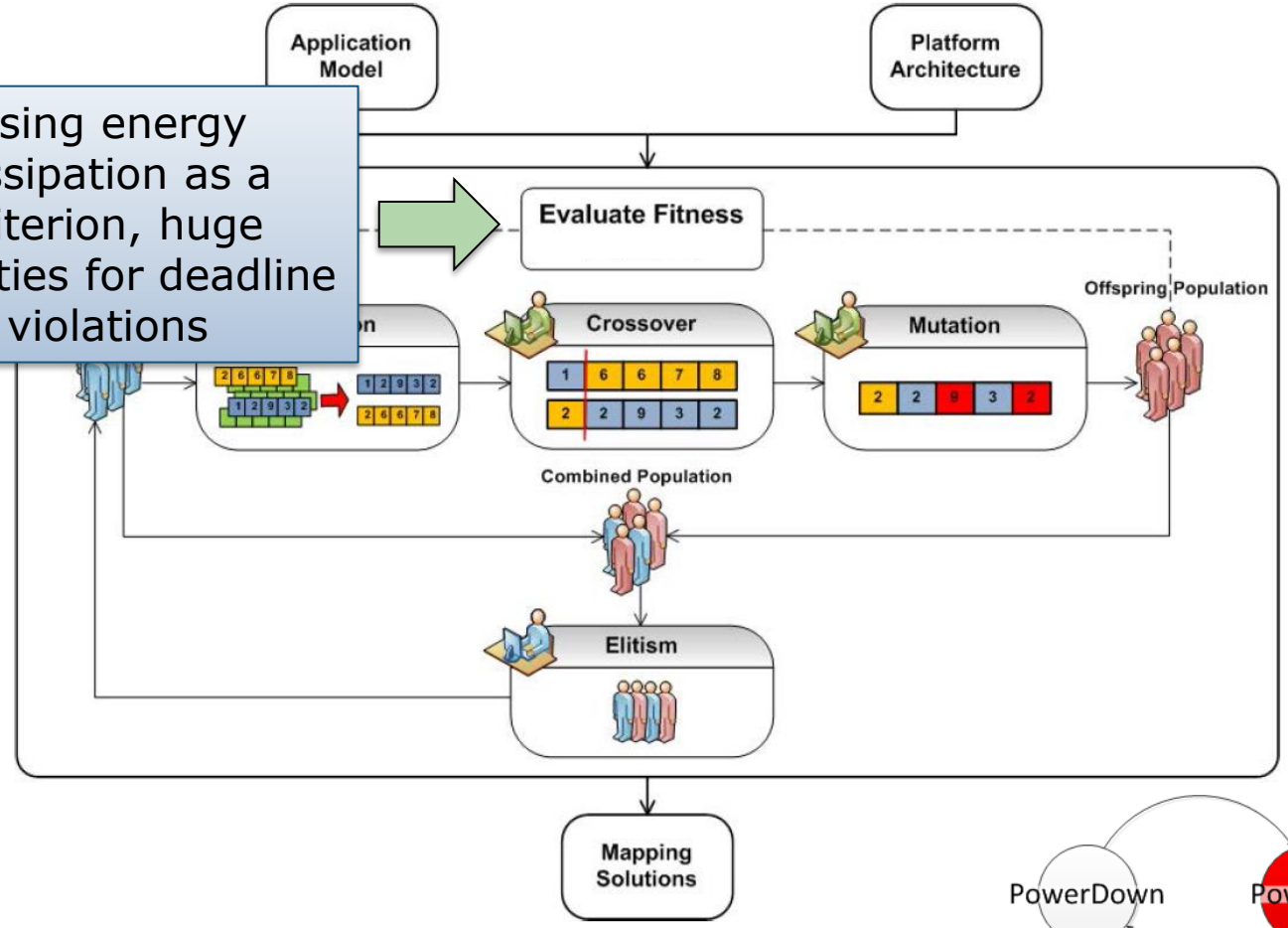
Spanning tree construction

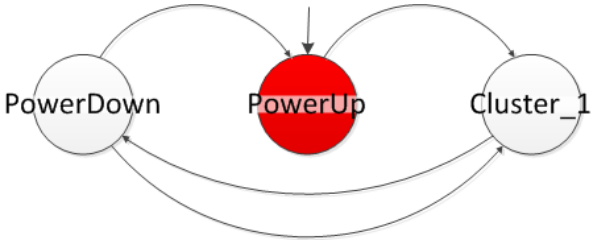




Static mapping for initial mode

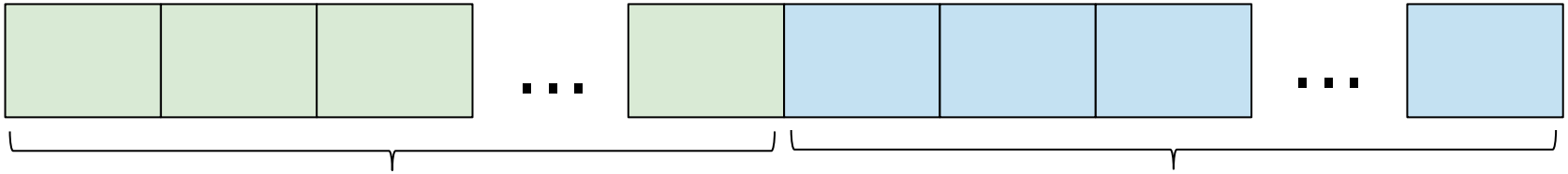
Using energy dissipation as a criterion, huge penalties for deadline violations

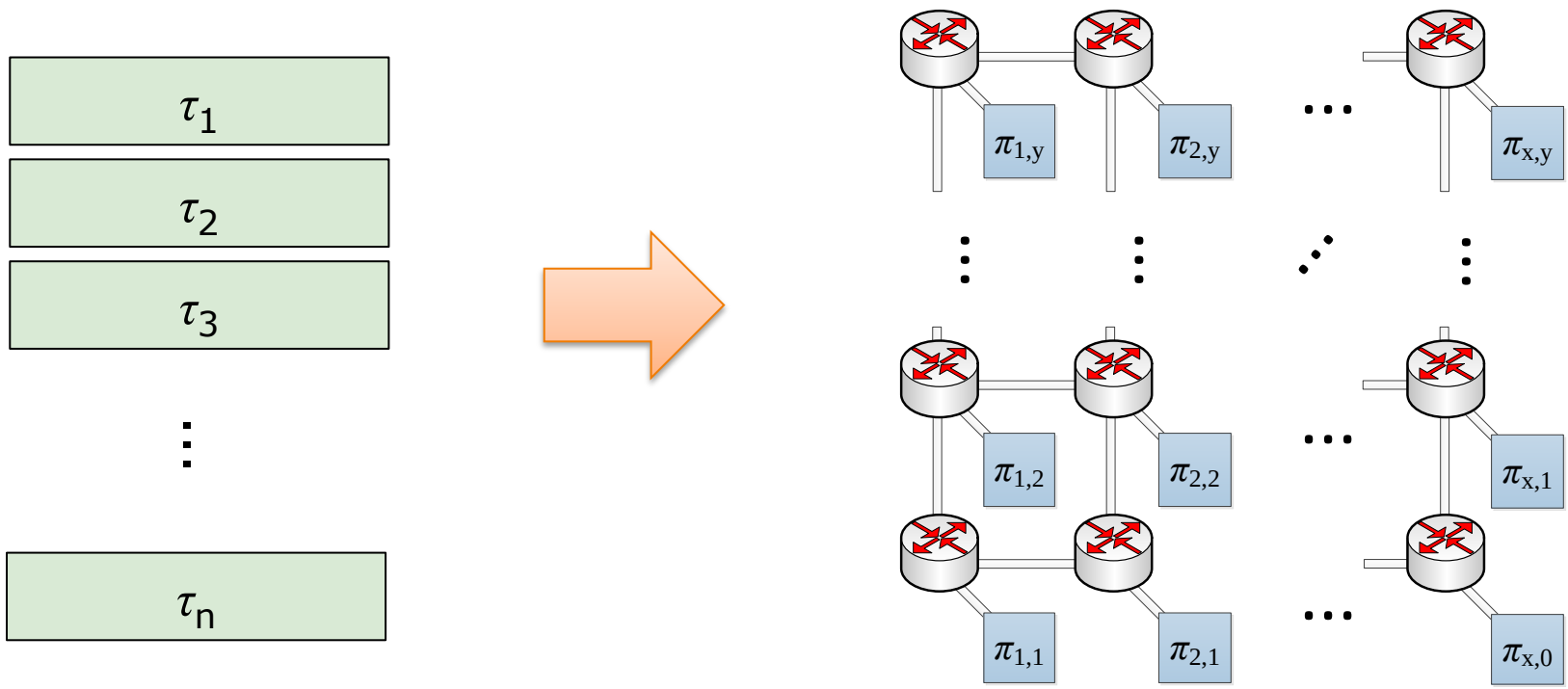




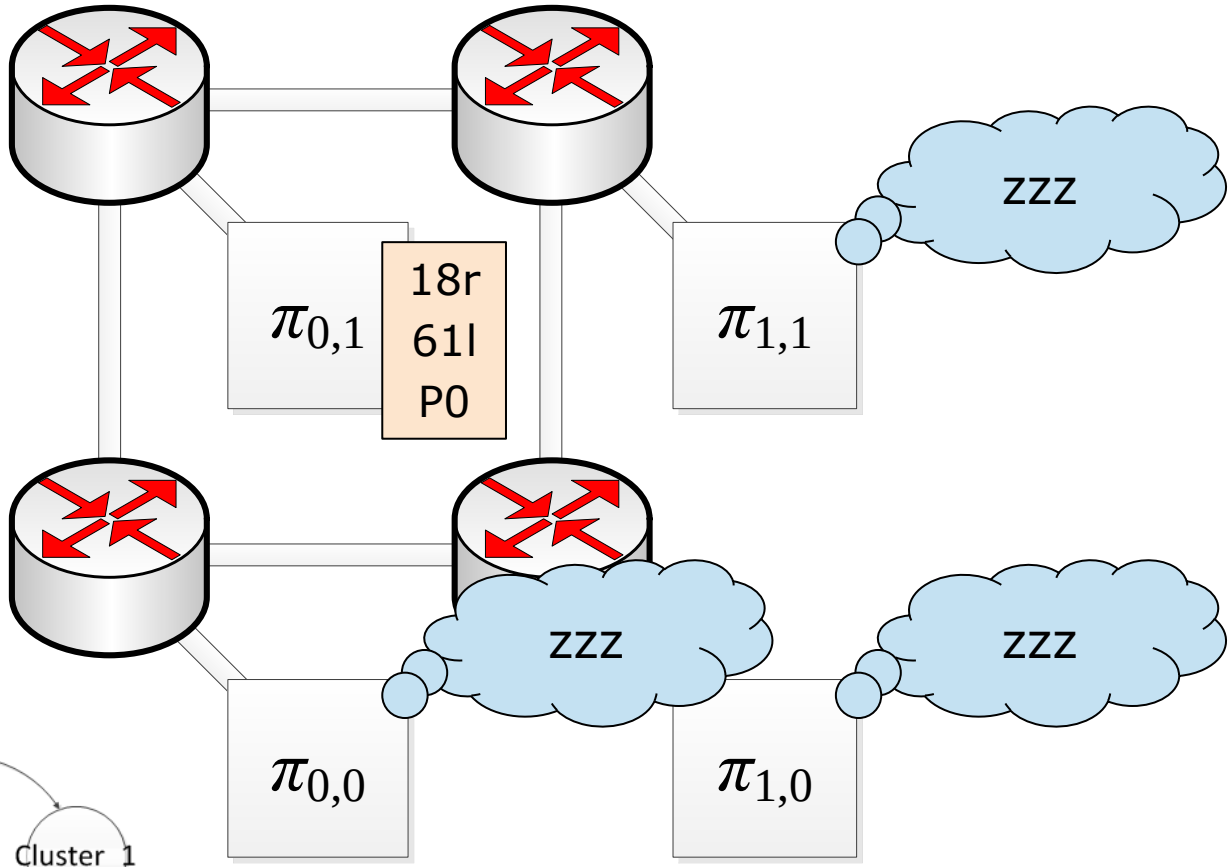
M. Norazizi

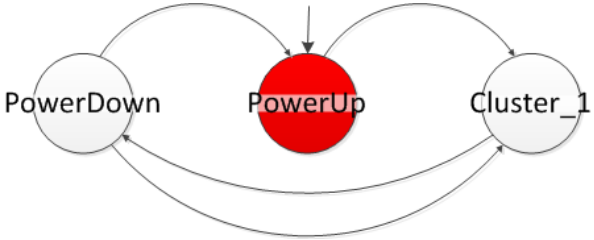
Genes in chromosomes





Optimization result

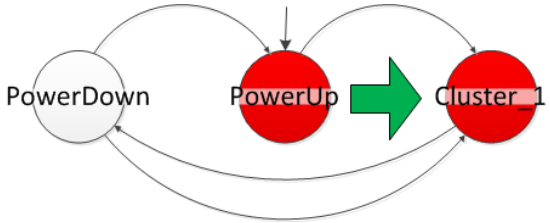




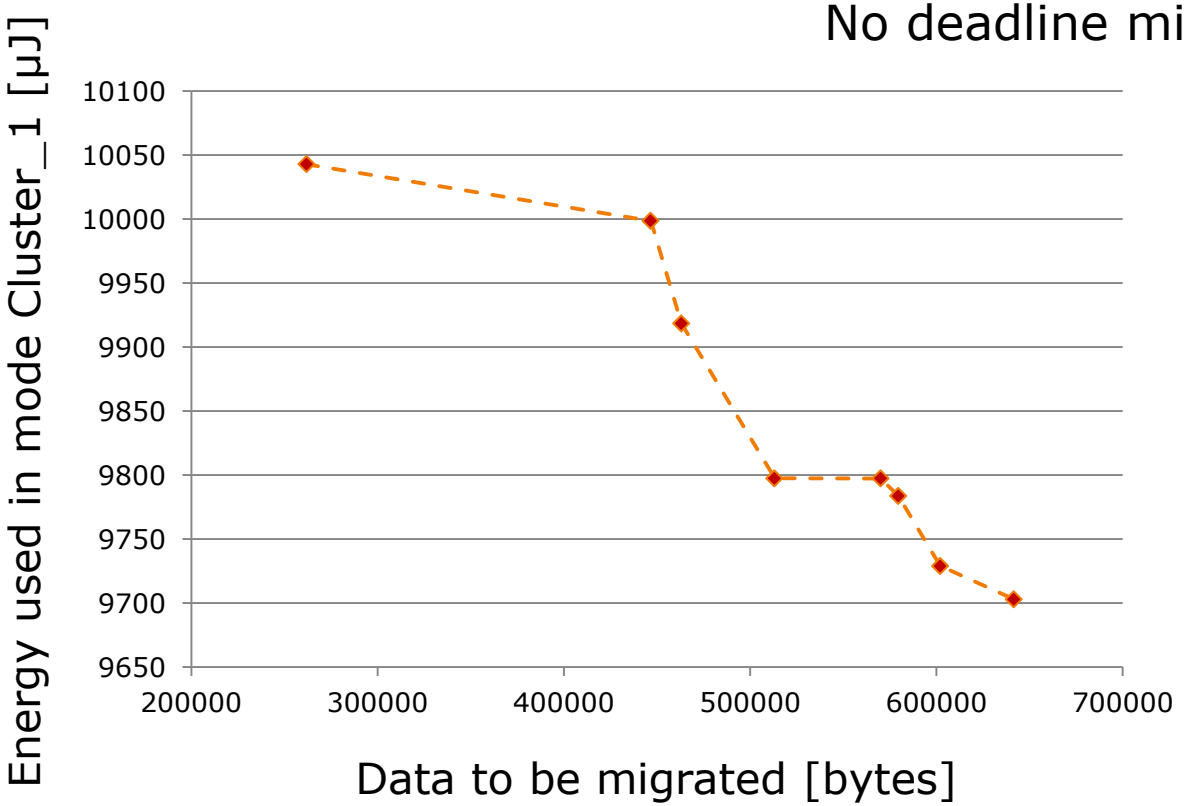
Static mapping for non-initial modes minimizing amount of migrated data

- Multicriteria optimization:
  - migration cost from the previous mode
  - energy dissipation
- Huge penalties for deadline violations

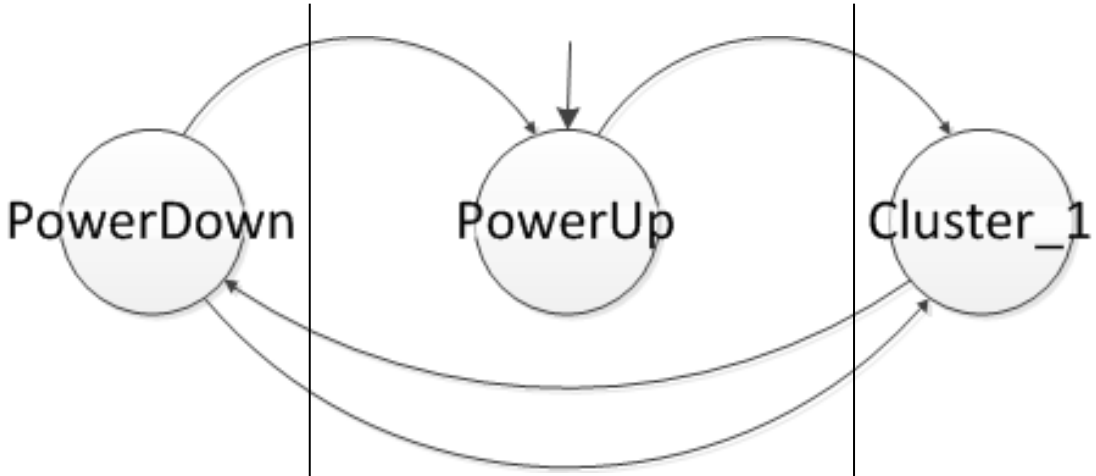
Pareto frontier - Cluster_1



No deadline misses.

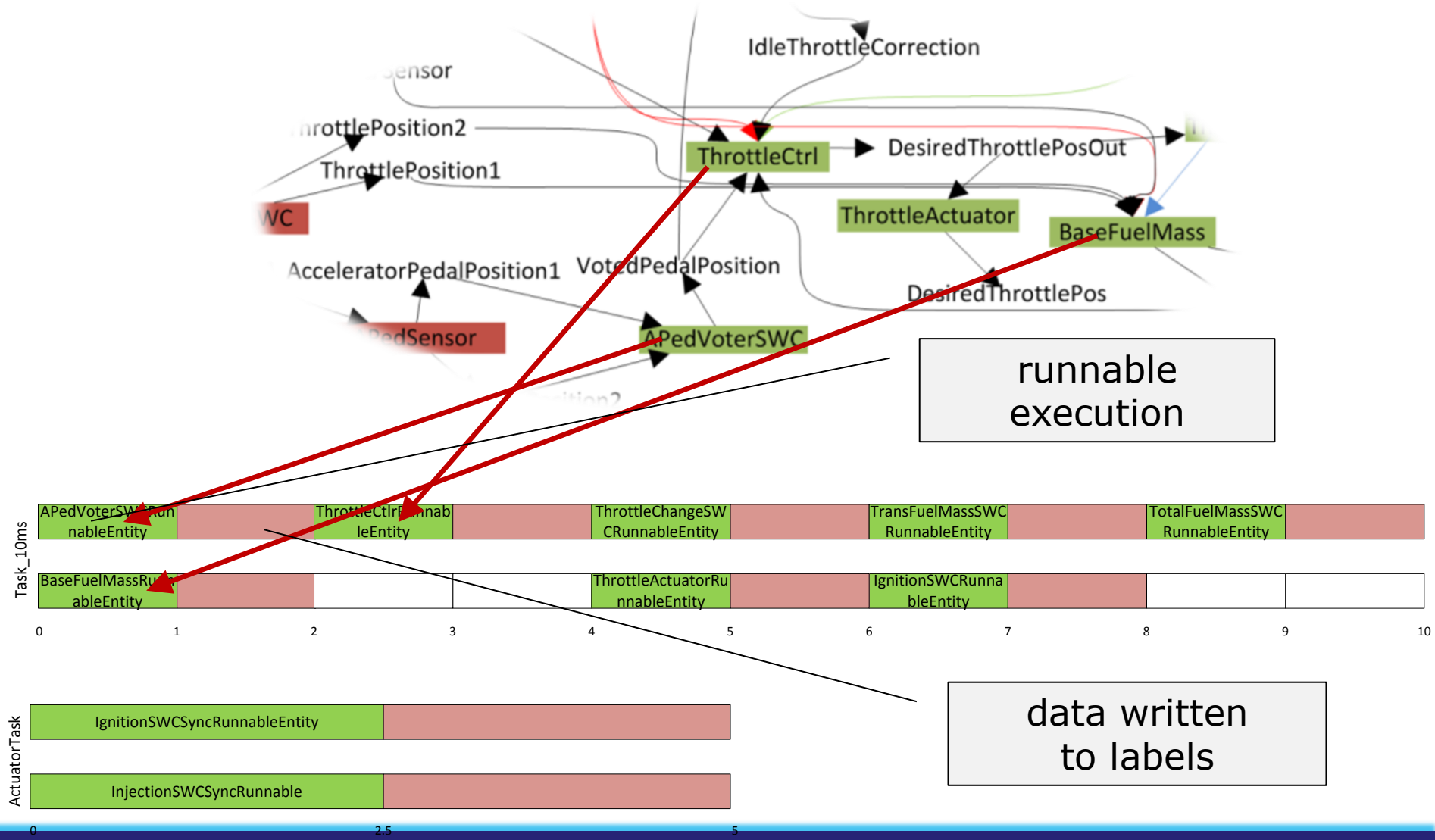


Idle hardware in states - 2x2 NoC

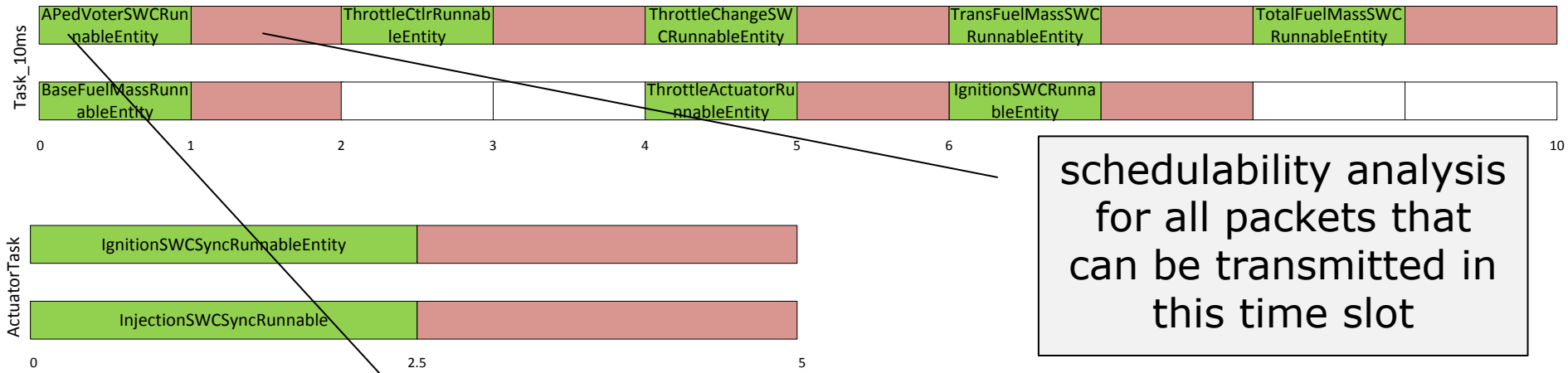


PowerDown	PowerUp	Cluster_1
• 2 idle cores • 10 idle links	• 3 idle cores • 16 idle links	• 0 idle cores • 10 idle links
5909.37 μJ per hyperperiod	3093.01 μJ per hyperperiod	9719.45 μJ per hyperperiod

Deadlines for runnables



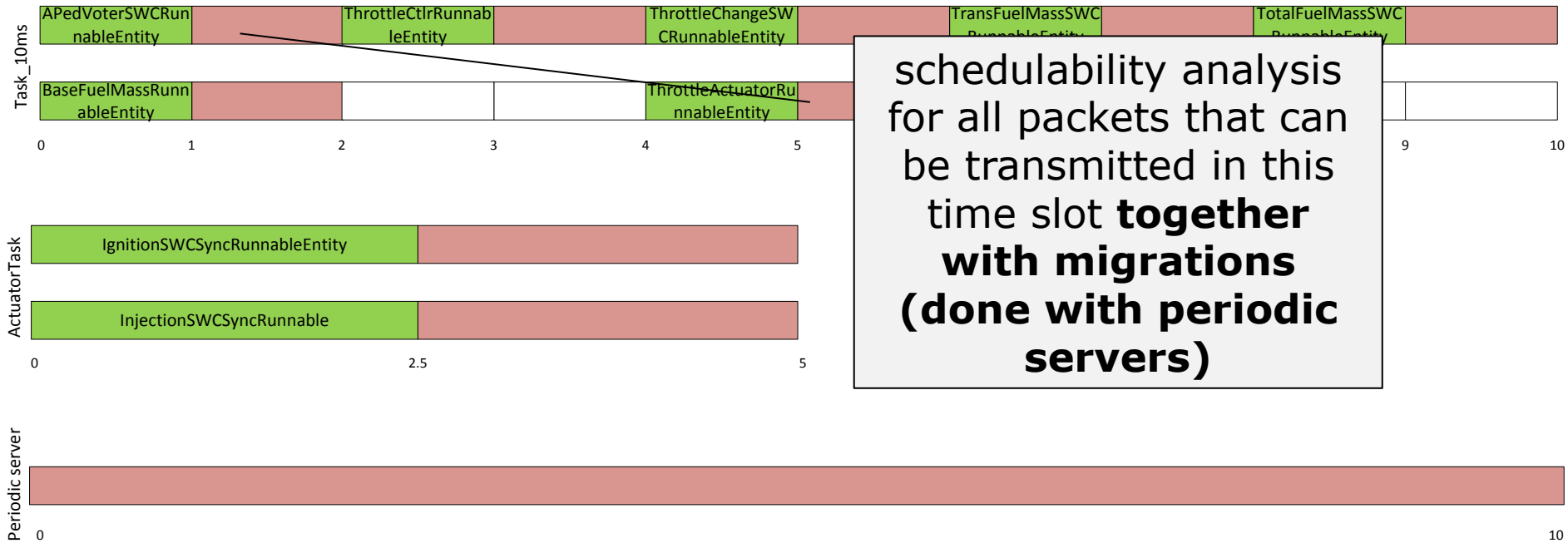
Schedulability analysis for taskset in any mode



schedulability analysis for all packets that can be transmitted in this time slot

schedulability analysis for all jobs that can be executed in this time slot

Schedulability analysis for taskset during mode changes

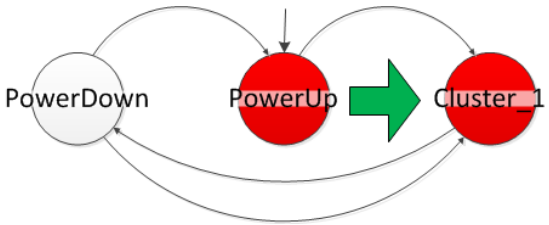


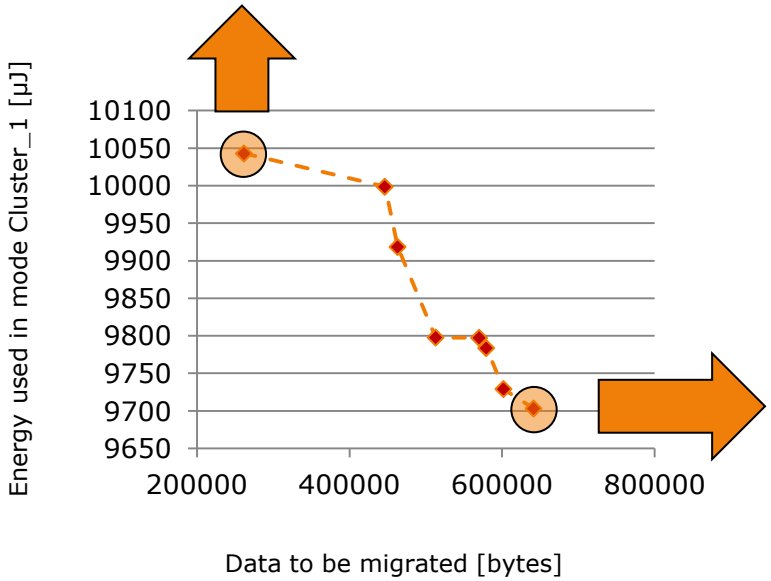
schedulability analysis for all packets that can be transmitted in this time slot **together with migrations (done with periodic servers)**

Network bandwidth selection

How many hyperperiods are required to migrate all the necessary data using periodic server?

Router latency [ns]	Link latency [ns]	#Hyperperiods (100ms)
100	200	1
100	400	1
200	500	1
400	800	2
500	1000	2





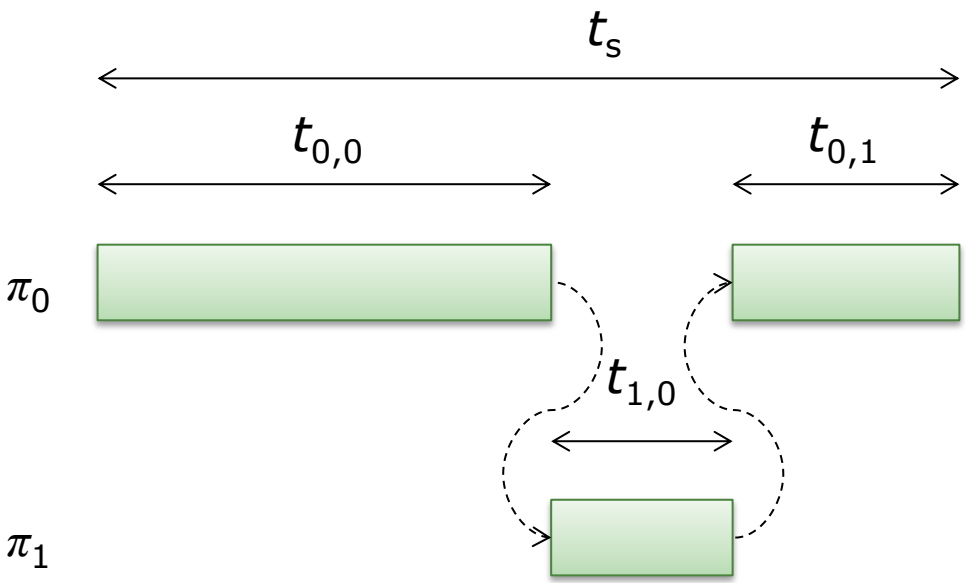
Router latency [ns]	Link latency [ns]	#Hyperperiods (100ms)
100	200	1
100	400	2
200	500	2
400	800	3
500	1000	3

Conclusion

- Static mapping of modes and dynamic migration at runtime on mode switch has been proposed to decrease the number of cores in ECUs.
- The technique benefits from the modal nature of the ECU.
- GA has been used to determine the allocation for all the ECU's modes, minimizing the energy dissipation and migration.
- NoC bandwidth for full schedulability has been determined.

Thank you!

Estimation of dissipated energy



$$E_0 = E_{0,\text{init}} + fd_0(t_{0,0}) + fd_0(t_{0,1}) + fs_0(t_s)$$

$$E_1 = E_{1,\text{init}} + fd_1(t_{1,0}) + fs_1(t_s)$$

Dynamic energy:

$$fd_0(t) = \text{const}_0 \cdot t$$

$$fd_1(t) = \text{const}_1 \cdot t$$

Static energy:

$$fs_0(t) = \text{const}_2 \cdot t$$

$$fs_1(t) = \text{const}_3 \cdot t$$