IETR Seminar “Energy Efficiency and Sustainable Electronics”
Tuesday 13th February 2024
Venue: CentraleSupélec, Rennes campus + online

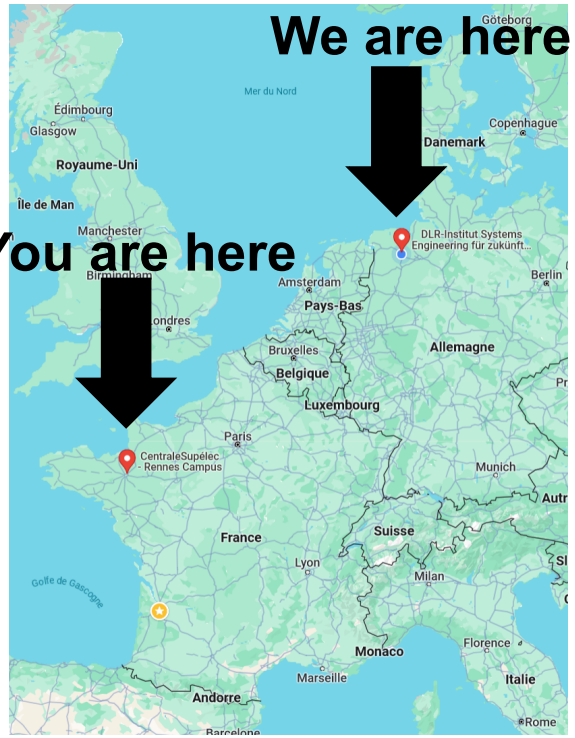
Energy Optimization of Neural Networks on Edge Multi-Core Platforms

Quentin Dariol[1], Sébastien Le Nours[2], Ralf Stemmer[1], Domenik Helms[1], Kim Grüttner[1], Sébastien Pillement[2]

1: German Aerospace Center (DLR) – Institute Systems Engineering for future mobility (SE), Oldenburg
2: IETR Nantes, ASIC team

IETR
Nantes Université
DLR

Deutsches Zentrum für Luft und Raumfahrt (DLR e. V.)
Institute Systems Engineering for future mobility (SE)



Opportunities:
Mobility for a sustainable society/
New urban concept
New business models and opportunities
Safe, efficient and reliable transportation

Threats:
Uncontrollability
Data misuse
Lack of security
Mobility divide
Being at the mercy

Requirements:
Transparency
justice and fairness
freedom from error
responsibility
privacy
reliability

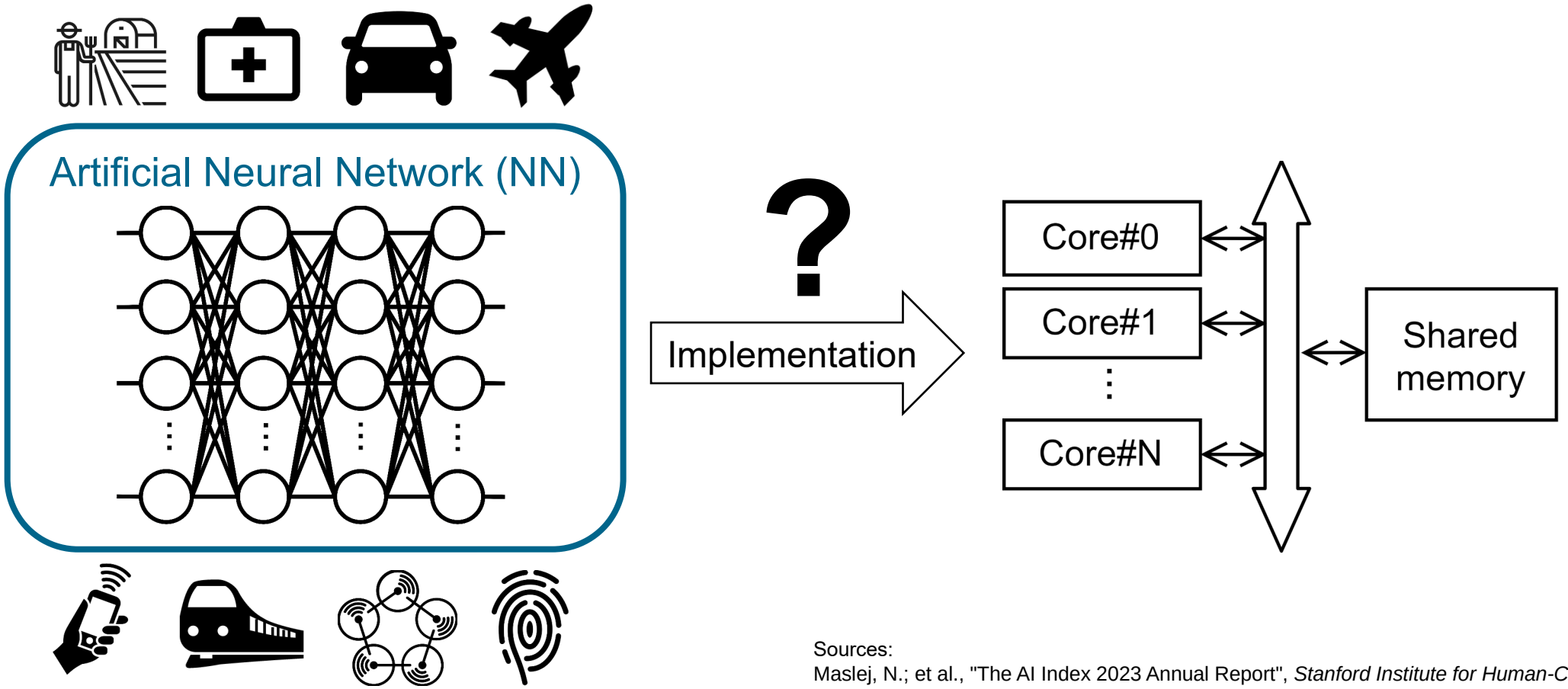
enables

Sensors, networking and artificial intelligence as the basis for new automated and autonomous systems

=> Embedded AI group: projects proposing models and architectures optimised for embedded AI.

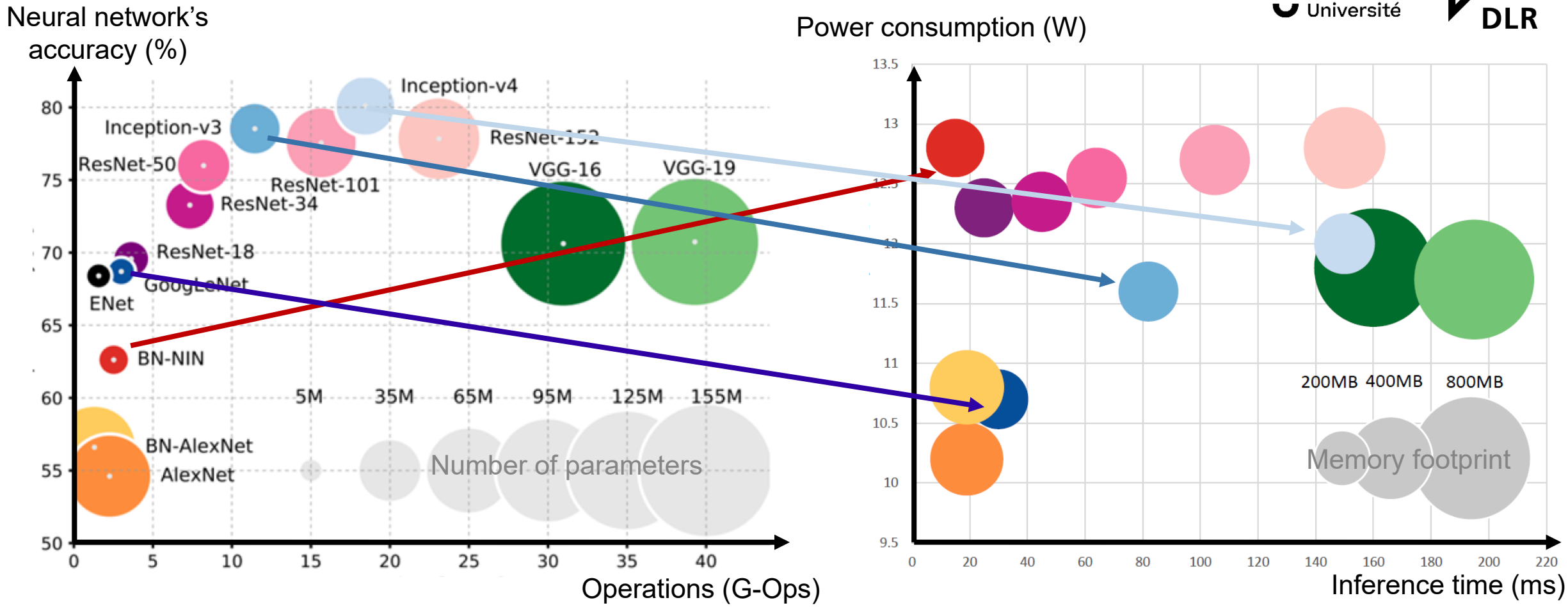
Context – Artificial Neural Networks (NNs)

- Raise of interest for AI algorithms and especially for NNs.



Sources:
Maslej, N.; et al., "The AI Index 2023 Annual Report", *Stanford Institute for Human-Centered Artificial Intelligence (HAI),* 2022
Open licenced symbols from: https://www.opensymbols.org/

Context – NNs on edge devices

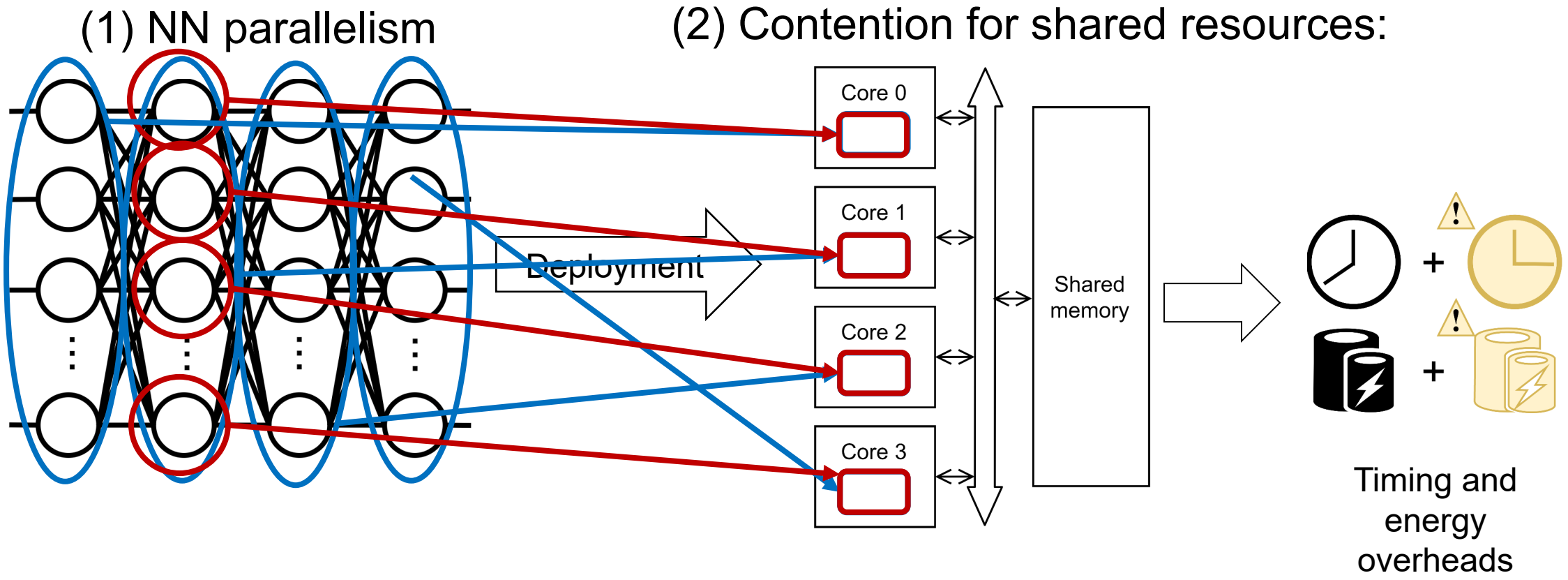


=> Metrics that matter at the edge

=> Need evaluation flow to find optimized mappings

Source: Alfredo Canziani, Adam Paszke, and Eugenio Culurciello. *An Analysis of Deep Neural Network Models for Practical Applications*. 2017. arXiv: 1605.07678.

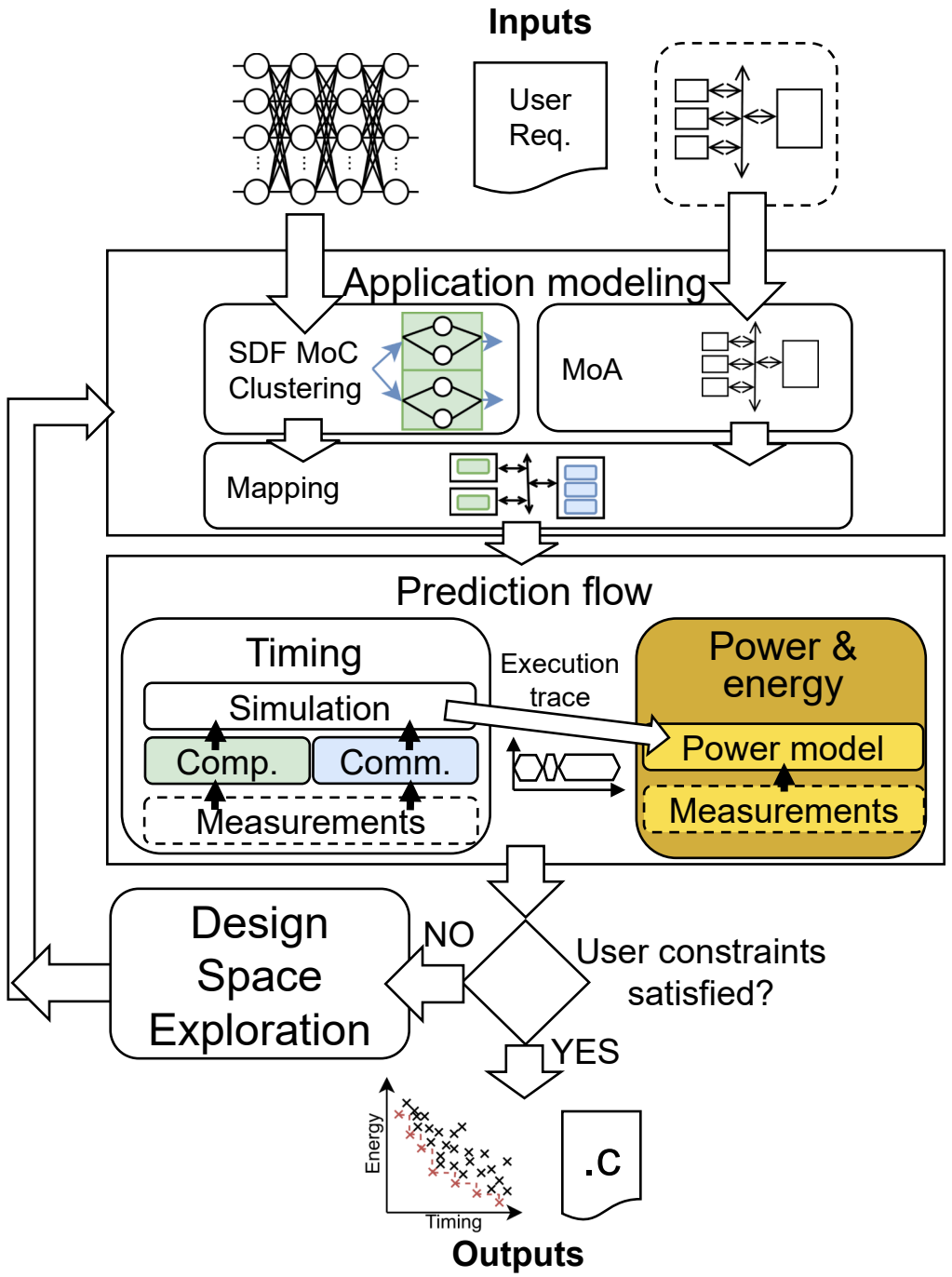
Challenges of NN deployment on multi-core platforms



Other aspects:

- Use of power management
- Platform size (number of cores, memory)
- NN different workloads => no « one fits all » solution

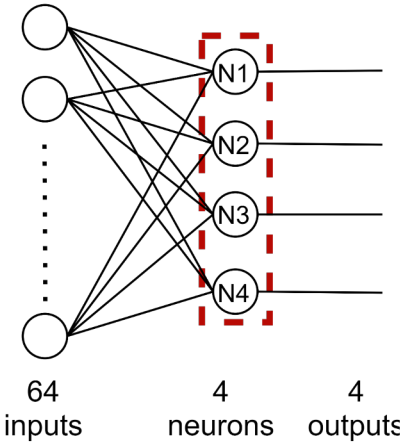
Proposed flow



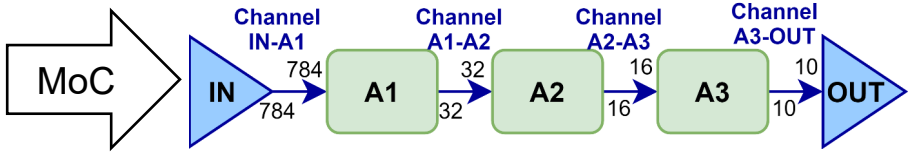
Ref:

- [1] Dariol2022 « A Hybrid Performance Prediction Approach for Fully-Connected Artificial Neural Networks on Multi-core Platforms » - SAMOS 2022
- [2] Dariol2023 « Fast Yet Accurate Timing and Power Prediction of Artificial Neural Networks Deployed on Clock-Gated Multi-Core Platforms » - RAPIDO 2023
- [3] Dariol2023b « Early Timing and Energy Prediction and Optimization of Artificial Neural Networks on Multi-Core Platforms » - PHD Thesis - Nantes Université Nov. 2023
- Open source Git repository: https://gitlab.univ-nantes.fr/lenours-s/pssim4ai-open

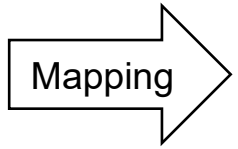
Model of Computation (MoC), Model of Architecture (MoA), mapping

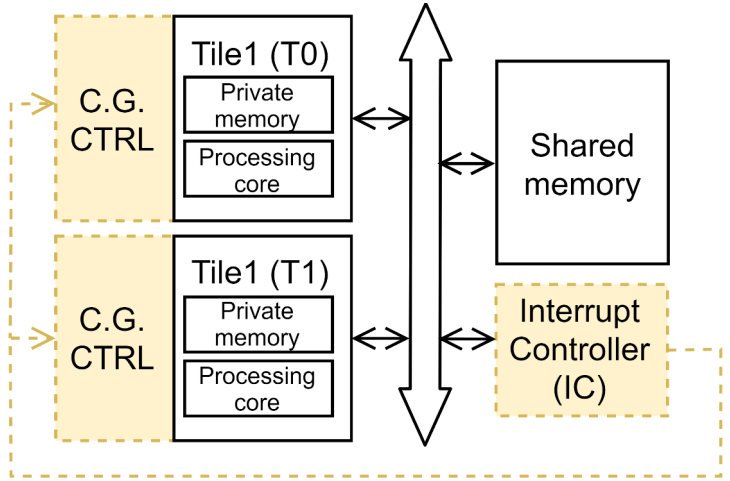


Artificial Neural Network (NN)



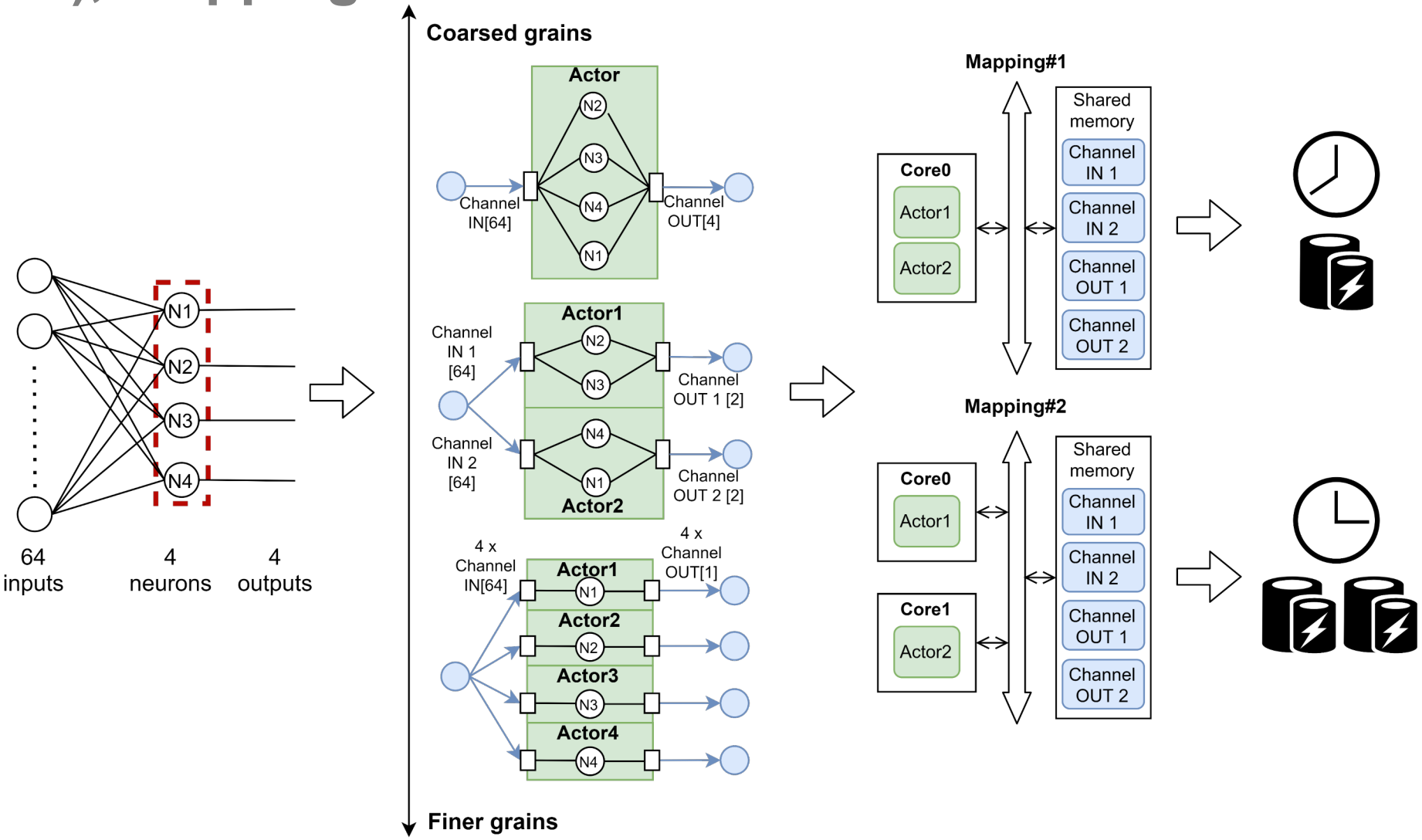
- SDF: Synchronous DataFlow
- Strict separation computation/communication
  - Actors,
  - Channels,
  - Tokens.





- MoA: Model of Architecture
- Two versions:
  - Without power management: polling
  - With power management: interrupt + clock gating

Model of Computation (MoC), Model of Architecture (MoA), mapping



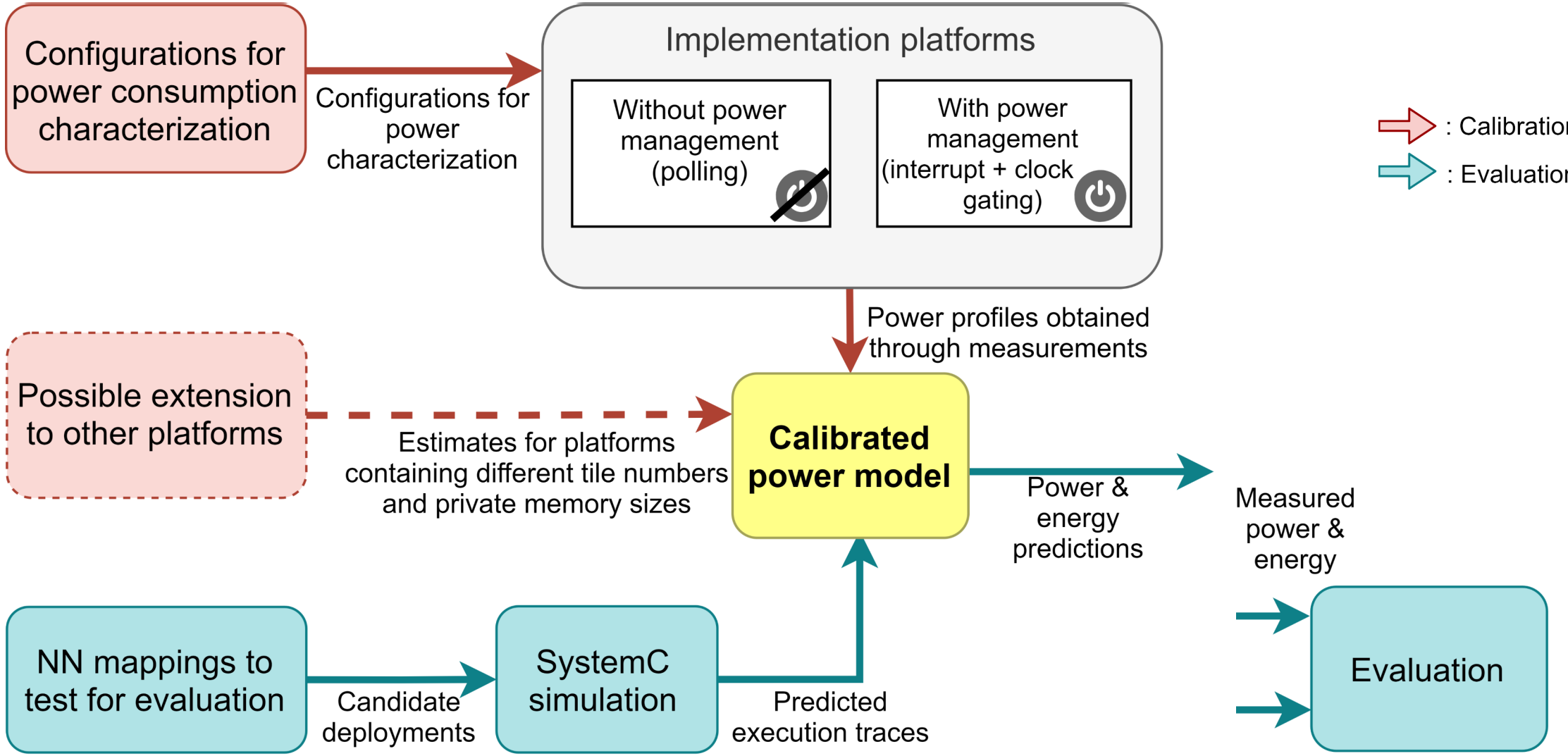
Artificial Neural Network (NN)

Clustering (description using a dataflow-oriented MoC)

Mapping

Performance/power analysis

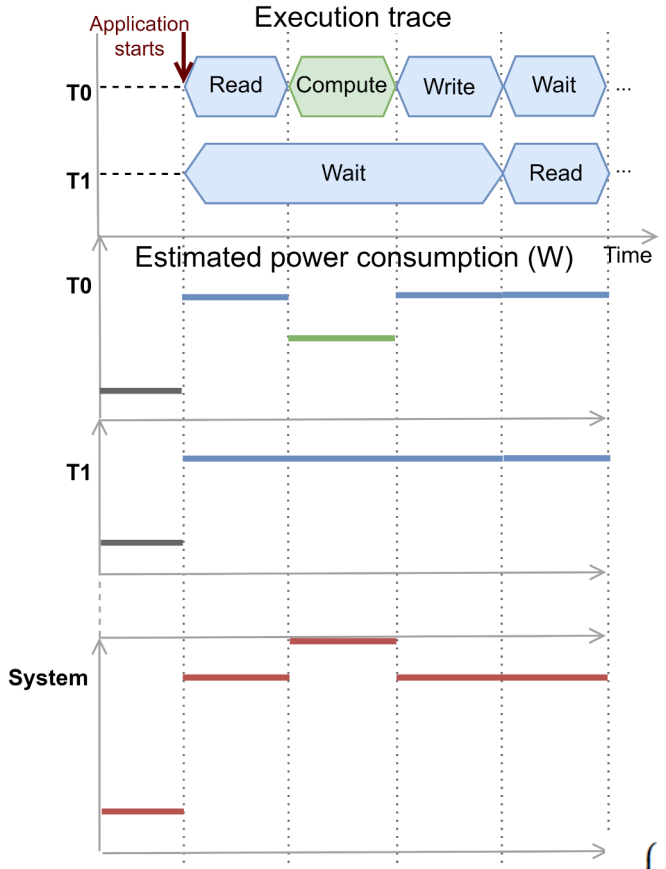
Power modeling flow - Overview



Power modeling flow – Proposed model

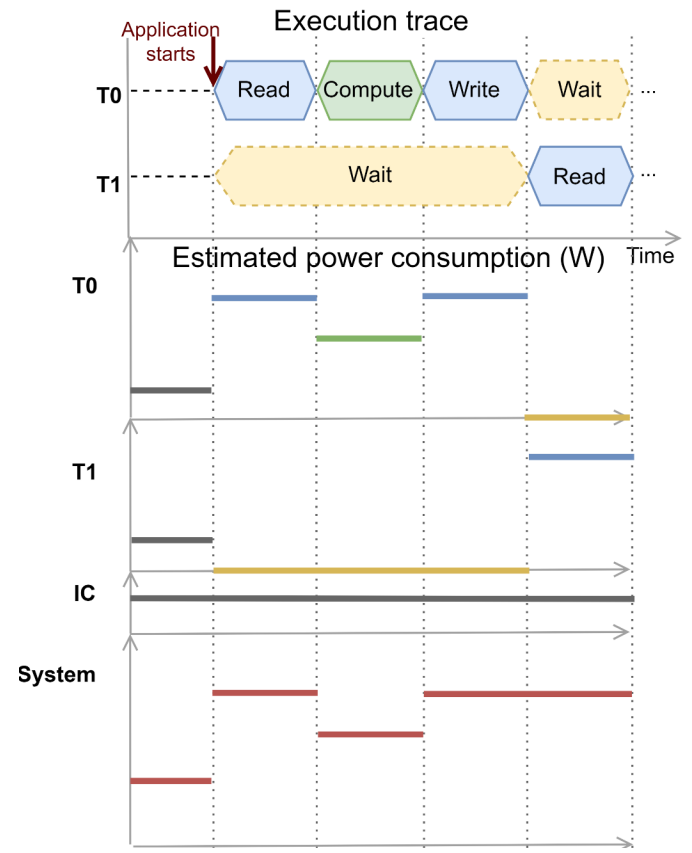
$$P(t) = P_{\text{static}} + P_{\text{comp}}(t) + P_{\text{comm}}(t)$$

Without power management △



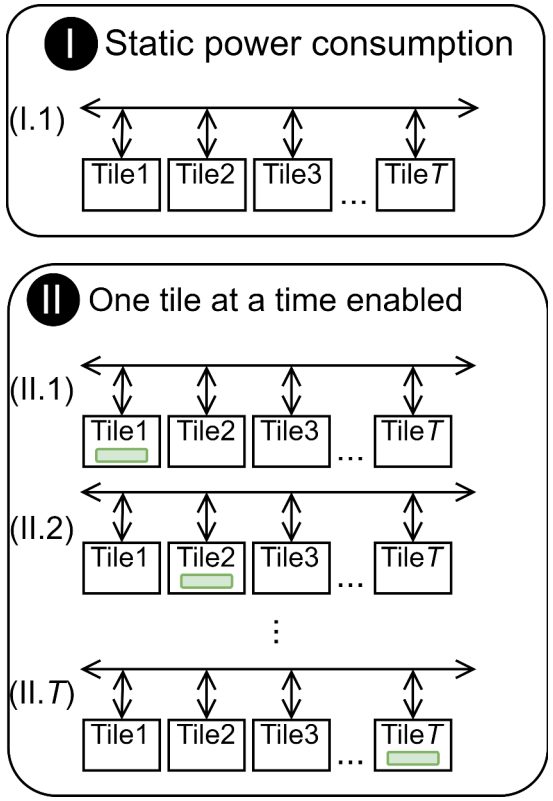
$$P_{\triangle,\,\text{comm}}(t) = P_{\triangle,\,\text{rwp}}(t) = \begin{cases} P_{\text{sm}} & \text{if at least one tile is reading, writing or polling on shared memory at time } t \\ 0 & \text{otherwise} \end{cases}$$

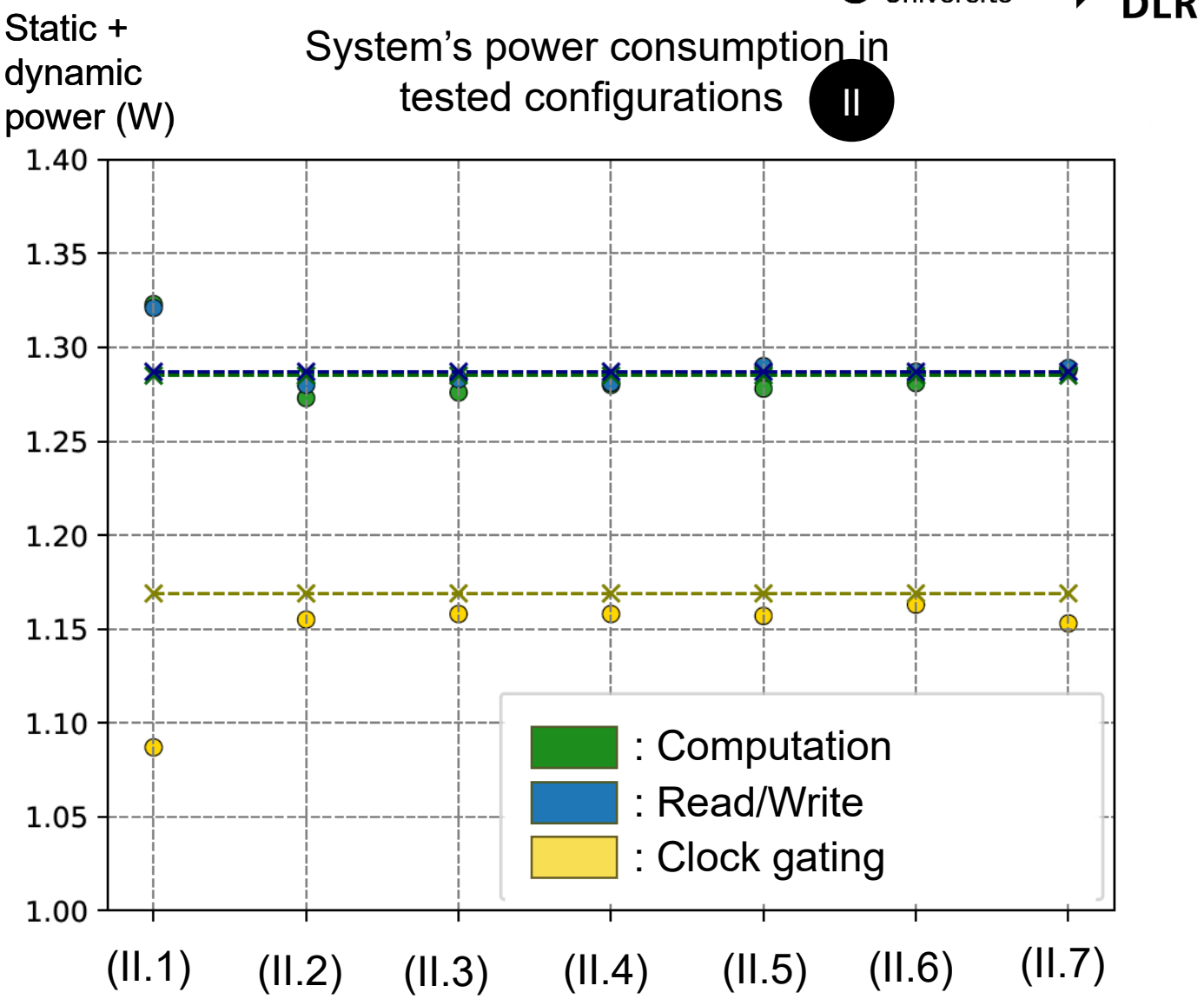
With power management ▲



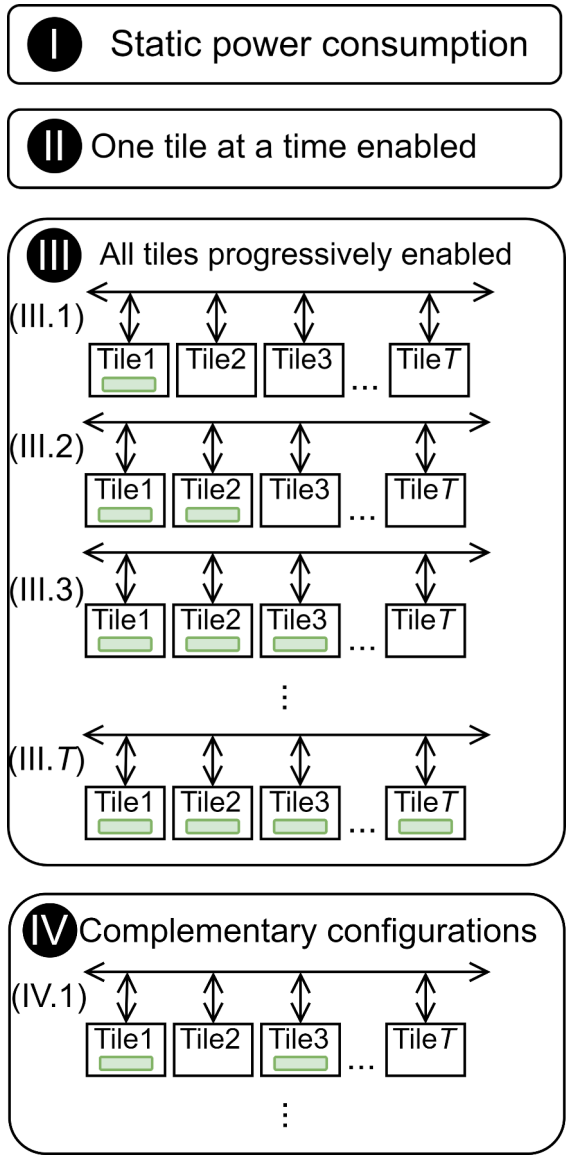
$$P_{\blacktriangle,\,\text{comm}}(t) = P_{\blacktriangle,\,\text{rw}}(t) + P_{\blacktriangle,\,\text{cg}}(t)$$

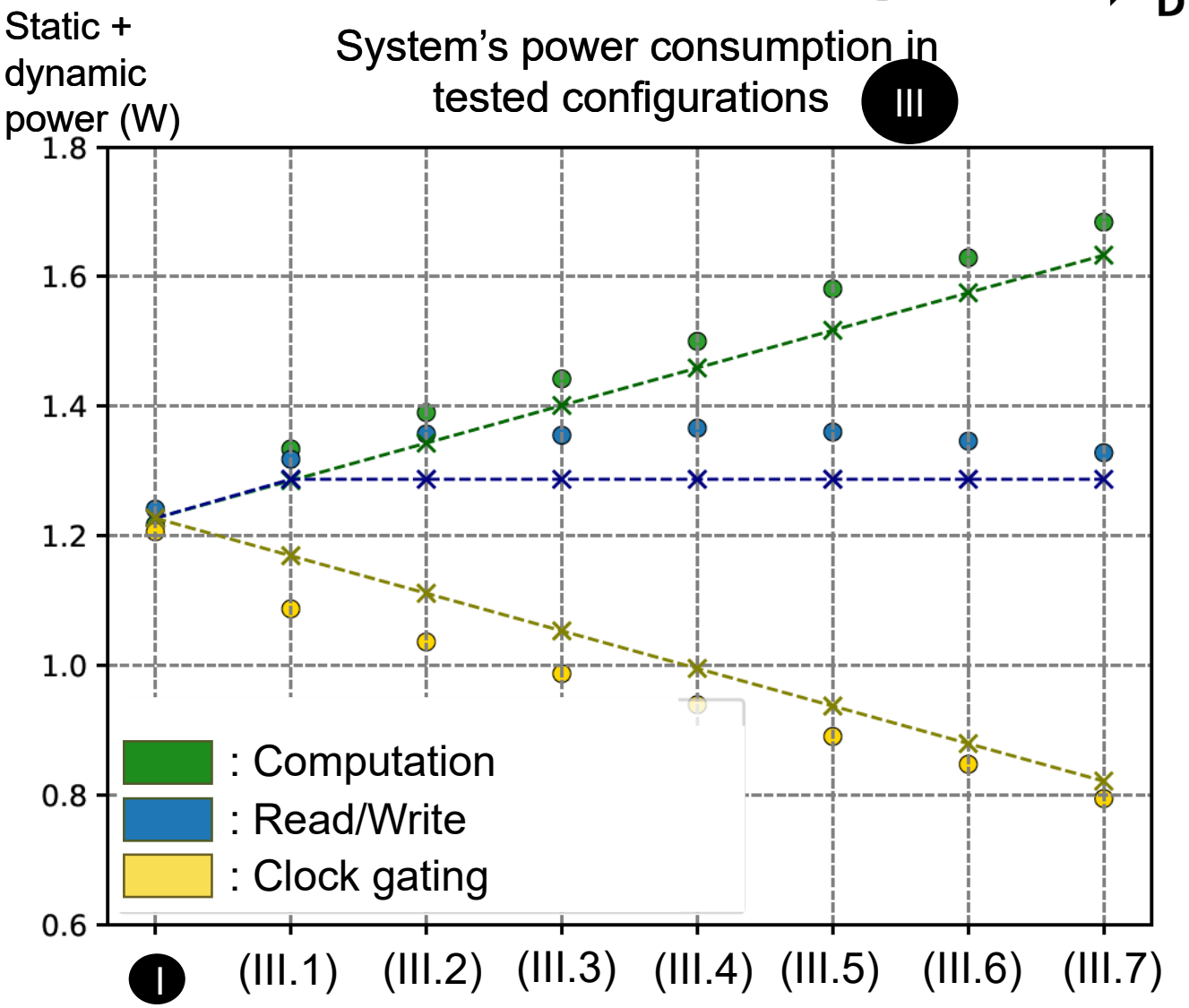
Power modeling flow – Power model calibration



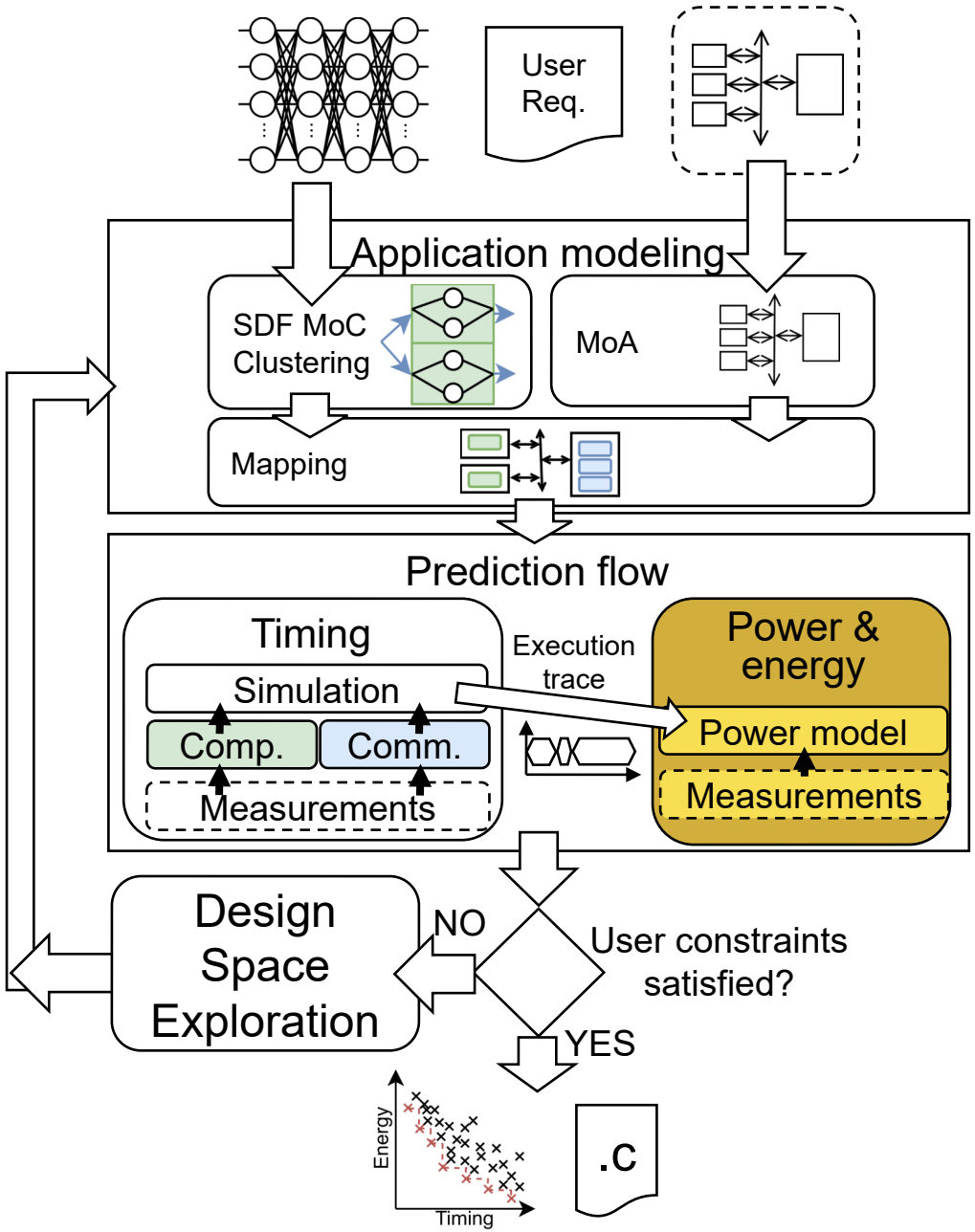


Power modeling flow – Power model calibration

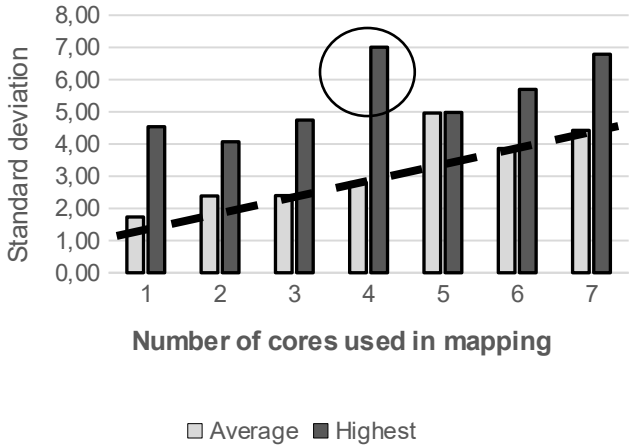


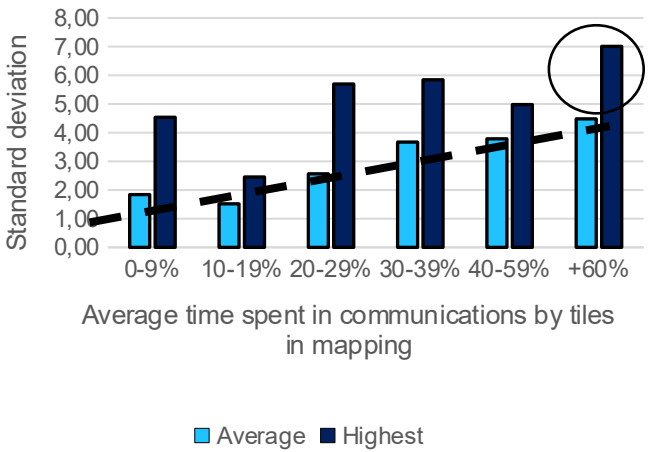


Power modeling flow – Experiments



- 1 - Overall accuracy: >93% on 54 mappings.
- 2 - Evaluation speed: ~20s.
- 3 - NN different workloads: average prediction error between 1,8% and 3% for the 4 NNs ✔
- 4 - Use of power management: average is 2,11% without, 3,92% with ✔
- 5 - Number of cores used and communication rates ✔
- 6 - Analytical model:
  - Maximum error: ~20%
  - Evaluation time: ~1ms
- Rapid prototyping: highest accuracy but 40s

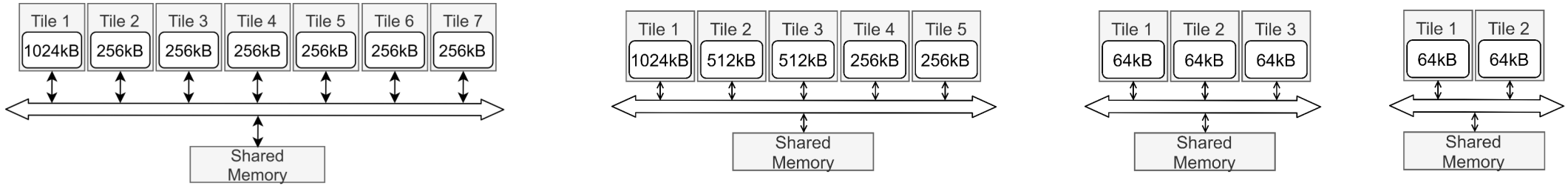




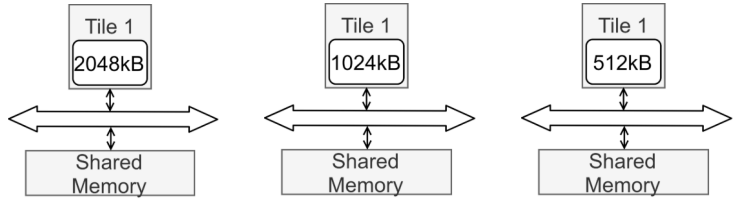
Power modeling flow – Experiments

- Use to jointly evaluate and optimize multi-core platform architectures and NN deployments under power and energy constraints

Multi-core platform versions:



Single-core platform versions:



Static power consumption only		Static + dynamic	
Multi-core	Single-core	Multi-core	Single-core
< 5%	< 5%	~ 5%	**> 10%**

Power modeling flow – Illustration of use for DSE

- Evaluation time per mapping: 20s
- Illustration of the use of the modeling flow in a DSE setup.
  - 84 mappings evaluated in 28 minutes (~20s per mapping)

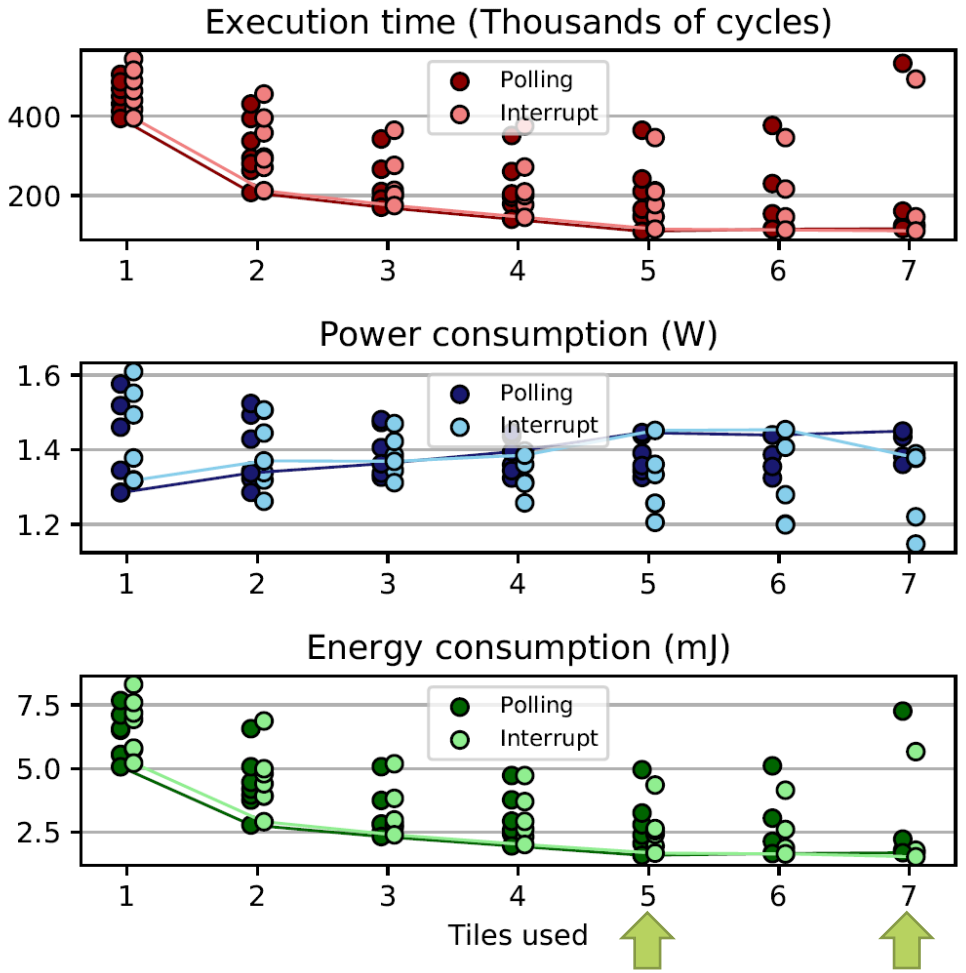
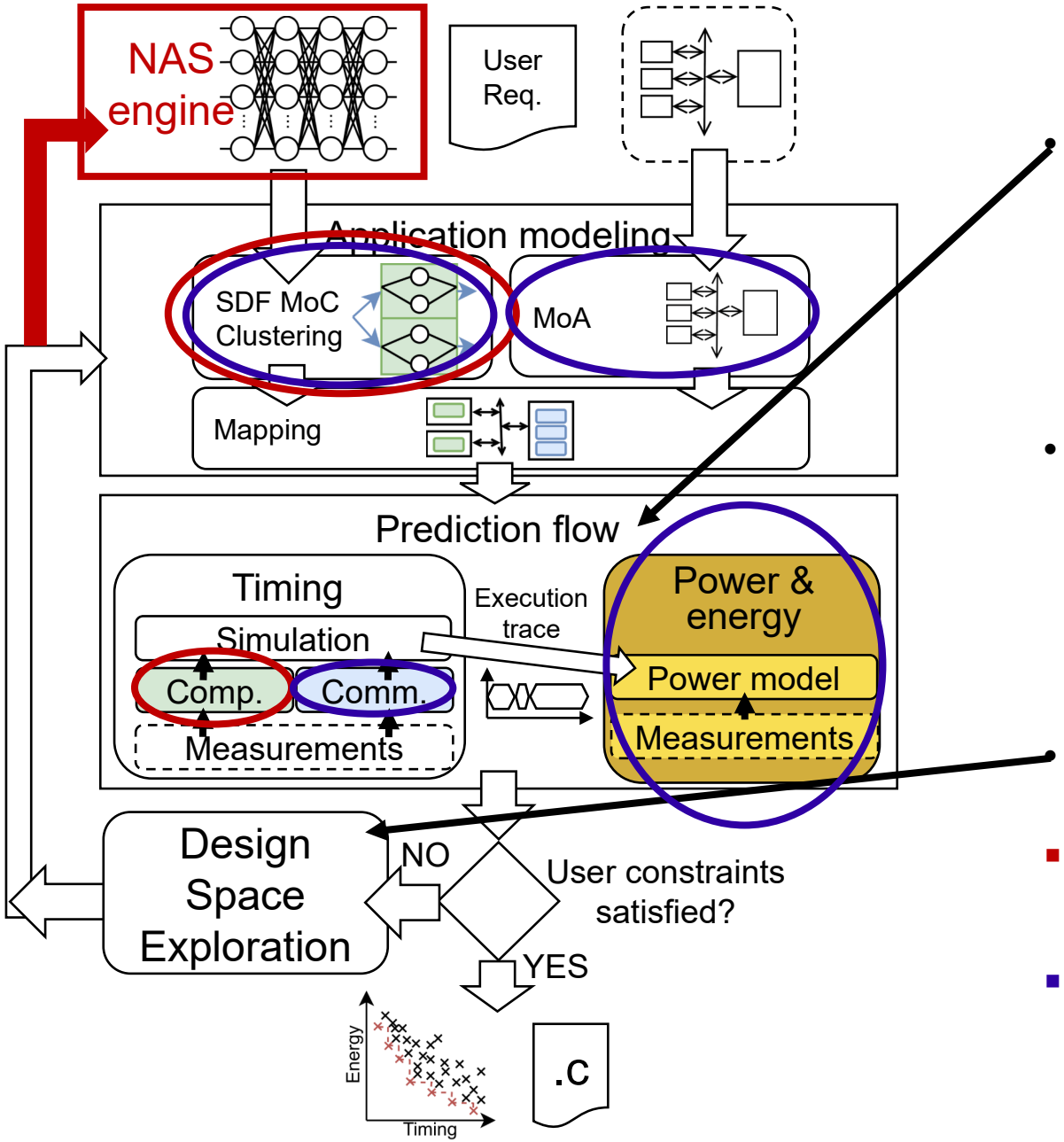


Figure – Design Space Exploration under timing, power and energy constraints

Conclusion



- Flow to provide **fast** yet **accurate evaluation** early in design phases of **timing** and **energy** properties for NNs implementations on multi-core platforms.
  - Hybrid: simulation, analytical models, measurements.
- More relevant than rapid prototyping / analytical models
  - 6 times faster than rapid prototyping with high accuracy + doesn't need the NN to be trained.
  - Analytical models not confident with too complex contention problems.
- Illustration of the use in a DSE setup.

- Extend the flow to support Neural Architecture Search (NAS)
- Offer modeling and exploration of external memory accesses (necessary for larger NNs)

IETR Seminar “Energy Efficiency and Sustainable Electronics”
Tuesday 13th February 2024
Venue: CentraleSupélec, Rennes campus + online

Energy Optimization of Neural Networks on Edge Multi-Core Platforms - QUESTIONS

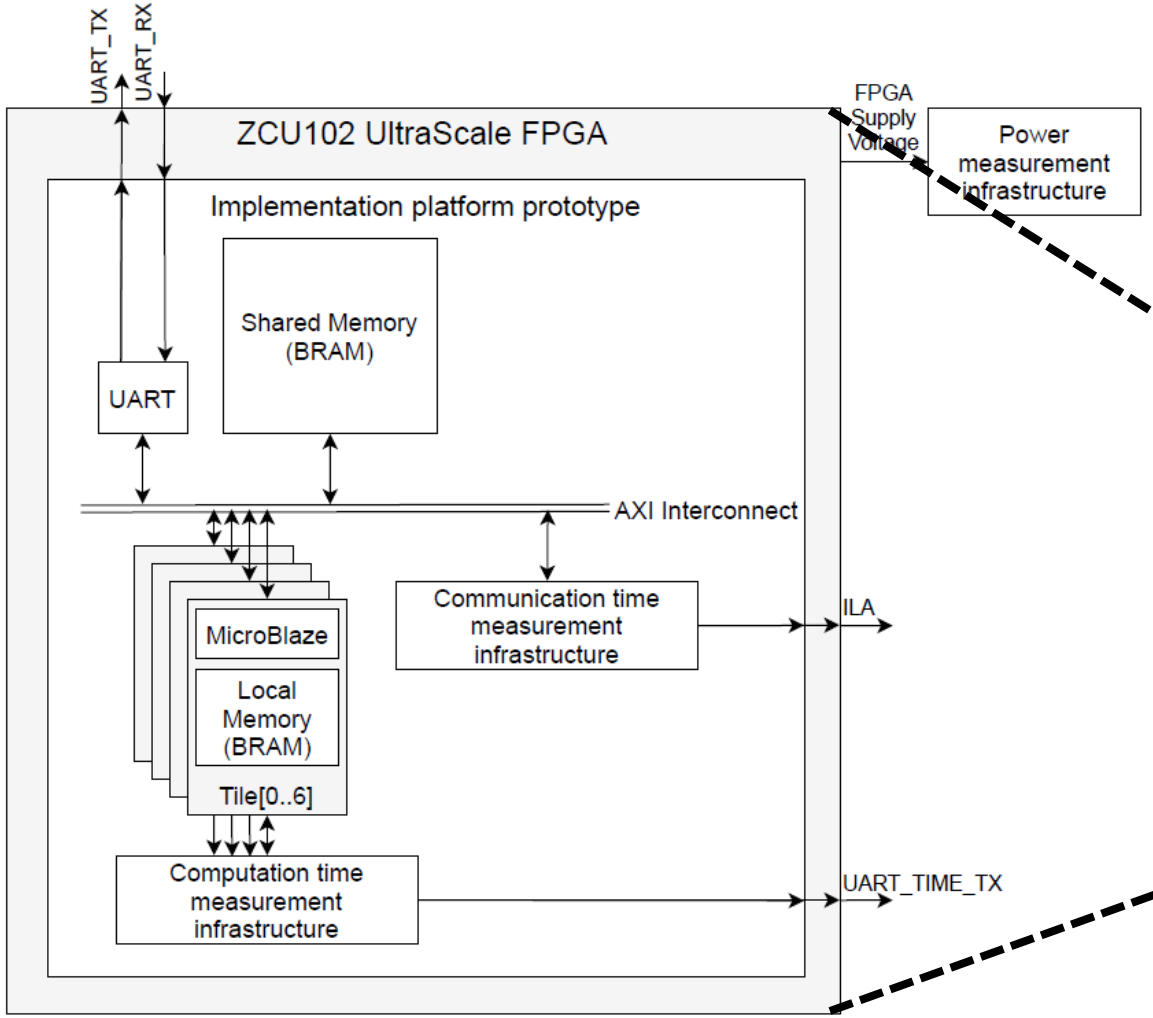
Quentin Dariol[1], Sébastien Le Nours[2], Ralf Stemmer[1], Domenik Helms[1], Kim Grüttner[1], Sébastien Pillement[2]

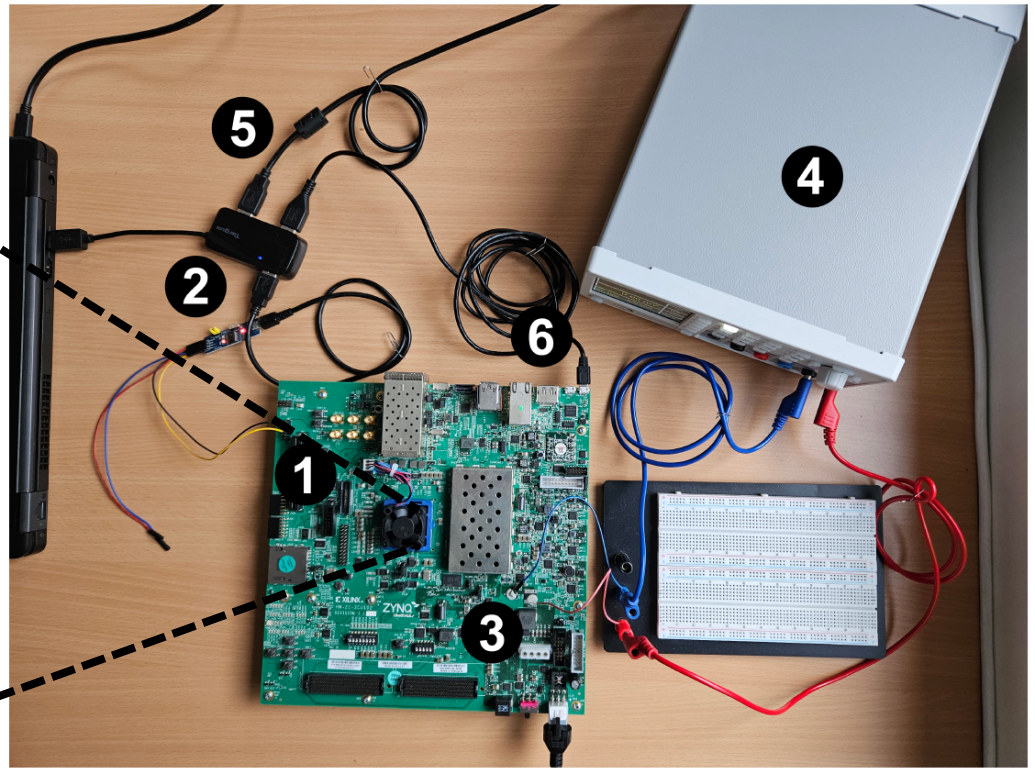
1: German Aerospace Center (DLR) – Institute Systems Engineering for future mobility (SE), Oldenburg
2: IETR Nantes, ASIC team

- Check our open source Git repository:
  https://gitlab.univ-nantes.fr/lenours-s/pssim4ai-open

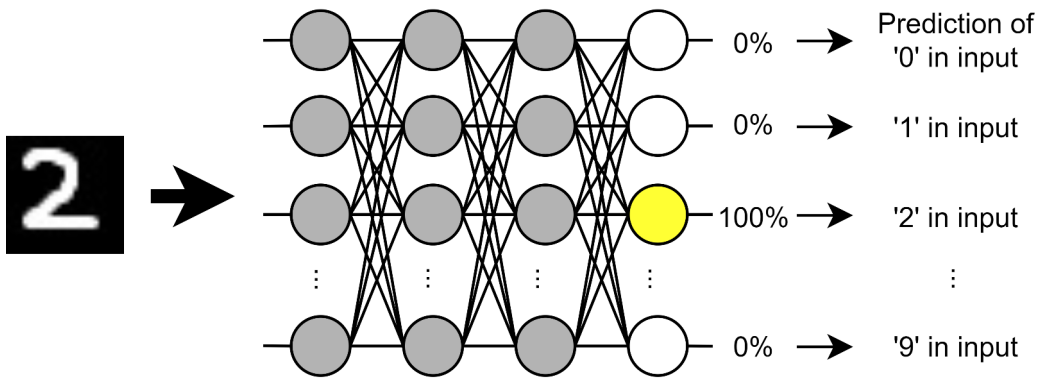
APPENDICES

Appendice – Prototype platform

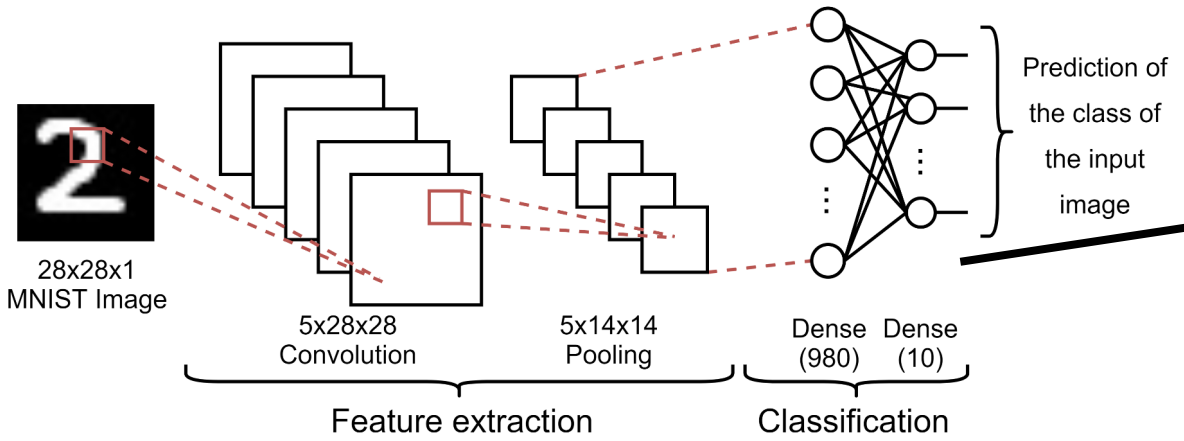


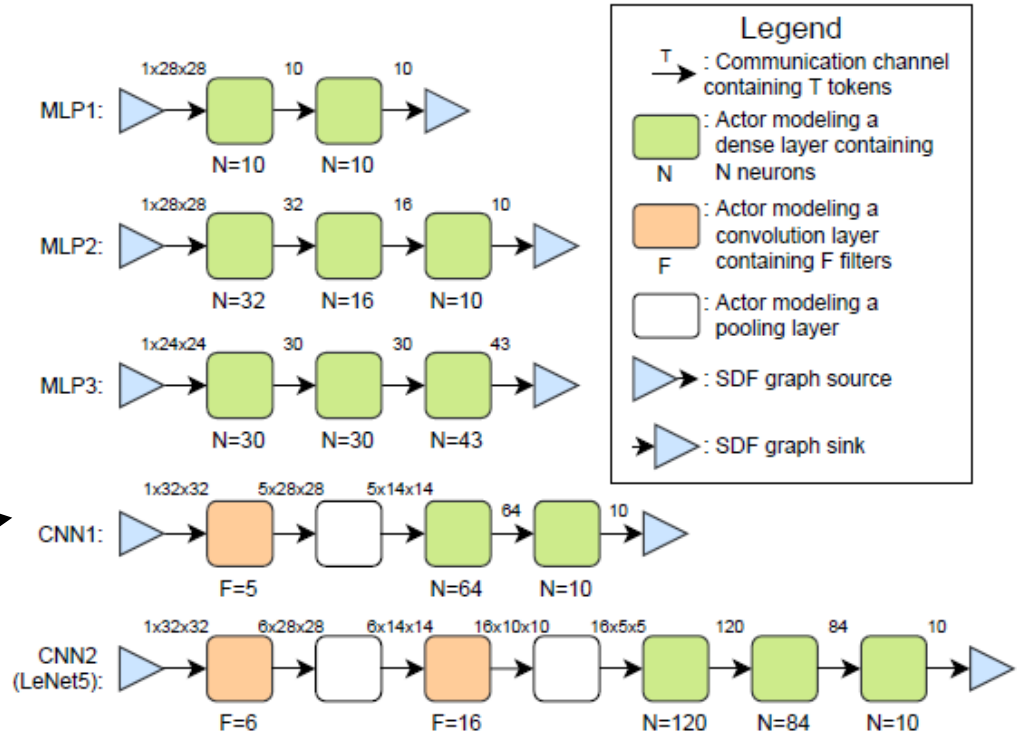


Appendice - Considered NNs

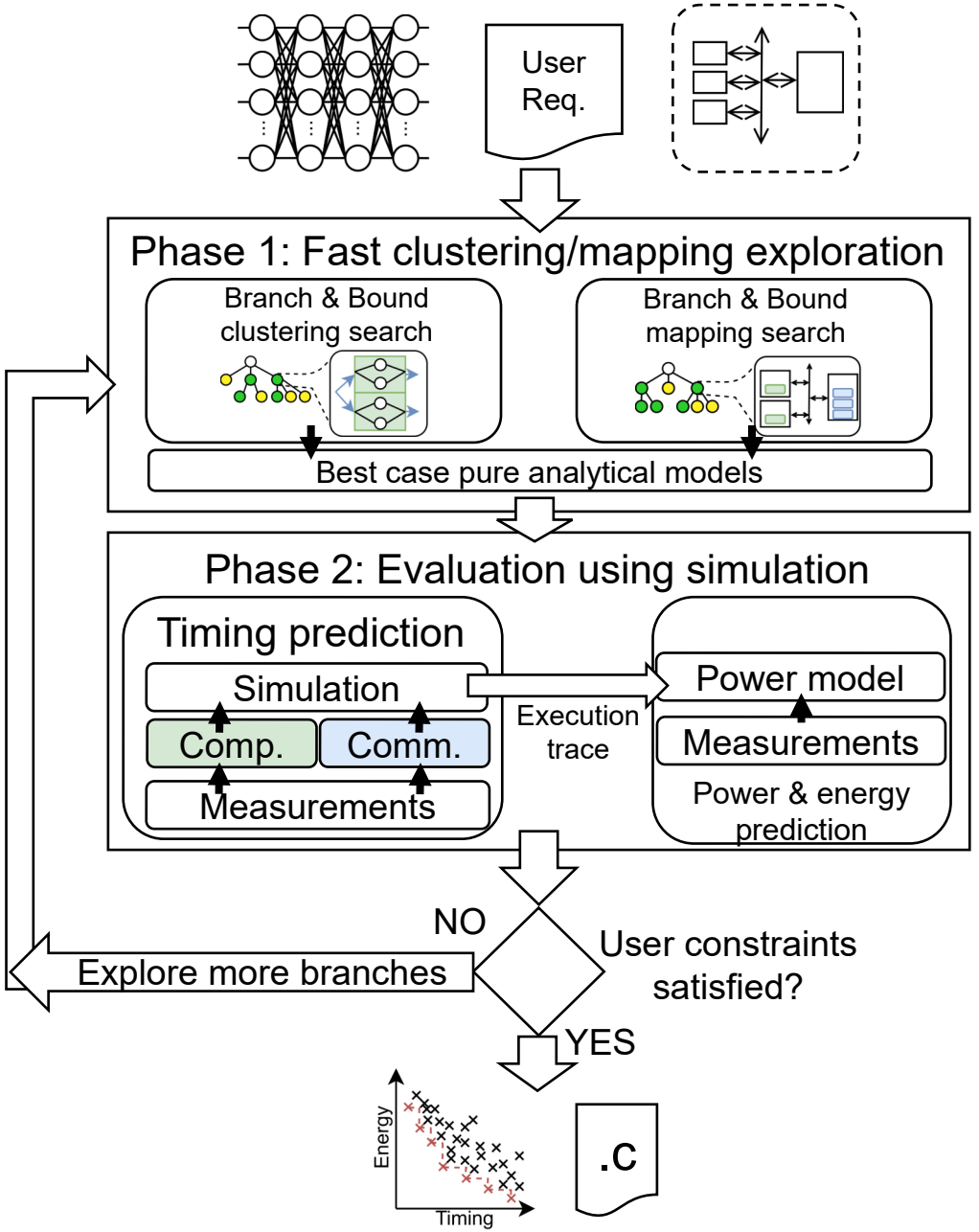


NN name	Number of layers	Data-set	Accuracy
MLP1	2	MNIST [11]	85%
MLP2	3	MNIST [11]	89%
MLP3	3	GTSRB [90]	20%
CNN1	4	MNIST [11]	77%
CNN2	7	MNIST [11]	N.A.





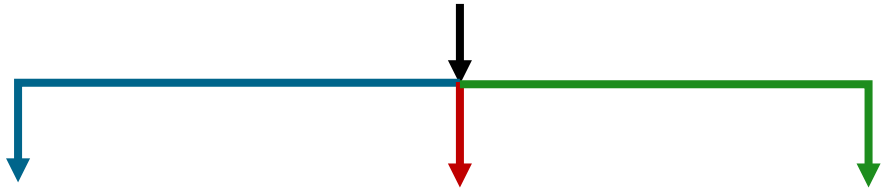
Appendice – Limitations



- Prediction error (standard deviation) on power and energy raises up to 7% with the communication rate per tile (70%).
- On single-core platforms with important private memory allocated (1024kB, 2048kB), power and energy modeling has error > 10%.

Appendice – Related work

Evaluation of NNs
on embedded platforms



Rapid prototyping

- Implementation and measurement on prototyping platform
- Highest accuracy
- Slow (need to deploy NN and measure)
- Limited architectural search

Analytical models

- Mathematical formula
- Fast to execute
- Lower accuracy than rapid prototyping
- Limitations to describe complex phenomenoms

Simulation

- Virtual platform described in HDL
- Slower than analytical, faster than rapid prototyping,
- High accuracy

- [Galanis2020] Galanis I. et al. “Inference and Energy Efficient Design of Deep Neural Networks for Embedded Devices”, IEEE Computer Society Annual Symposium on VLSI (ISVLSI), 2020
- [Tsimpourlas2018] Tsimpourlas F. et al. “A Design Space Exploration Framework for Convolutional Neural Networks Implemented on Edge Devices”, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCADICS), 2018
- [VelascoMontero2020] Velasco Montero D. et al. "PreVIous: A Methodology for Prediction of Visual Inference Performance on IoT Devices", IEEE Journal of Internet of Things, 2020
- [Guo2023] Guo X. et al. "Automated Exploration and Implementation of Distributed CNN Inference at the Edge", IEEE Journal of Internet of Things, 2023
- [Osterwind2022] Osterwind A. et al. "Hardware Execution Time Prediction for Neural Network Layers", IoT, Edge, and Mobile for Embedded Machine Learning (ITEM), 2022
- [Venieris2019] Venieris, S. and Bouganis, C.-S. "fpgaConvNet: Mapping Regular and Irregular Convolutional Neural Networks on FPGAs", IEEE Transactions on Neural Networks and Learning Systems, 2019
- [Parashar2019] Parashar, A. et al. "Timeloop: A Systematic Approach to DNN Accelerator Evaluation", ISPASS 2019
- [Garbay2021] Garbay, T. et al. "CNN Inference Costs Estimation on Microcontrollers: the EST Primitive-based Model", IEEE International Conference on Electronics, Circuits, and Systems (ICECS), 2021
- [Lee2022] Lee, J. et al. "Implication of Optimizing NPU Dataflows on Neural Architecture Search for Mobile Devices" - ACM Transactions on Design Automation of Electronic Systems (TODAES), 2022
- [Sombatsiri2019] Sombatsiri, S. et al. "A Design Space Exploration Method of SoC Architecture for CNN-based AI Platform", Synthesis And System Integration of Mixed Information technologies (SASIMI), 2019

Work	HW target	Evaluation speed	Accuracy Timing	Accuracy Power/Energy	Shared resource contention	Inter-layer parallelism	Intra-layer parallelism	Power management	HW dimensions
[Galanis2020]	GPU	✗	✓✓	✓✓	✓	✓	≈	✗	✗
[Tsimpourlas2018]	VPU	✗	✓✓	✓✓	✓	✓	✓	✗	✗
[VelascoMontero2020]	Multicore	✓✓	✓	⊘	✗	✓	≈	✗	✗
[Guo2023]	All	✓✓	✓	✓	✗	✓	≈	✗	✗
[Osterwind2022]	VPU	✓✓	✗	⊘	✗	✓	✓	✗	✗
[Venieris2019]	FPGA	✓✓	✓	⊘	✗	✓	✓	✗	✗
[Parashar2019]	FPGA, GPU	✓✓	✓	✓	✗	✓	✓	✗	✓
[Garbay2023]	MCU	✓✓	✓	✓	✗	✓	✗	✓	✗
[Lee2022]	NPUs	?	✓	⊘	≈	✓	✓	✗	✓
[Sombatsiri2019]	SoC	✗ (~100s)	✓	⊘	✓	✓	✓	✗	✓
THIS WORK	Multicore	✓	✓	✓	✓	✓	✓	✓	✓

Evaluation speed:
- ✓✓ : ~1ms
- ✓ : <60s
- ✗ : >60s

Accuracy:
- ✓✓ : ~100%
- ✓ : >90%
- ✗ : <90%
- ⊘ : N.C.

Other criterias:
- ✓ : Yes
- ✗ : No
- ≈ : Partial