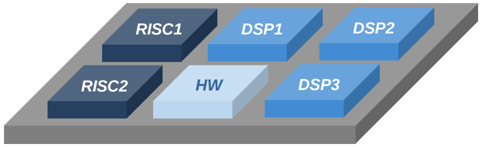
ASP-DAC 2019, January 24th

A Heuristic for Multi Objective Software Application Mappings on Heterogeneous MPSoCs

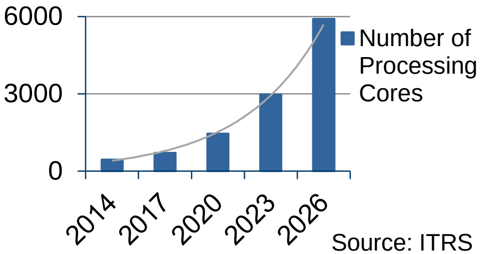
Gereon Onnebrink, Ahmed Hallawa, Rainer Leupers, Gerd Ascheid, Awaid-Ud-Din Shaheen

Introduction

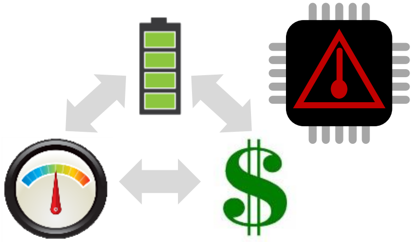
Current Platforms are Getting More Complex



MPSoC Platforms



Mobile Devices Became Part of our Daily Lives

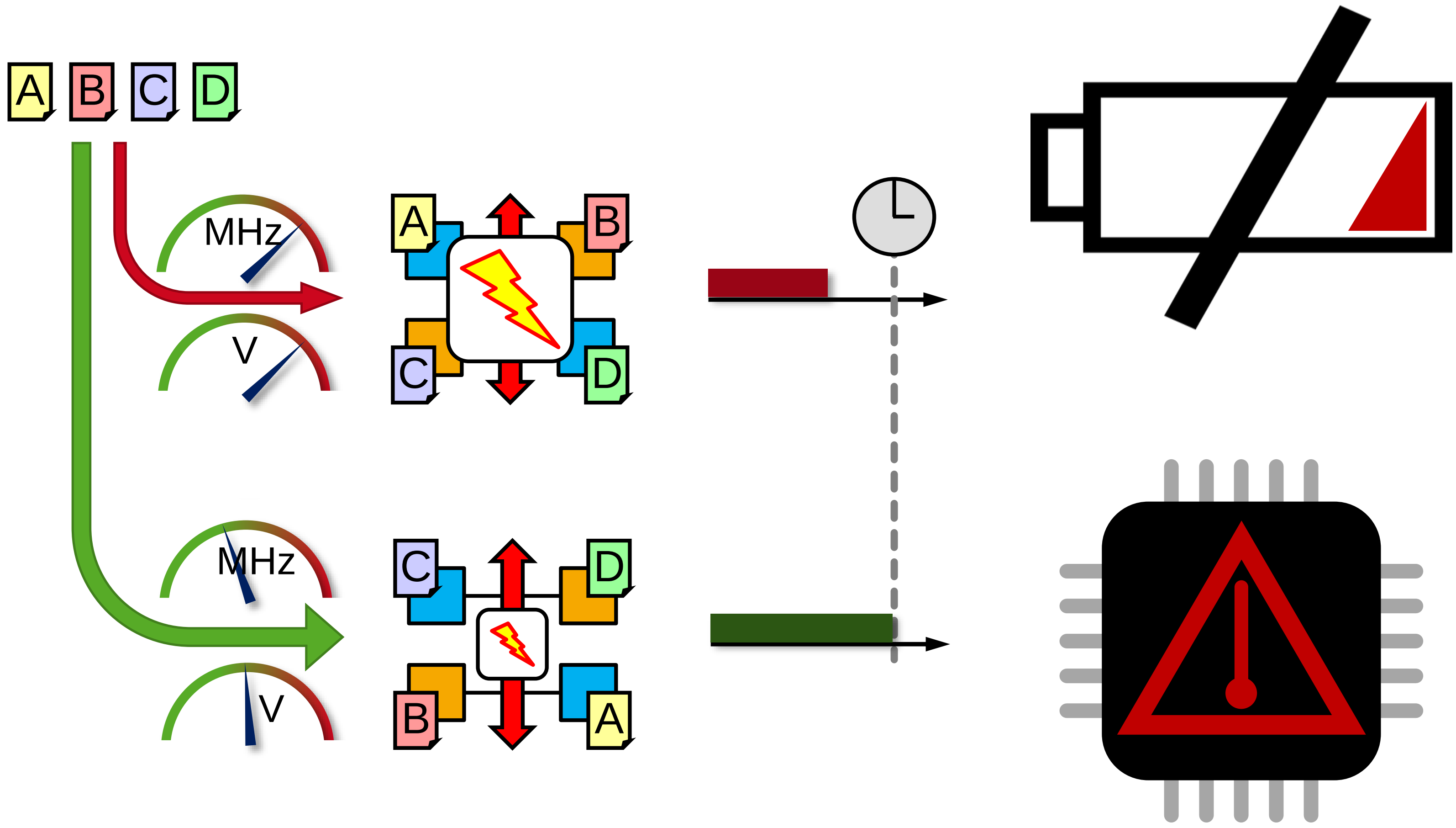


Facts

- ✓ The market imposes strict demands in terms of performance, energy and costs
- ✓ Complex applications require multicore processors to meet their demands

Parallel programming and optimization becomes more and more important

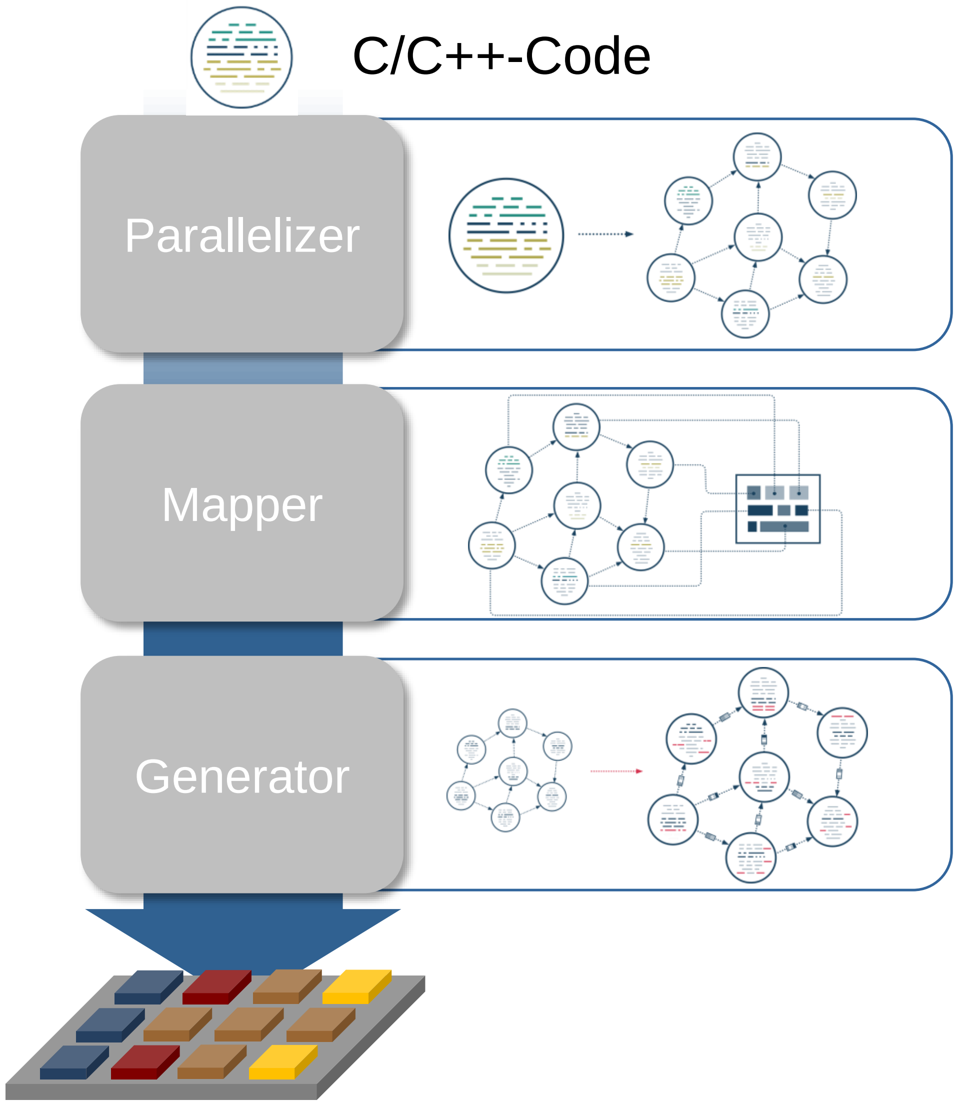
Motivation



Outline

- Motivation
- **MPSoC Compilation Tool Suite**
- Multi Objective Heuristic *TONPET*
- Case Studies
- Summary

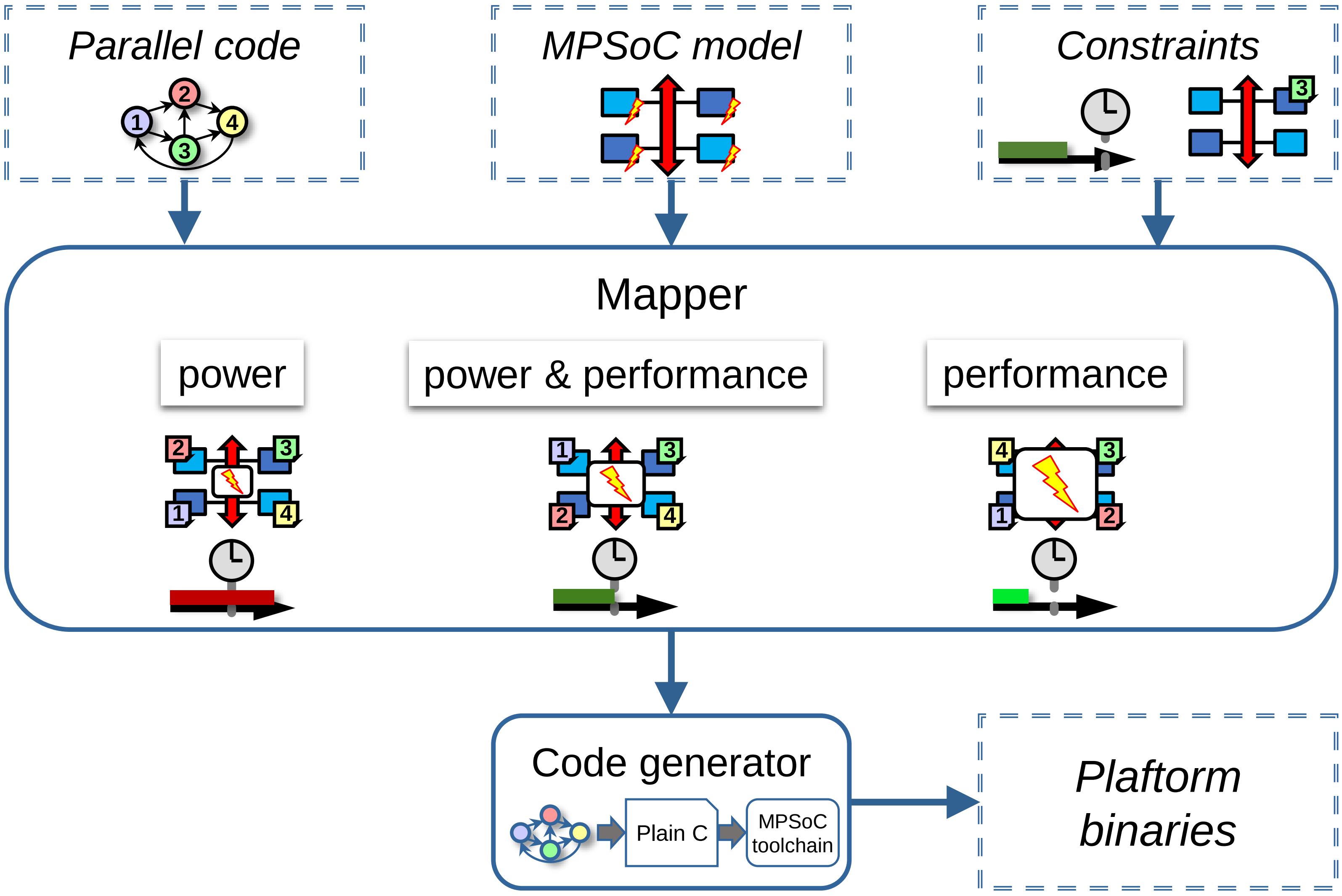
MPSoC Compilation with the Silexica Tool Suite



- ✓ Expose multiple forms of parallelism
- ✓ Define where and when execute tasks
- ✓ Generate code for the target platform

SILEXICA
multicore meets simplicity

Mapper



Parallel code
1
2
3
4
MPSoC model
Constraints
3
Mapper
power
power & performance
performance
Code generator
Plain C
MPSoC toolchain
Plaftorm binaries

Outline

Motivation

MPSoC Compilation Tool Suite

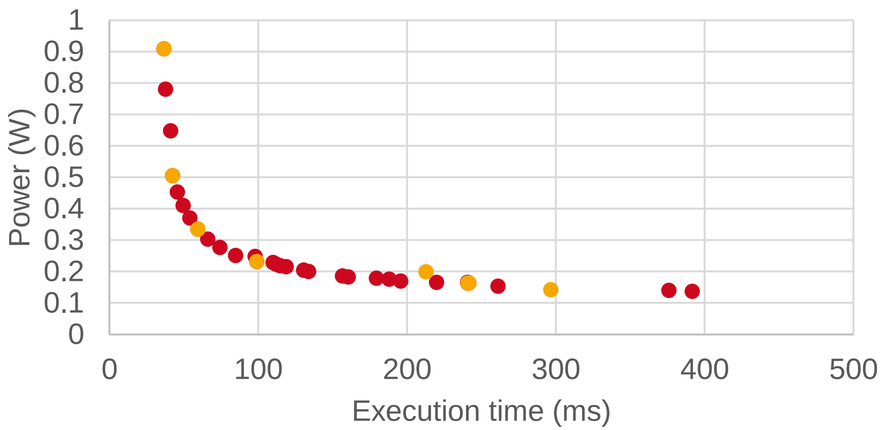
Multi Objective Heuristic *TONPET*

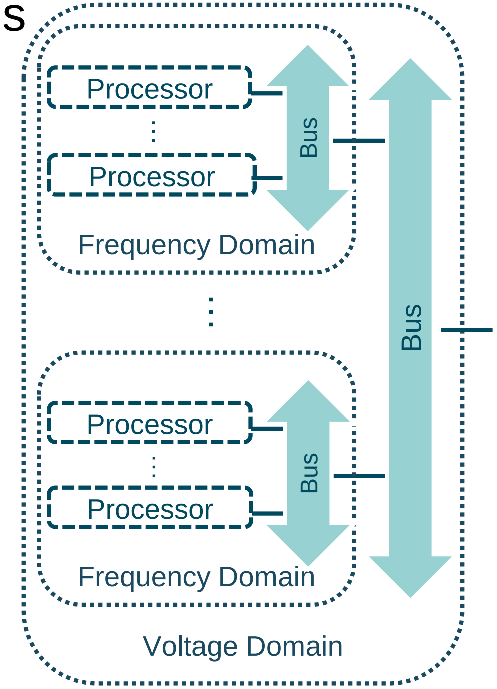
Case Studies

Summary

Multi Objective Heuristic *TONPET*

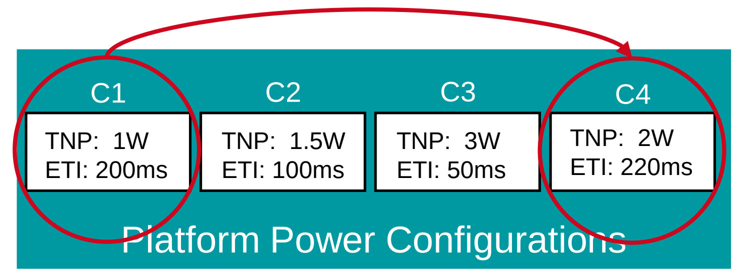
- Objectives
  - Performance, average power(, peak power, energy)
- Steps
  - Platform configuration classification
    - Classification for each combination of frequency settings
    - Mapping independent
  - Pruning of classified platform configurations
  - Pareto front calculation

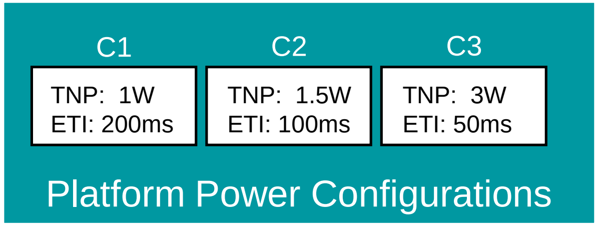


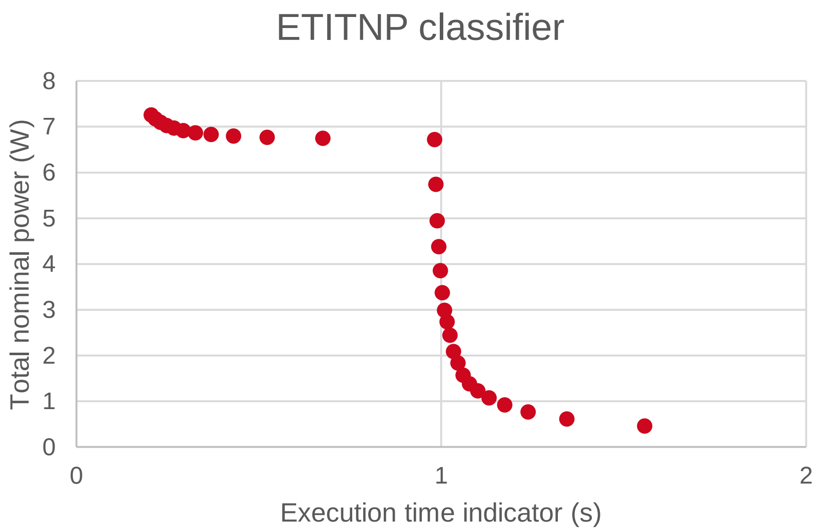


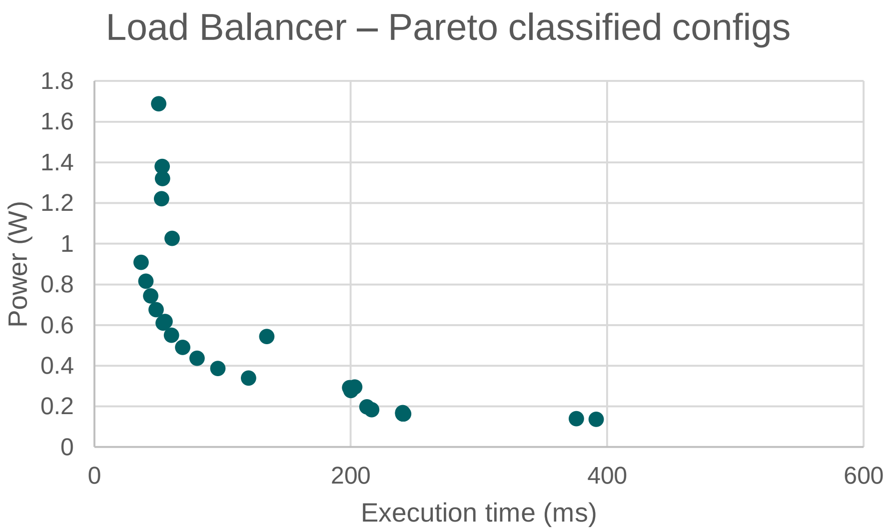
Platform Configuration Classification & Pruning

- Total Nominal Power (TNP):
  - Maximum power consumption
- Execution Time Indicator (ETI):
  - Sum of running all processes on all core types



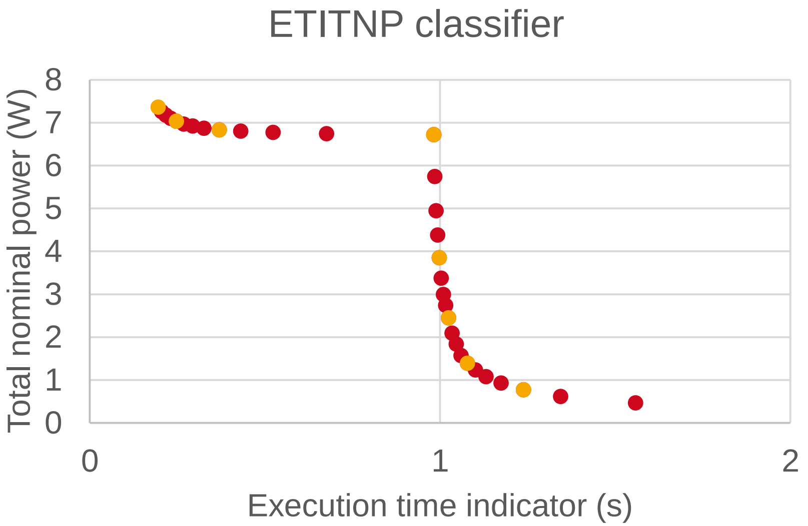


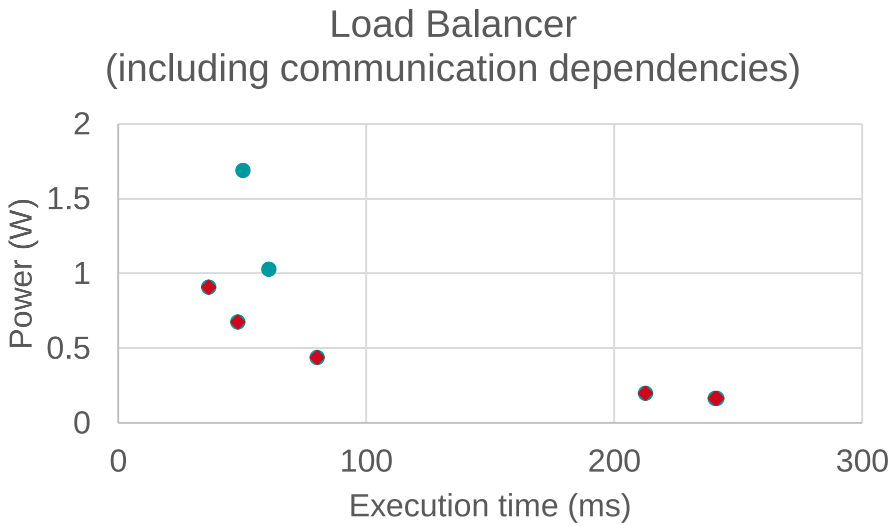




Pareto Front Calculation – 1/2

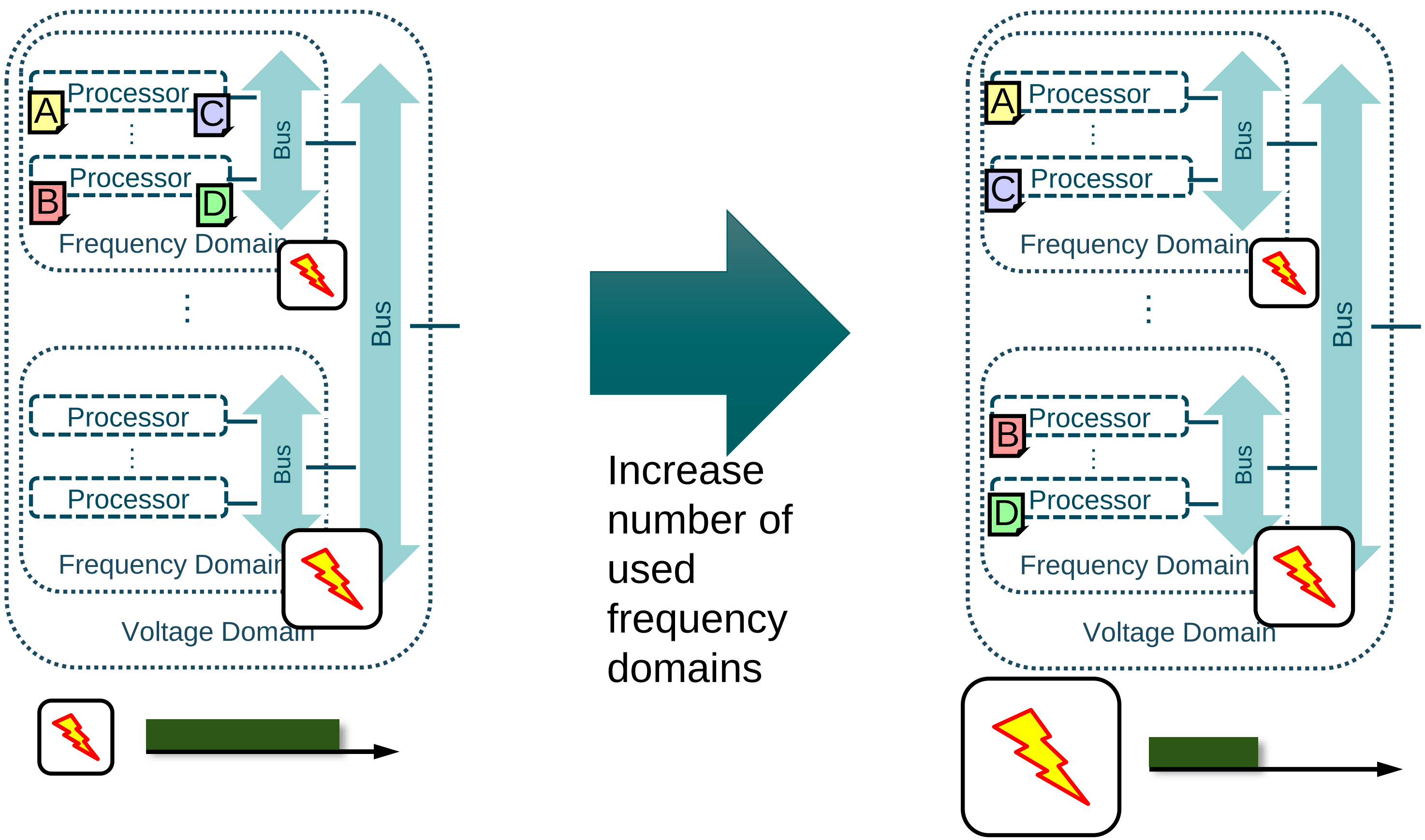
- Select every $\lfloor \log_2(|paretoClassConfigs|) \rfloor$ for further analysis
  - $\log_2$: trade-off between
    - fixed number of selected Pareto classified configurations
    - constant step size
- Keep $\text{pareto}(selectedConfigs)$





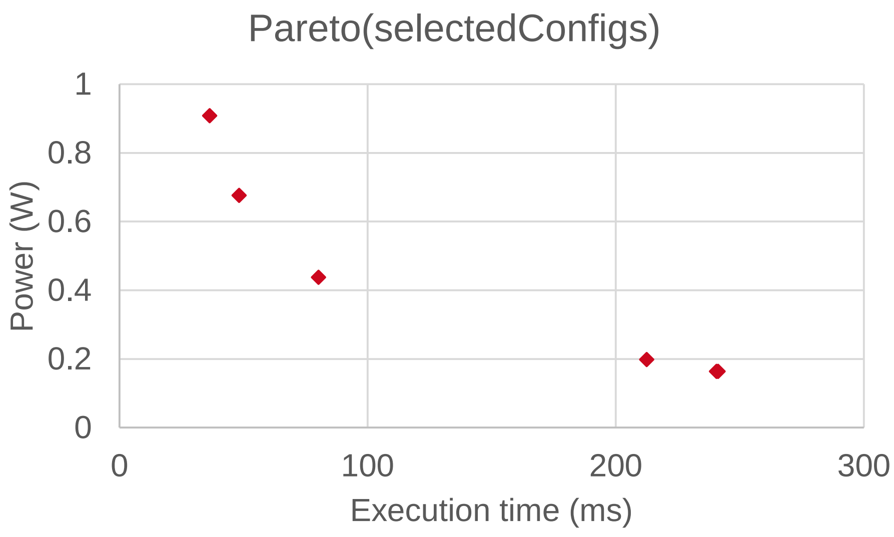
Process to Processor Mapping

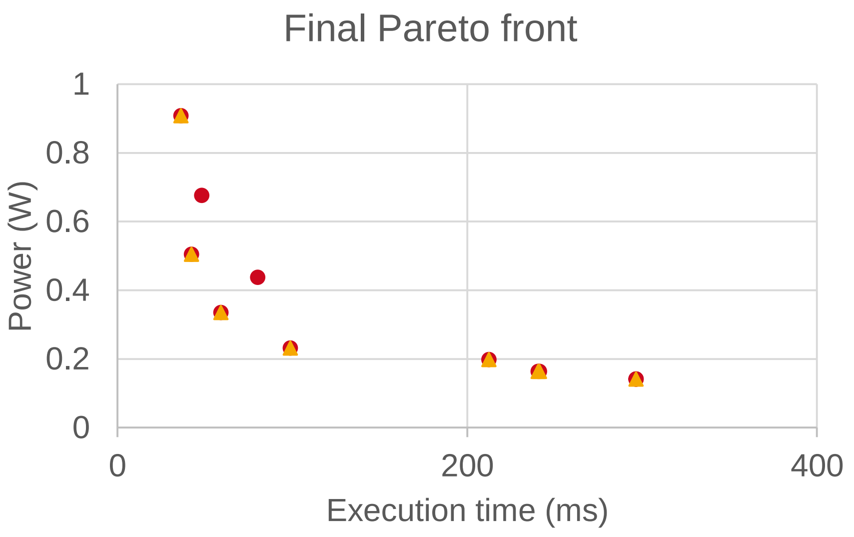
- Dependency on frequency domain
  - Covering full range of power and performance



Pareto Front Calculation – 2/2

- For *EvalConfig* in $\text{pareto}(selectedConfigs)$
  – Set *EvalConfig*
    - Allow 1, 2, 3,… frequency domains (ordered by power consumption)
    - Load balancing
    - Calculate power and execution time
    - Keep if Pareto optimal





Outline

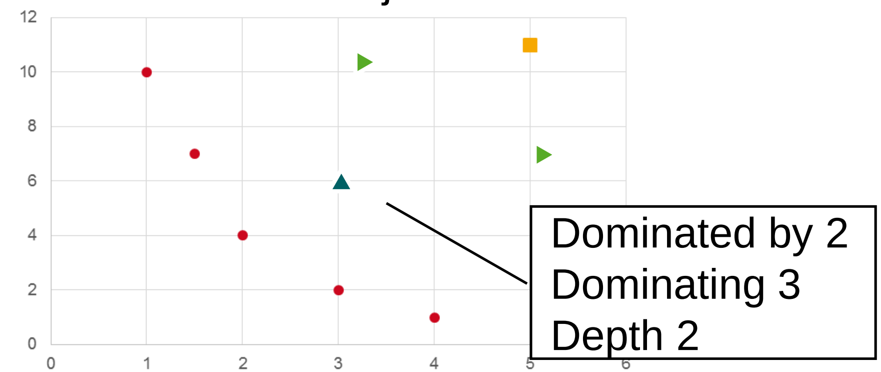
Case Studies

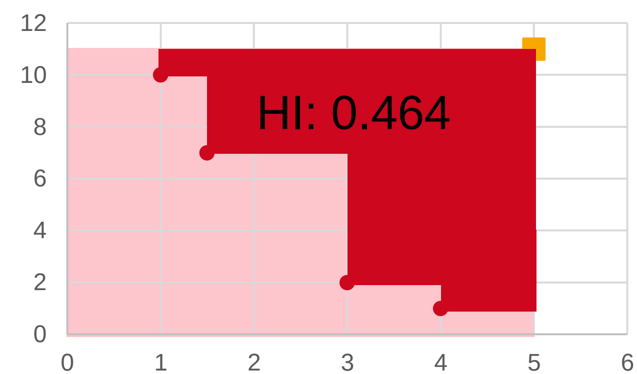
- ODROID-XU3 – Samsung Exynos-5422
  - 4+4 ARM Cortex A7 + A15, 247 frequency configurations
- Keystone II – Texas Instruments
  - 4+8 ARM Cortex A15 + DSP TI C66x, 26 frequency configurations
- Heterogeneous many-core virtual platform
  - 16+16 ARM Cortex A9 + AD Blackfin 609, $3.5 * 10^9$ frequency configurations
- Benchmarks (written as Kahn Process Network):
  - Audio filter (11), JPEG (24), LTE (19), Mandelbrot with 16 and 150 workers (18/152), Sobel filter (5), MIMO OFDM (36), STAP (16)
- Evaluation
  - Speed-up compared to R2 indicator EMOA
  - Quality of Pareto front compared to R2 indicator EMOA

Evolutionary Multi Objective Algorithm (EMOA)

- Evolutionary Algorithms
  - Inspired by biological evolution (black box optimization)
  - Population based
  - Genotype to phenotype mapping
- Taxonomy
  - Single objective
  - Multi objective
    - Dominance based
      - NSGA-2 (up to 2 objectives)
    - Indicator based
      - Hypervolume indicator (HI) (slow for more than 2 objectives)
      - R2 indicator (faster)

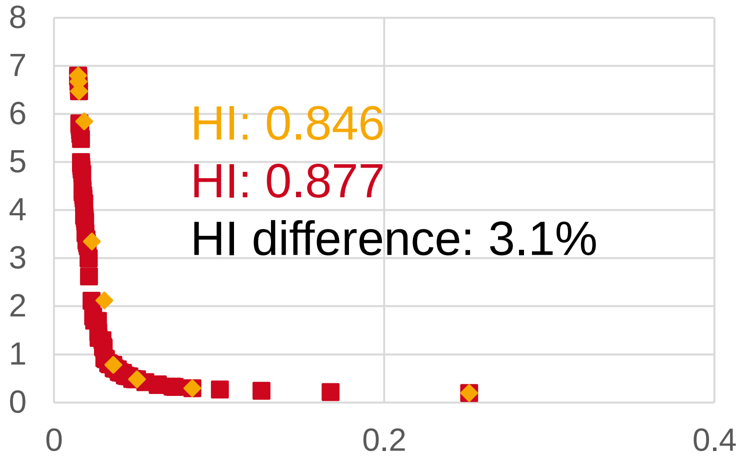
Minimize 2 objectives

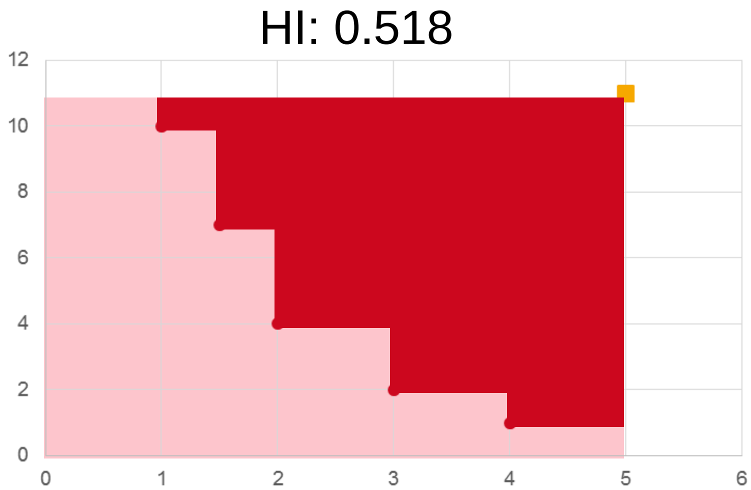


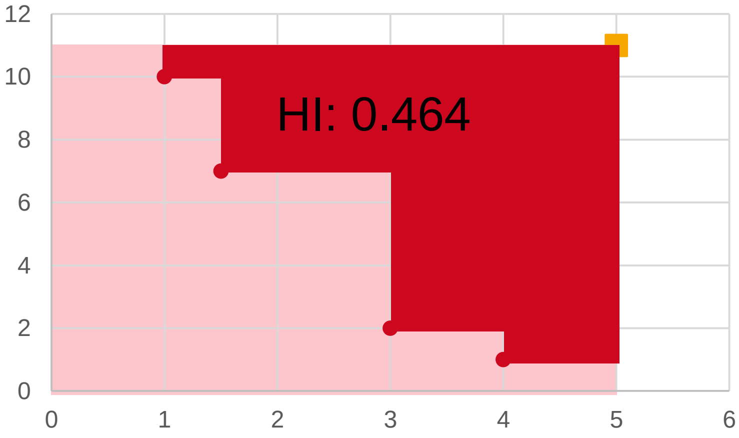


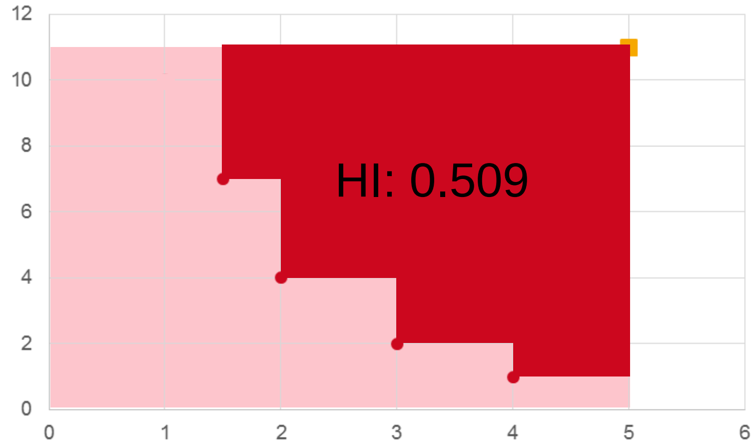
Comparing Pareto Solutions

- Hypervolume Indicator (HI)

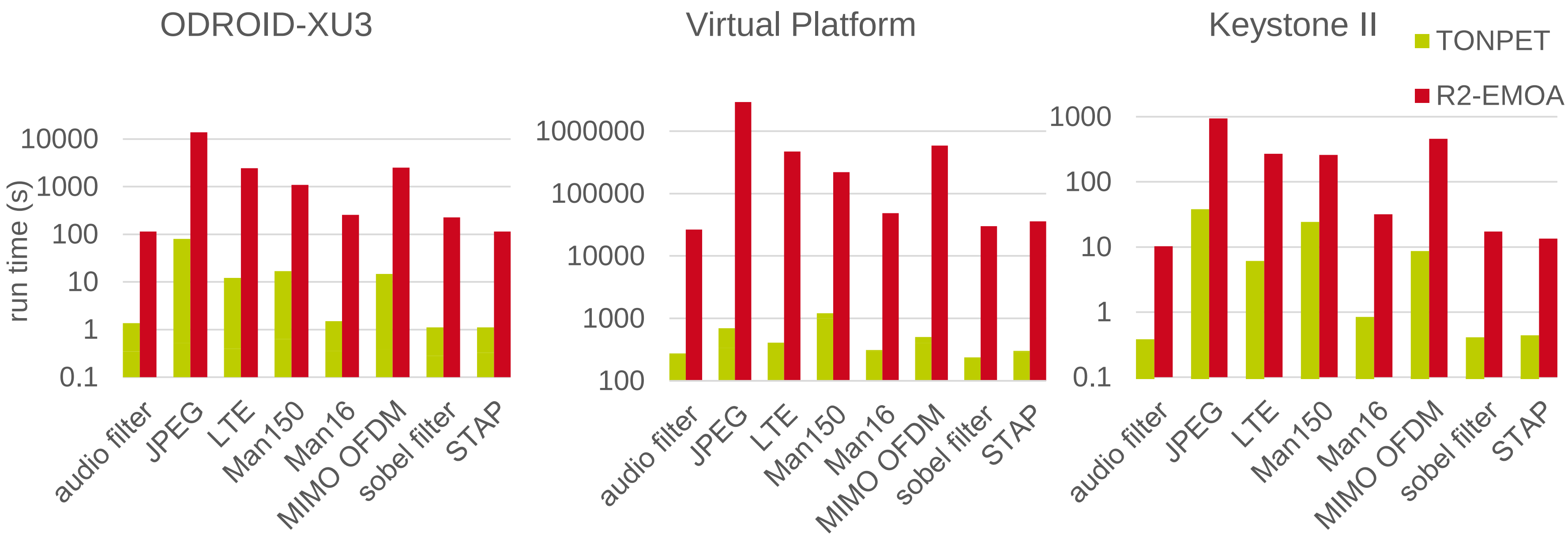








Run time compared to constraint R2-EMOA



- ODROID-XU3:
  - Speed-up 80x (worst case), 120x (average)
- Keystone II:
  - Speed-up 18x (worst case), 30x (average)
- Heterogeneous many-core virtual platform
  - Speed-up 88x (worst case), 150x (average)

Quality of Pareto front compared to R2-EMOA

- TONPET HI performance relative to R2-EMOA
  – Better than budget constrained EMOA: “+”
  – Better than budget unconstrained EMOA: “++”

	ODROID-XU3	Keystone II
audio filter	-0.4%	+
JPEG	+	+
LTE	-0.3%	-4.6%
Man150	++	++
Man16	+	++
MIMO OFDM	+	+
sobel filter	-1%	+
STAP	-0.9%	-6.3%

Outline

- Motivation
- MPSoC Compilation Tool Suite
- Multi Objective Heuristic *TONPET*
- Case Studies
- Summary

Summary

- Multi objective heuristic *TONPET*
  - Pareto optimum w.r.t. two objectives: power and performance
  - Classification and pruning to reduce search space
    - Applicable to many-core platforms

- Evaluation with R2-EMOA
  - Worst case speed-up
    - 18x (Keystone II), 80x (ODROID-XU3), 88x (Virtual Platform)
  - HI performance
    - 4.7% better than constraint R2-EMOA (Keystone II and ODROID-XU3)
    - 3% less than constraint R2-EMOA (Virtual Platform)

Thank you for your attention!