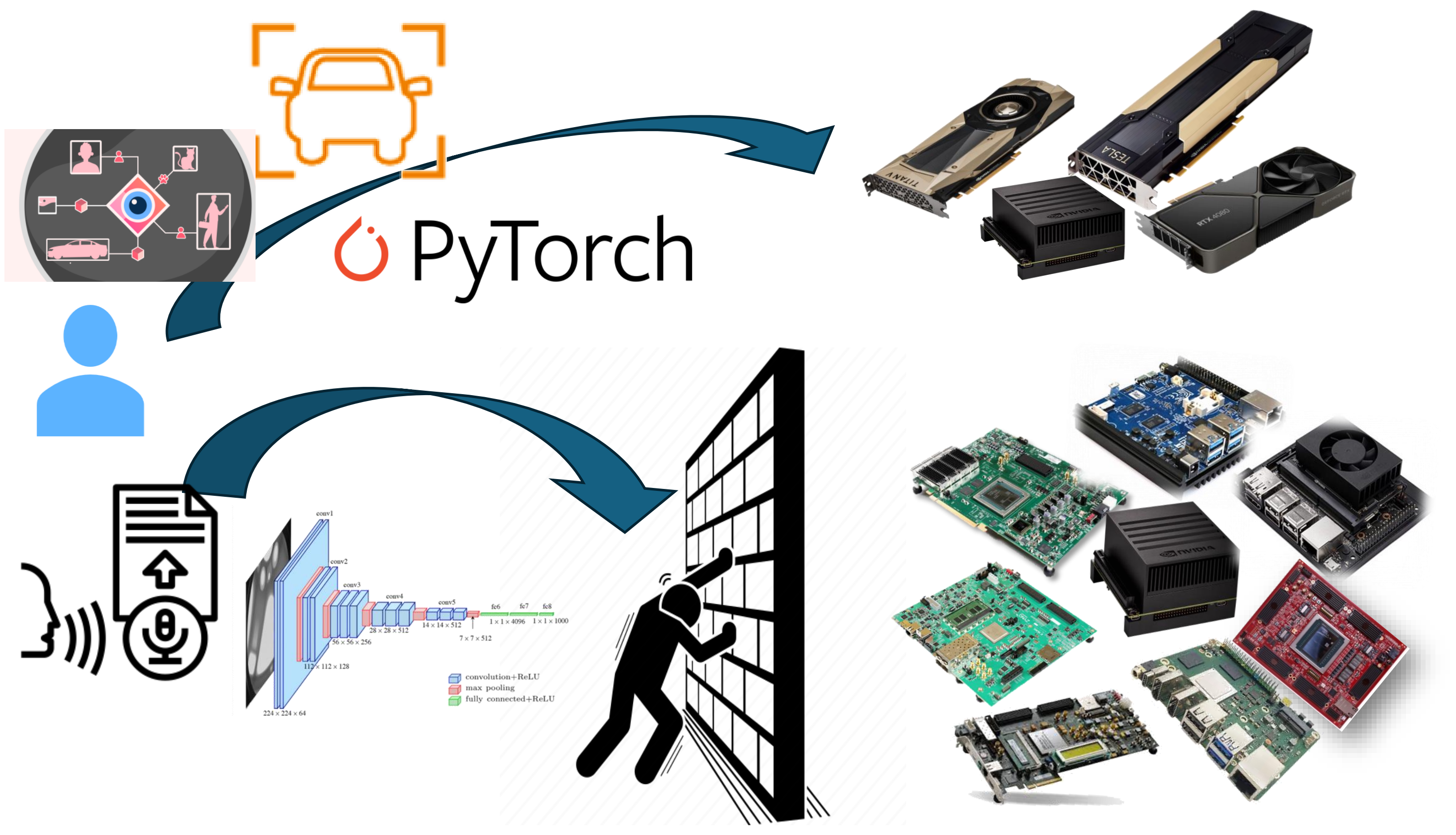
Seamless and Rapid PyTorch Model Deployment in Heterogeneous SoC

H. Umut Suluhan, Ali Akoglu
{suluhan,akoglu}@arizona.edu
Electrical and Computer Engineering Department, University of Arizona

IPDPS 2024 • San Francisco, California USA

Reconfigurable Computing Lab

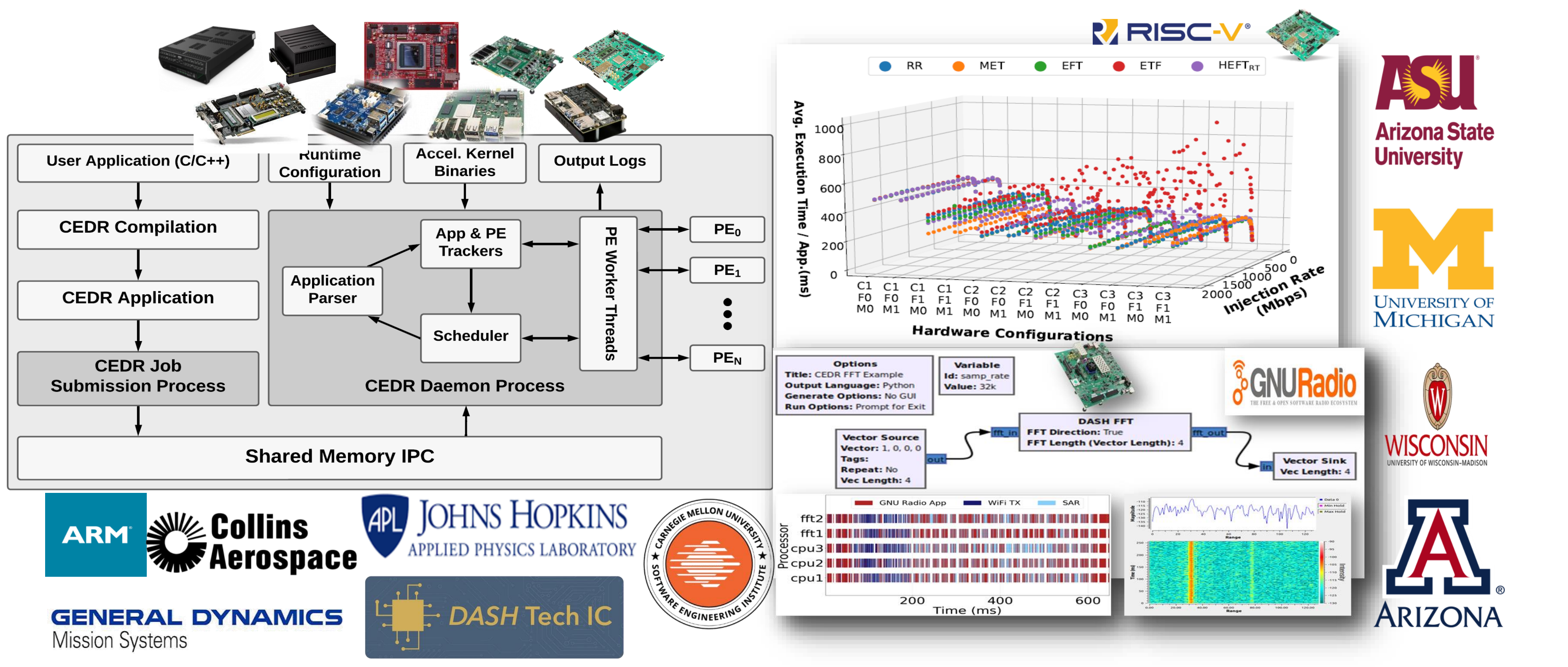
Motivation



Runtime capable of managing pool of CPU cores and accelerators are necessary

- PyTorch models can be deployed to GPUs seamlessly. However, GPU fails to meet energy requirements of edge devices. While PyTorch offers deployment on energy efficient FPGA systems, a rapid and seamless flow is yet to be provided.
- Deployment on heterogeneous systems is particularly challenging requiring a degree of hardware expertise. There are challenges from resource management perspective, productive application development and hardware agnostic deployment.
- Although various approaches tackle some of the above challenges, a system level solution that addresses all of them has yet to be designed.
- **Goal: Enabling productive, rapid and seamless PyTorch model deployment on heterogeneous SoCs considering**
- hardware agnostic application development
- balance trade-off between throughput and energy efficiency
- explore SoC configurations for PyTorch based workflows

Background

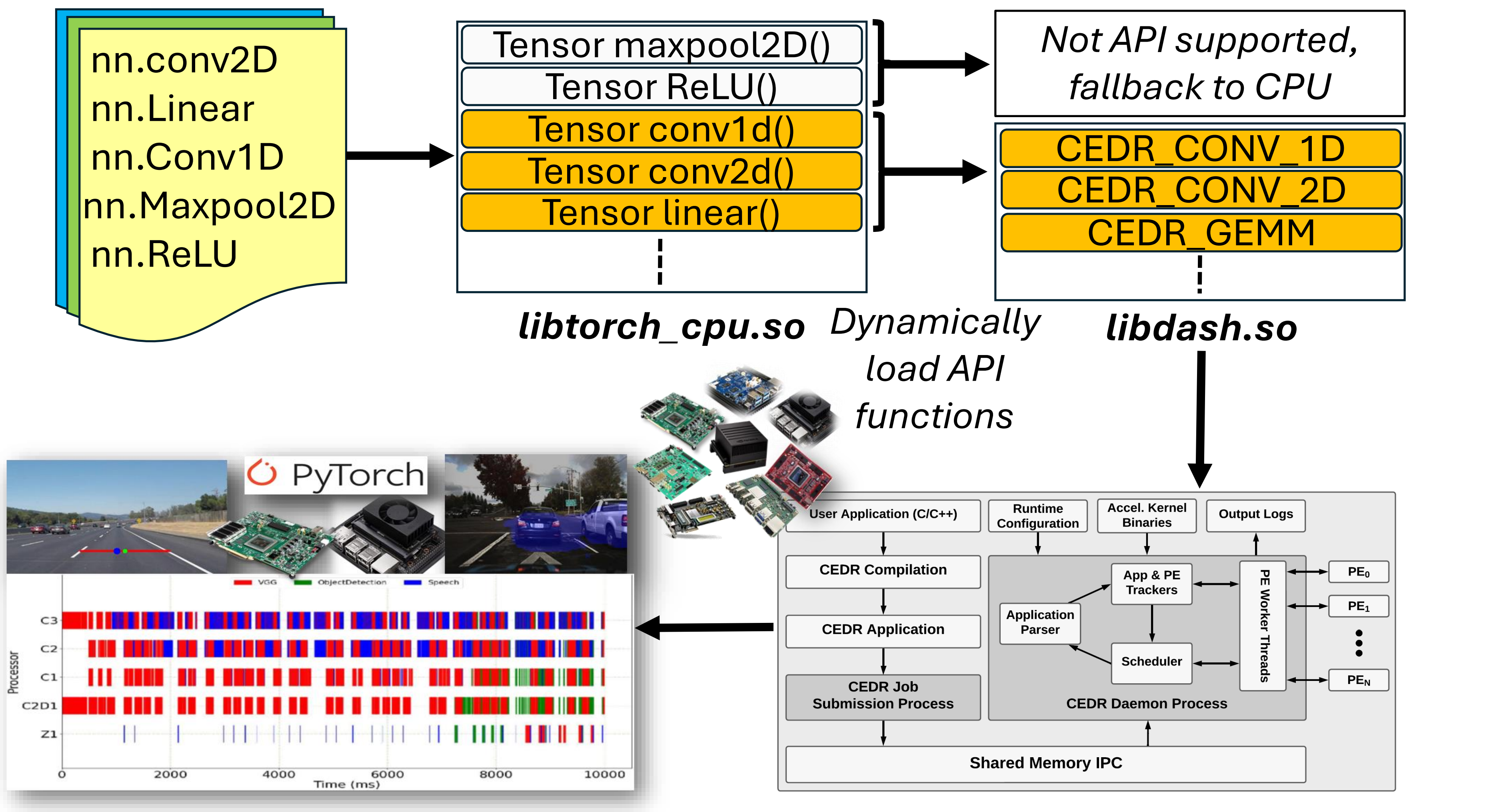


CEDR[1] – A Compiler-Integrated, Extensible DSSoC Runtime

[1]J. Mack, S. Hassan, N. Kumbhare, M. Castro Gonzalez, and A. Akoglu, "CEDR: A compiler-integrated, extensible DSSoC runtime," *ACM Trans. Embed. Comput. Syst.*, vol. 22, no. 2, Jan. 2023, issn: 1539-9087. doi: 10.1145/3529257

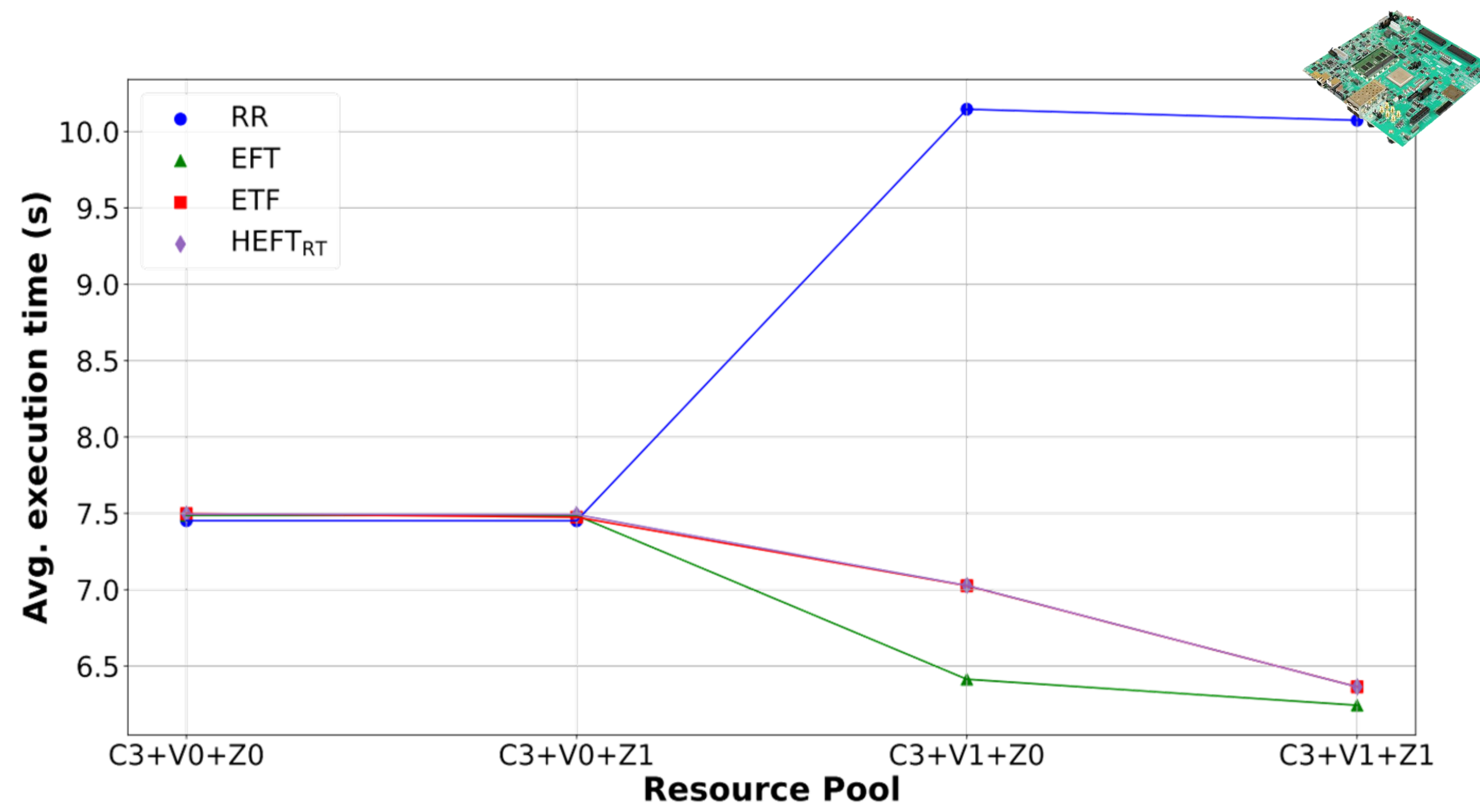
A rapid and seamless PyTorch model deployment technique is needed

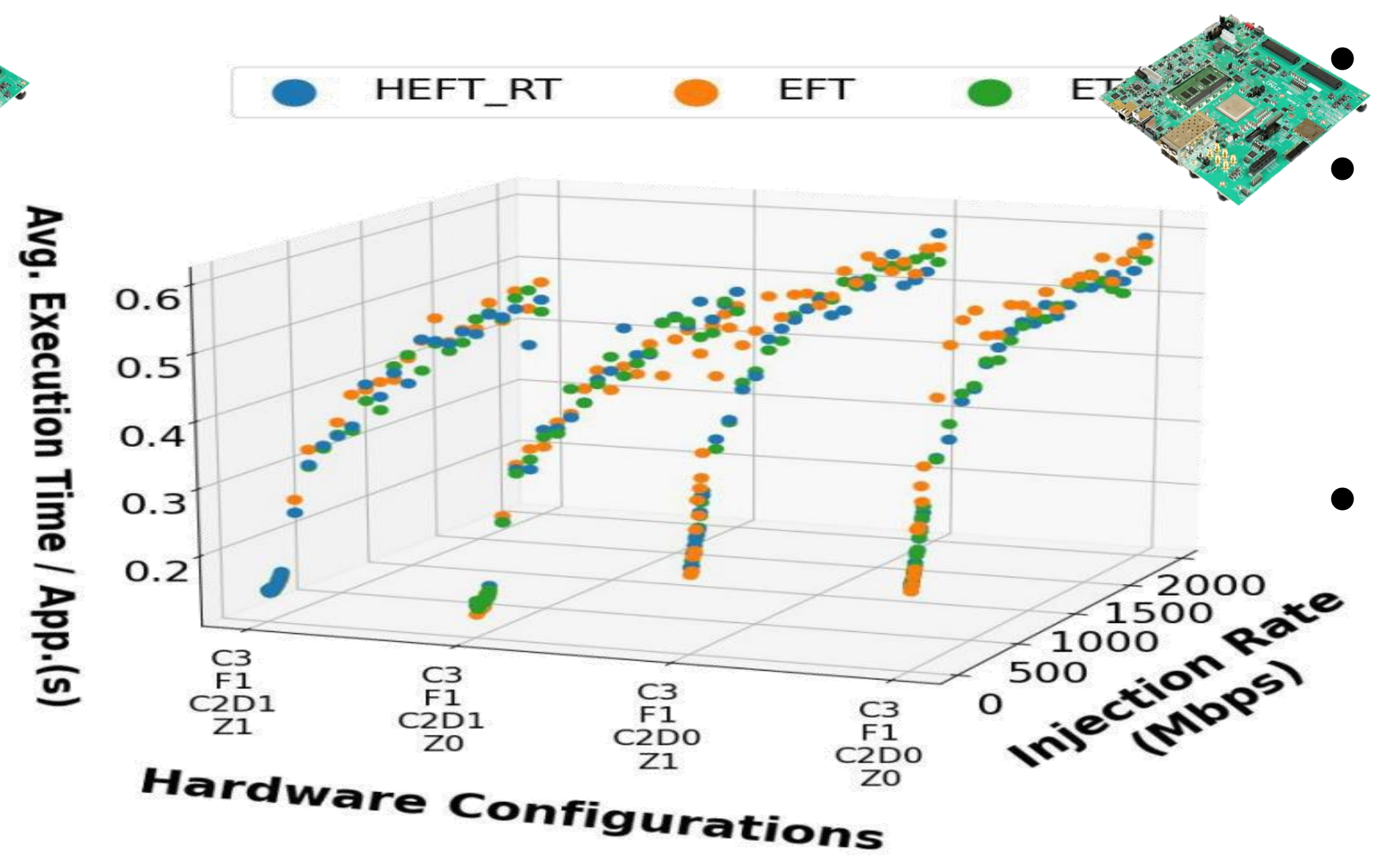
Approach



- Dynamically loads accelerator supported functions during runtime
- Capable of running any PyTorch model on the heterogeneous SoC

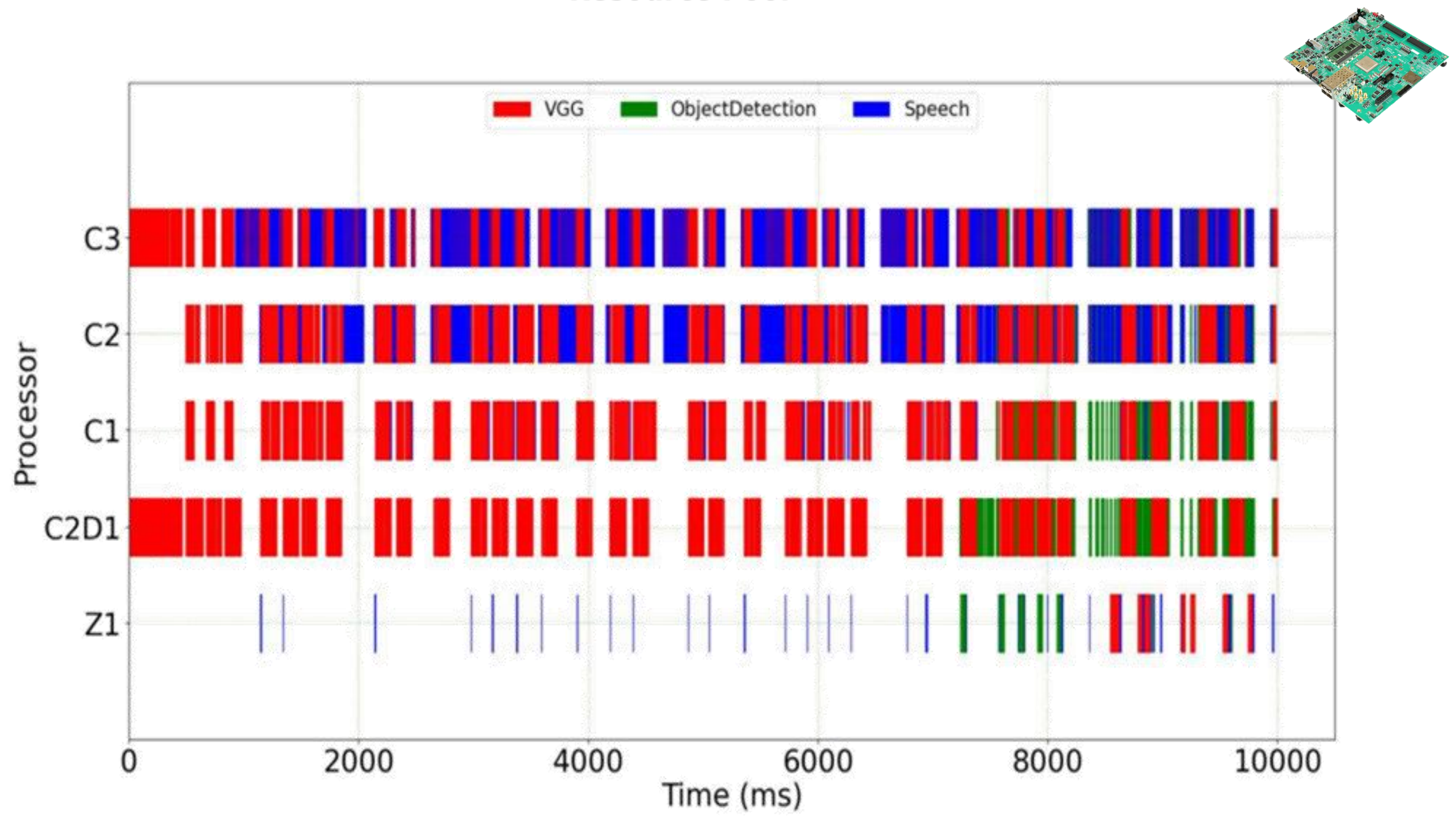
Experimental Setup and Results





- 3 CPUs and Conv2D, FFT and ZIP accelerators
- 3 PyTorch and 2 signal processing applications
  - Object Detection, VGG, Speech Classification
  - WiFi-TX, Pulse Doppler
- 3 distinct scheduling heuristics
  - Earliest Finish Time (EFT)
  - Earliest Time to Finish (ETF)
  - Heterogeneous Earliest Finish Time (HEFT-RT)[2]

[2] J. Mack et al. "Performant, multi-objective scheduling of highly interleaved task graphs on heterogeneous system on chip devices," IEEE Transactions on Parallel and Distributed Systems, vol. 33, no. 9, pp. 2148–2162, 2022 doi.org/10.1109/TPDS.2021.3135876



Conclusions and Future Work

For the first time, PyTorch application developers have access to FPGA-based execution without having to become hardware experts, which balances trade-off between throughput and energy efficiency, and enables exploration of SoC configurations for dynamic workloads. Our next step involves designing resource management heuristics for machine learning workloads sharing an edge system with tight constraints utilizing this framework.

Acknowledgments

This material is based on research sponsored by the Air Force Research Laboratory (AFRL) and the Defense Advanced Research Projects Agency (DARPA) under agreement number FA8650-18-2-7860. We appreciate the continuous and generous support from the AMD University Program, including the donation of the FPGA prototyping board used in this work.