# DECIDABLE VARIABLE-RATE DATAFLOW FOR HETEROGENEOUS SIGNAL PROCESSING SYSTEMS

Yujunrong Ma[1], Jiahao Wu[1] , Shuvra S. Bhattacharyya[1], Jani Boutellier[2]

*[1] University of Maryland, United States*

*[2] University of Vaasa, Finland*

# Motivation

- PRUNE (PSM Runtime Environment, Boutellier et al. 2018): A high-performance, dynamic, decidable dataflow model of computation and associated design framework for signal processing applications

- Variable-Rate Dataflow (VRDF, Wiggers et al. 2008): modeling of data-dependent communication behavior
  - VRDF can improve model analyzability due to the pre-indicated token rate limits and maintain sufficient expressiveness for adaptive behavior at the same time

- In previous work[1], PRUNE with adaptations for variable token rates has been preliminarily demonstrated to have performance and expressiveness benefits

[1] J. Boutellier and S. S. Bhattacharyya (2017) "Low-power heterogeneous computing via adaptive execution of dataflow actors", IEEE SiPS

University of Vaasa

University of Maryland 1856 — Department of Electrical & Computer Engineering

UMIACS — University of Maryland Institute for Advanced Computer Studies

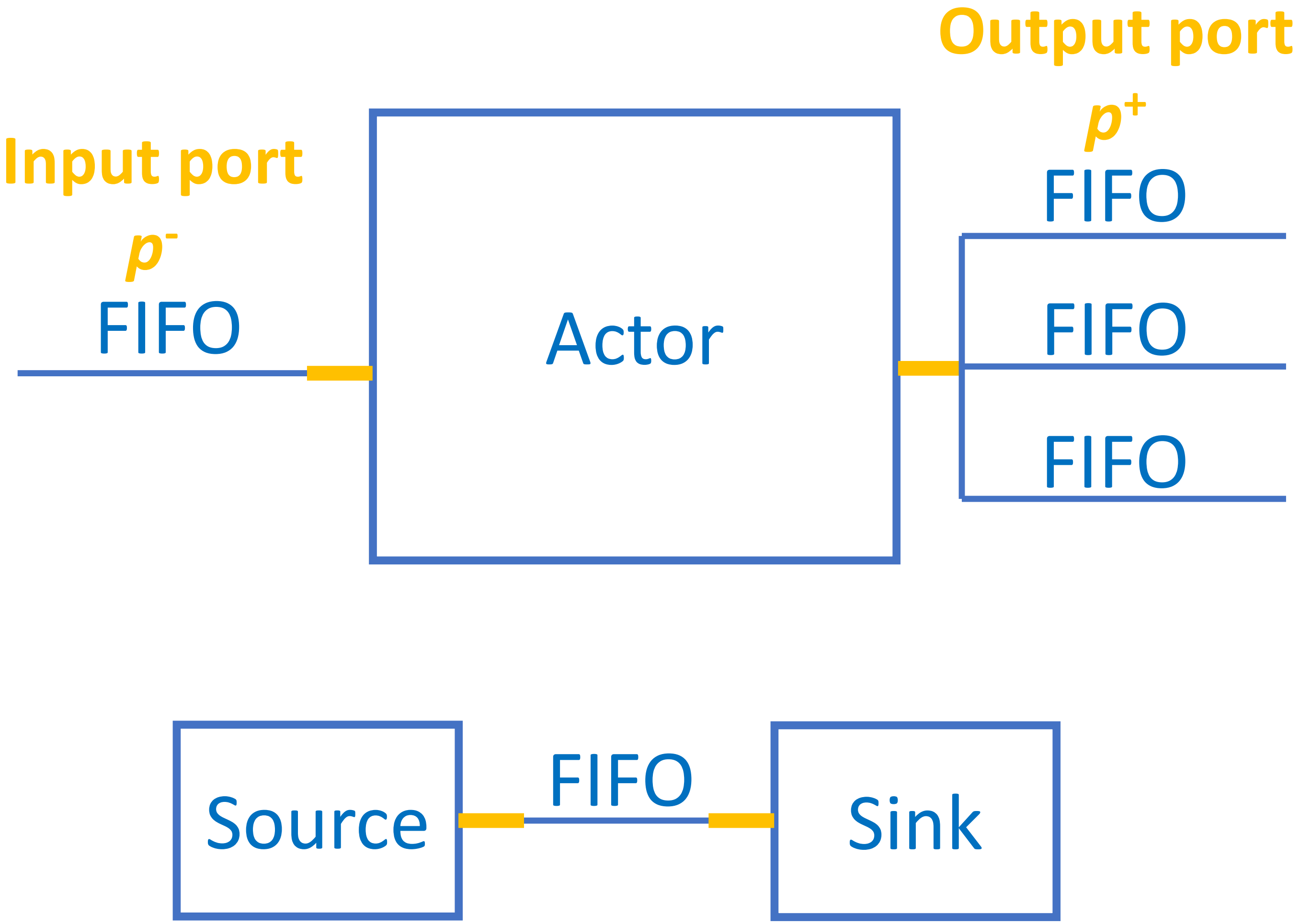
# Introduction

- This work: VR-PRUNE – a high-performance, flexible, dynamic decidable dataflow model of computation
  - Formalization of the preliminary work[1]
  - Elaboration of PRUNE Model of Computation and framework

- VR-PRUNE
  - is a hybrid between the PRUNE MoC  and the VRDF MoC: VR-PRUNE keeps the decidability of PRUNE, but adds to it support for variable token rates
  - is a Linux-based dataflow computing framework
  - supports OpenCL devices such as GPUs and multicore CPUs

University of Vaasa
DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING
UMIACS

# Background: Model of Computation




- A FIFO channel is connected to an actor through an actor port
- An output port $p^+$ can be connected to multiple FIFOs, but every input port $p^-$ has only one FIFO connected to it
- Each FIFO has unique source and sink ports.

University of Vaasa
UNIVERSITY OF MARYLAND 1856 DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING
UMIACS University of Maryland Institute for Advanced Computer Studies

# VR-PRUNE: MoC

## VR-PRUNE Port Types

1) Static regular ports (SRP): Fixed token consumption/production rate
2) Dynamic regular ports (DRP): Two fixed token rates, which are called *active token rate* `atr`(*p*) and *inactive token rate* `itr`(*p*) respectively

## VR-PRUNE Actor Types

1) Static processing actor:
   - All ports have just one token rate, the `atr`
2) Configuration actor:
   - Have one or multiple control output ports, which must be connected to the control input port of a dynamic actor or a dynamic processing actor
   - The control output ports must be SRPs with a token rate of unity

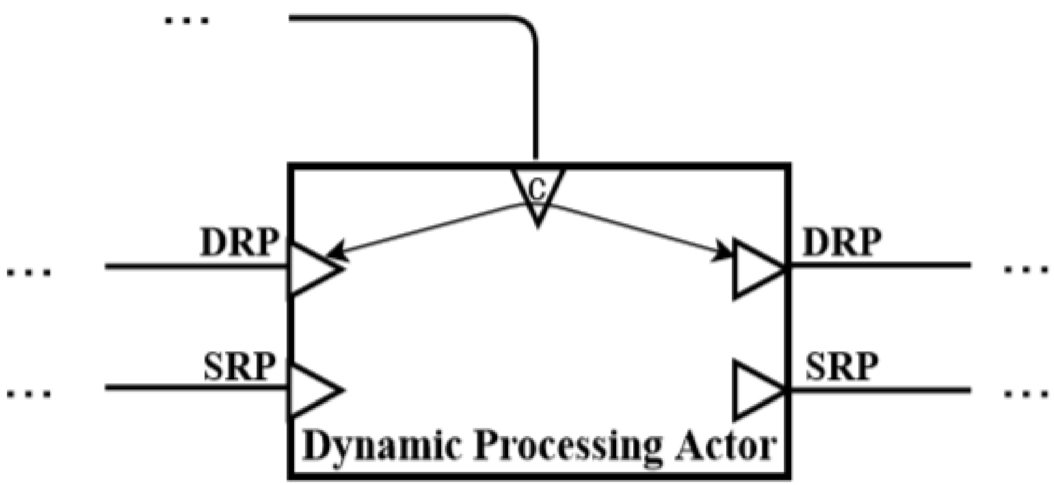
# VR-PRUNE: MoC

## VR-PRUNE Actor Types (cont'd)

3) Dynamic Actor (DA)
   - At least one control port (1,2,…)
   - At least one dynamic rate port (DRP) (1,2,…)
   - Any number of static rate ports (SRPs) (0,1,2,…)

4) Dynamic Processing Actor (DPA)
   - At least one control port (1,2,…)
   - At least one dynamic rate port (DRP) on both input and output side (2,…)
   - Any number of static rate ports (SRPs) (0,1,2,…)

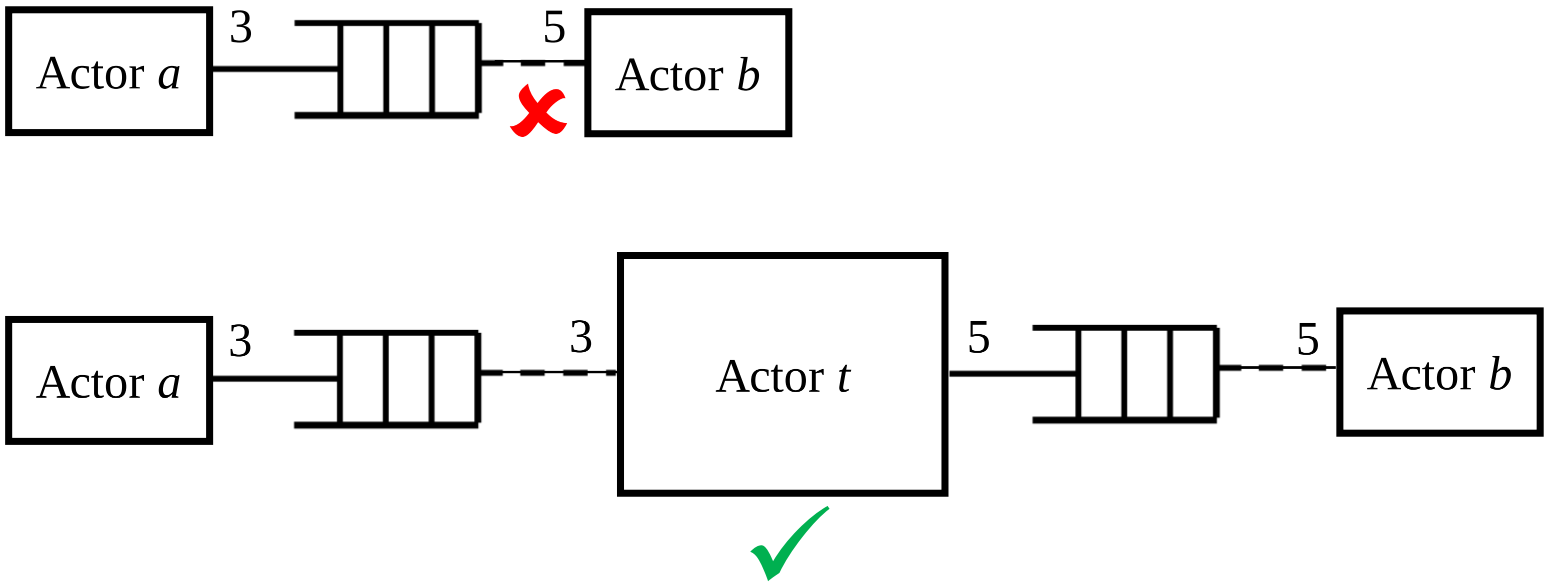
Note: For VR-PRUNE, the token rates of DRPs in DA and DPA can be set to any integer value between 0 and `atr`(*p*).



# VR-PRUNE: MoC

## Other constraints

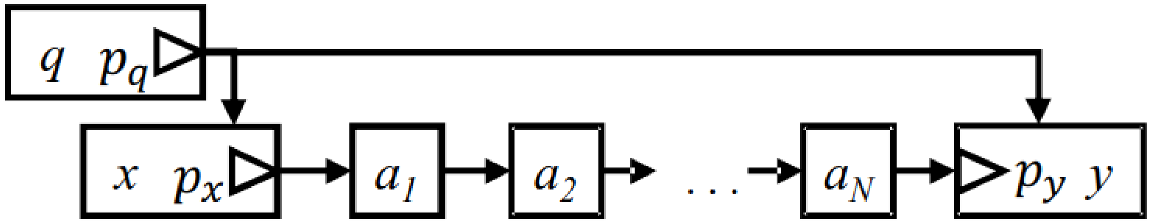
- Symmetric-rate dataflow behavior:  The token consumption rate equal the production rate for every FIFO



# VR-PRUNE: Design Rules

- **Linked port control rule**

For each pair $\{p_x, p_y\}$ of linked DRPs, the ports must be controlled by the same control output port $p_q$, and by the same element of the associated control token

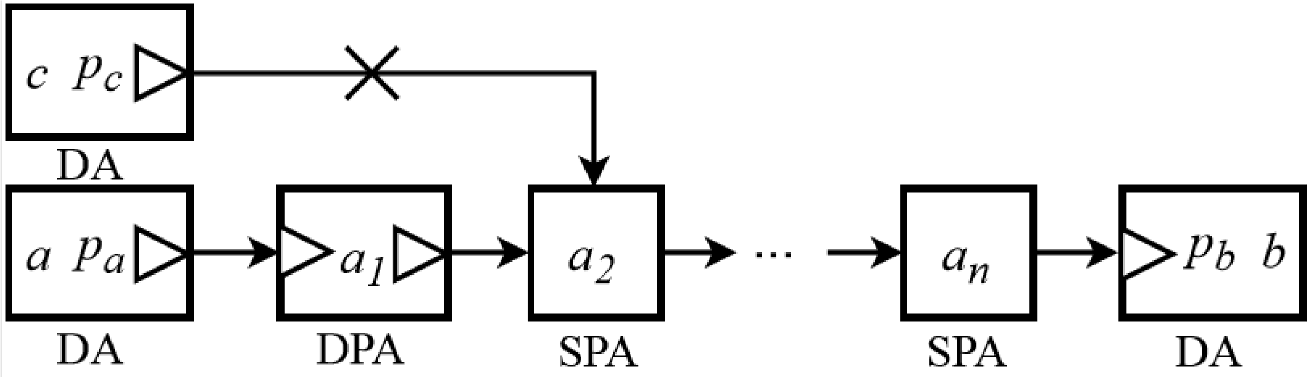



| | $p_x$ | $p_y$ |
|---|---|---|
| $p_q$ | 1 | 1 |

# VR-PRUNE: Design Rules

- **Connecting subchain rule:**

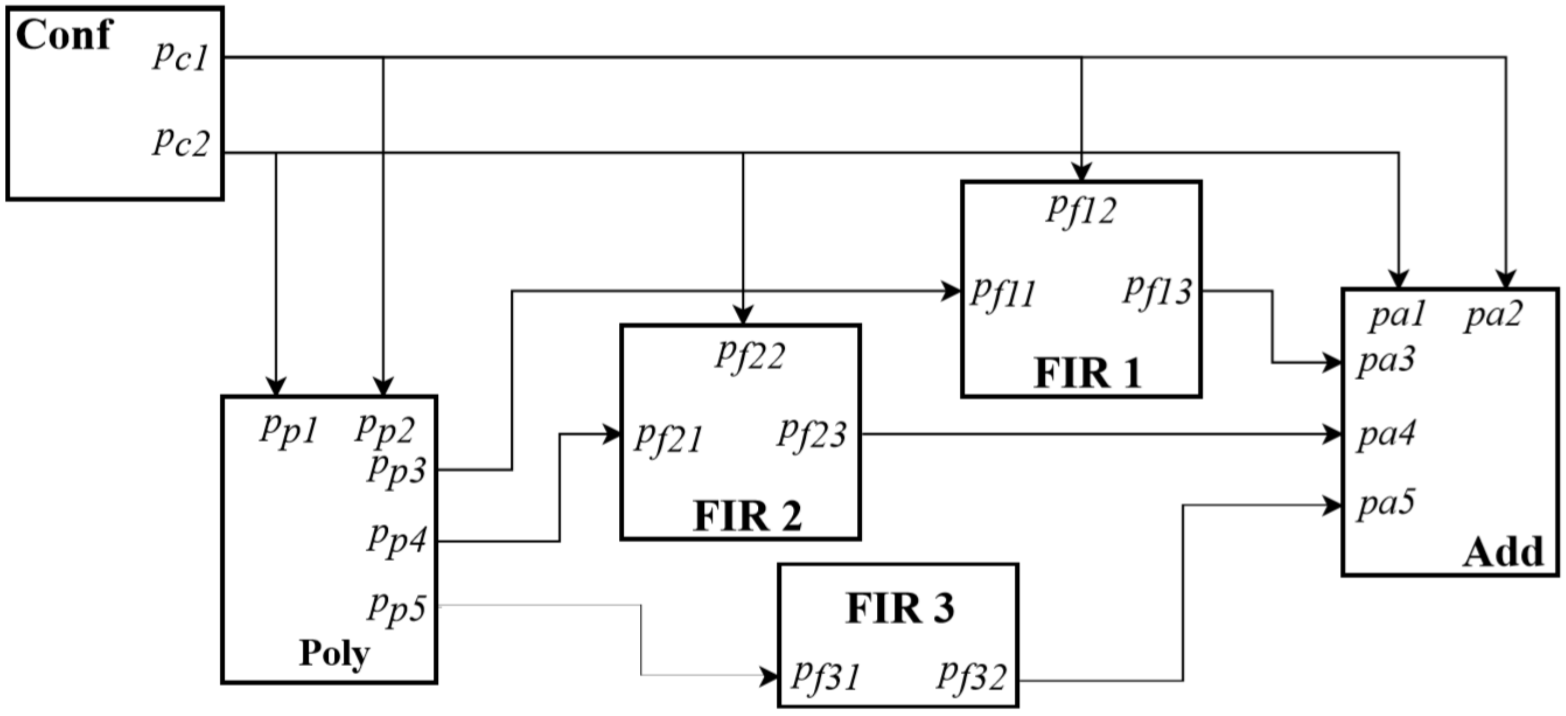
  1) Actor $a_i$ must be a SPA or DPA
  2) Each connecting subchain, to which actor $a_i$ belongs, must be associated with the two dynamic actors *a* and *b*




- In total, VR-PRUNE has five design rules – the rest are specified in the paper

University of Vaasa
University of Maryland Department of Electrical & Computer Engineering
UMIACS

# VR-PRUNE: Dynamic Processing Graph




- A DPG consists of one configuration actor, two dynamic actors and any number of DPAs or SPAs
- FIR1 and FIR2 are dynamic processing actors, Poly and Add actors are DAs with DPRs on output and input side each

University of Vaasa

UNIVERSITY OF MARYLAND 18 56

DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING

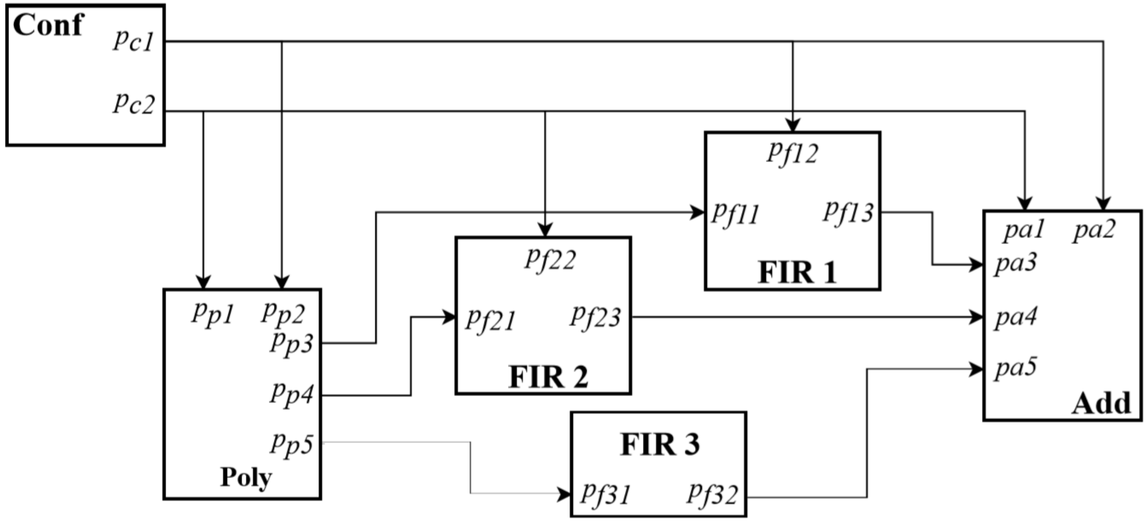
UMIACS

University of Maryland Institute for Advanced Computer Studies

# VR-PRUNE: Control Table

**Table 1**. Control table for the graph in Fig. 2.

| | $p_{p3}$ | $p_{p4}$ | $p_{f11}$ | $p_{f13}$ |
|---|---|---|---|---|
| $p_{c1}$ | [0 .. 2] | - | [0 .. 2] | [0 .. 2] |
| $p_{c2}$ | - | [0 .. 3] | - | - |
| | $p_{f21}$ | $p_{f23}$ | $p_{a3}$ | $p_{a4}$ |
| $p_{c1}$ | - | - | [0 .. 2] | - |
| $p_{c2}$ | [0 .. 3] | [0 .. 3] | - | [0 .. 3] |




- Rows are indexed by control output ports
- Columns are indexed by dynamic regular ports (DRPs)
- Blank entries indicate the absence of any control relationship
- Positive-valued entries specify indices into *control*

University of Vaasa

DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING

UMIACS

# VR-PRUNE: Experiments

**Table 2**. Platforms used for experiments.

| Tag | GPPs | GPU | Operating System |
|---|---|---|---|
| i7-940MX | Intel i7-6700HQ @ 2.60 GHz | NVidia GeForce 940MX | Ubuntu 18.04, g++ 7.0.0 |
| i7-GTX1080 | Intel i7-8700K @ 3.70GHz | NVidia Geforce GTX1080 | Ubuntu 18.04, g++ 7.0.0 |

The designer provides actor descriptions in C or OpenCL, based on which the VR-PRUNE compiler generates a top level file that realizes inter-actor communication

# VR-PRUNE: Experiments

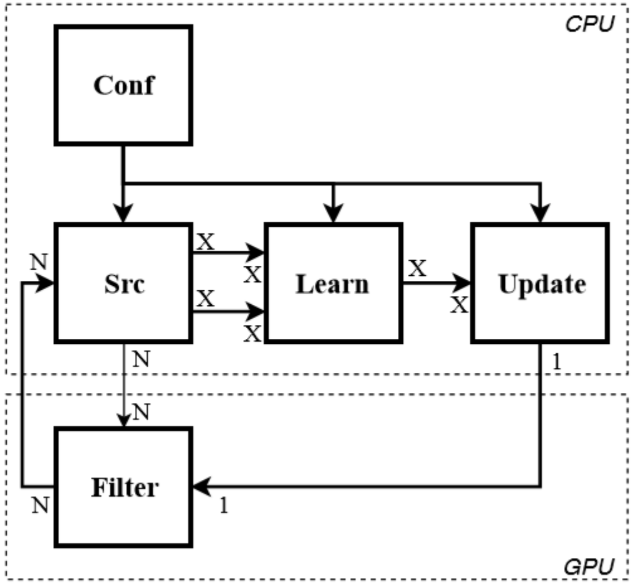



**Fig. 3**. VR-PRUNE model for the DU-DPD use case.

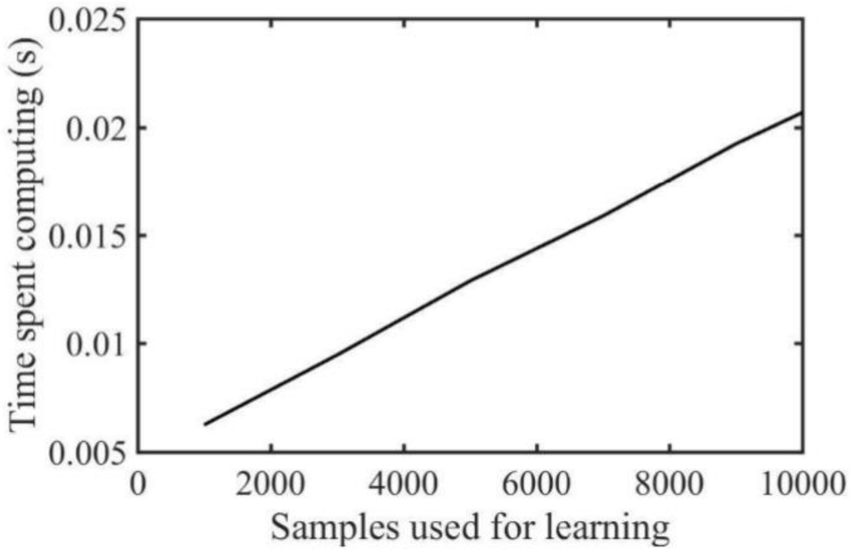



**Fig. 4**. Time spent in DU-DPD under VR-PRUNE on i7-GTX1080, as a function of samples used for learning.

Dynamic-token rate feature allows any number of samples between 1 and X

VR-PRUNE allows reducing the number of samples used for runtime learning

University of Vaasa

DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING

UMIACS

# VR-PRUNE: Experiments

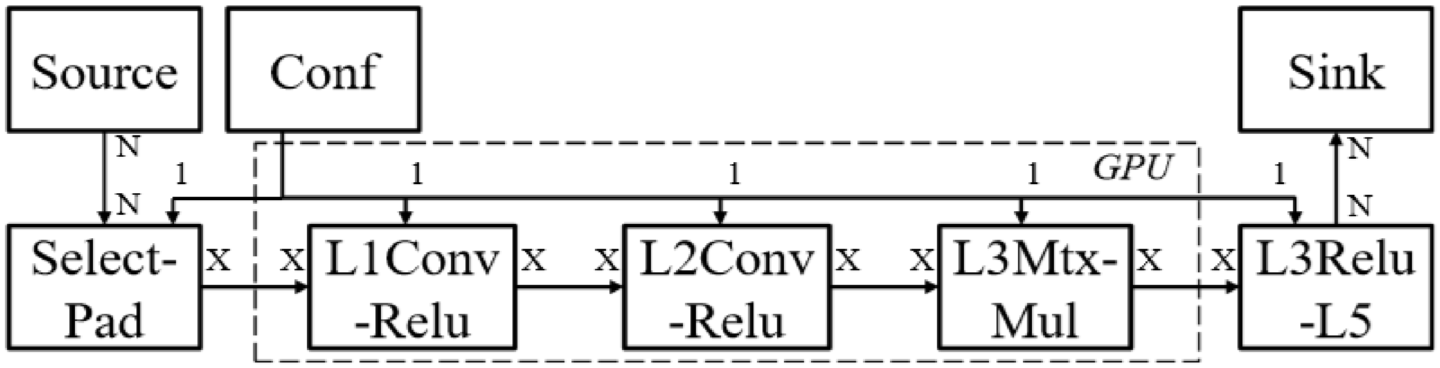



**Fig. 5**. The adaptive CNN application.

Components: two GPU-accelerated convolution layers followed by a GPU-accelerated dense layer and two more dense layers that have been combined into a single actor

University of Vaasa

DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING

UMIACS

# VR-PRUNE: Performance

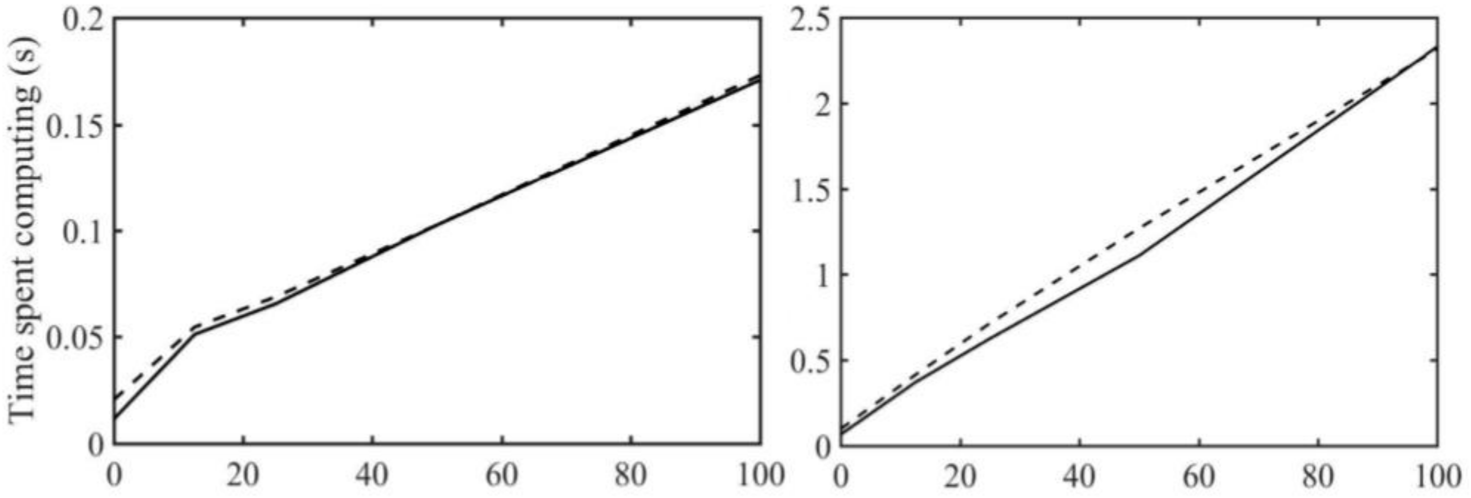



**Fig. 6**. Time spent in the CNN application as a function of % of frames processed: PRUNE (dashed line) vs. VR-PRUNE (solid line) on i7-GTX1080 (left) and i7-940MX (right).

The CNN inference (on/off) for each frame was randomly varied at runtime achieving the average percentage of frames processed for 0%, 12.5%, 25.0%, 50.0% and 100%, measuring the execution time for each percentage value

University of Vaasa

DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING

UMIACS

# Conclusion

- VR-PRUNE is a high-performance, flexible, dynamic decidable dataflow model of computation
  - Combines the high-performance, decidable PRUNE Model of Computation with some features of the VRDF Model of Computation
- The experiments show that the variable token rate feature of VR-PRUNE does not impose any overhead compared to conventional PRUNE – in contrast, VR-PRUNE is computationally slightly more efficient than conventional PRUNE

# More sources

- PRUNE is introduced in more detail in the following paper

J. Boutellier, J. Wu, H. Huttunen, and S. S. Bhattacharyya. PRUNE: Dynamic and decidable dataflow for signal processing on heterogeneous platforms. *IEEE Transactions on Signal Processing*, 66 (3), 654-665, 2018

https://ieeexplore.ieee.org/document/8106744

- The main PRUNE repository:

https://gitlab.com/jboutell/Prune