Status update of TA5

PUNCH4NFDI General Meeting

Jul 07 2025

M. Kramer A. Redelbach

Updates today - overview

Dynamic filtering

- → Prototype for dynamic filtering at MPIfR based on Effelsberg data successfully tested, TA5 document finalized, document on the outcome of ML training and link to code plus open data

- → Preparations for „Generic tools to both convert trained neural networks into efficient HLS/VHDL FPGA firmware … and to establish comparable software solutions“, document circulated in TA5
- → Presentation by Johann Voigt at FairMAT meeting last week

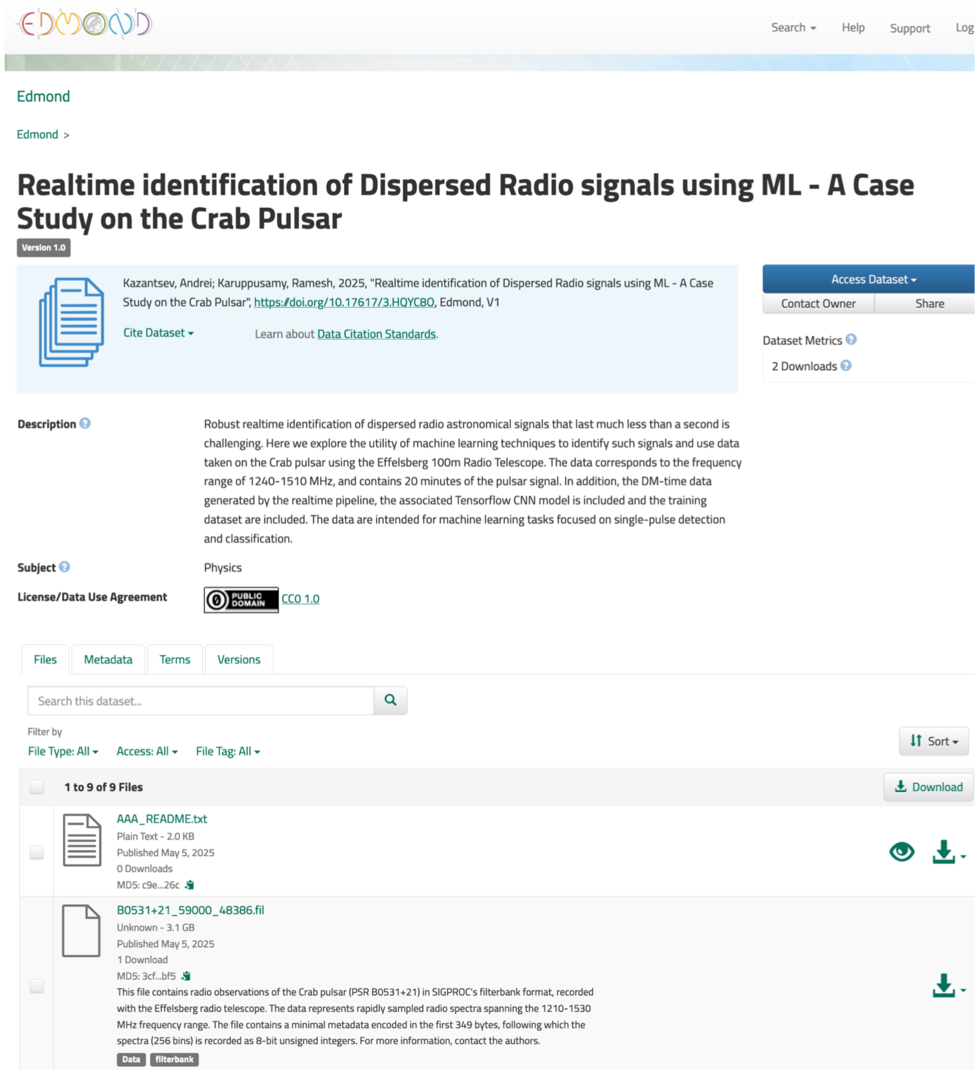
Scaling workflows

- → ML-PPA results from Tanumoy Saha

- → Developments for efficient caching

Dynamic filtering – test environment and open data

Robust realtime identification of dispersed radio astronomical signals that last much less than a second is challenging. Here we explore the utility of machine learning techniques to identify such signals and use data taken on the **Crab pulsar using the Effelsberg 100m Radio Telescope**.
Data corresponds to the frequency range of 1240-1510 MHz, and contains 20 minutes of the pulsar signal. In addition, the DM-time data generated by the realtime pipeline, the associated Tensorflow CNN model is included and the training dataset are included. The data are intended for machine learning tasks focused on single-pulse detection and classification.



Realtime identification of Dispersed Radio signals using ML - A Case Study on the Crab Pulsar

Kazantsev, Andrei; Karuppusamy, Ramesh, 2025, "Realtime identification of Dispersed Radio signals using ML - A Case Study on the Crab Pulsar", https://doi.org/10.17617/3.HQYC8O, Edmond, V1

Learn about Data Citation Standards.

Dataset Metrics

2 Downloads

Description Robust realtime identification of dispersed radio astronomical signals that last much less than a second is challenging. Here we explore the utility of machine learning techniques to identify such signals and use data taken on the Crab pulsar using the Effelsberg 100m Radio Telescope. The data corresponds to the frequency range of 1240-1510 MHz, and contains 20 minutes of the pulsar signal. In addition, the DM-time data generated by the realtime pipeline, the associated Tensorflow CNN model is included and the training dataset are included. The data are intended for machine learning tasks focused on single-pulse detection and classification.

Subject Physics

License/Data Use Agreement PUBLIC DOMAIN CC0 1.0

1 to 9 of 9 Files

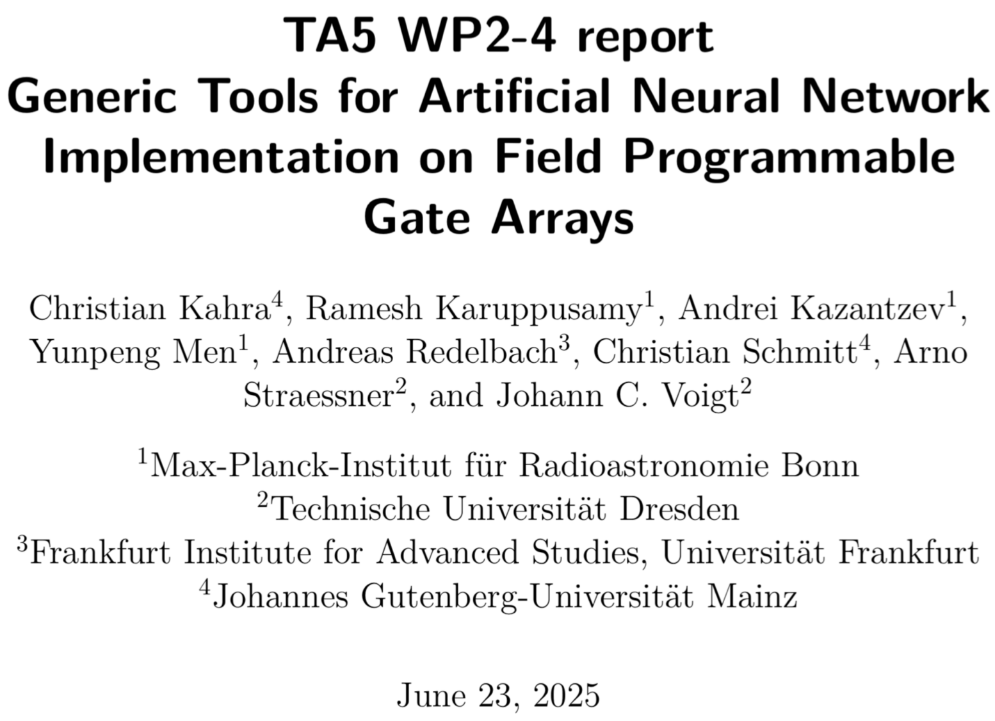
AAA_README.txt
Plain Text - 2.0 KB
Published May 5, 2025
0 Downloads
MD5: c9e...26c

B0531+21_59000_48386.fil
Unknown - 3.1 GB
Published May 5, 2025
1 Download
MD5: 3cf...bf5
This file contains radio observations of the Crab pulsar (PSR B0531+21) in SIGPROC's filterbank format, recorded with the Effelsberg radio telescope. The data represents rapidly sampled radio spectra spanning the 1210-1530 MHz frequency range. The file contains a minimal metadata encoded in the first 349 bytes, following which the spectra (256 bins) is recorded as 8-bit unsigned integers. For more information, contact the authors.

Nice example for fast ML use case
Feedback from users welcome!

https://edmond.mpg.de/dataset.xhtml?persistentId=doi:10.17617/3.HQYC8O

ML on FPGAs - document with results

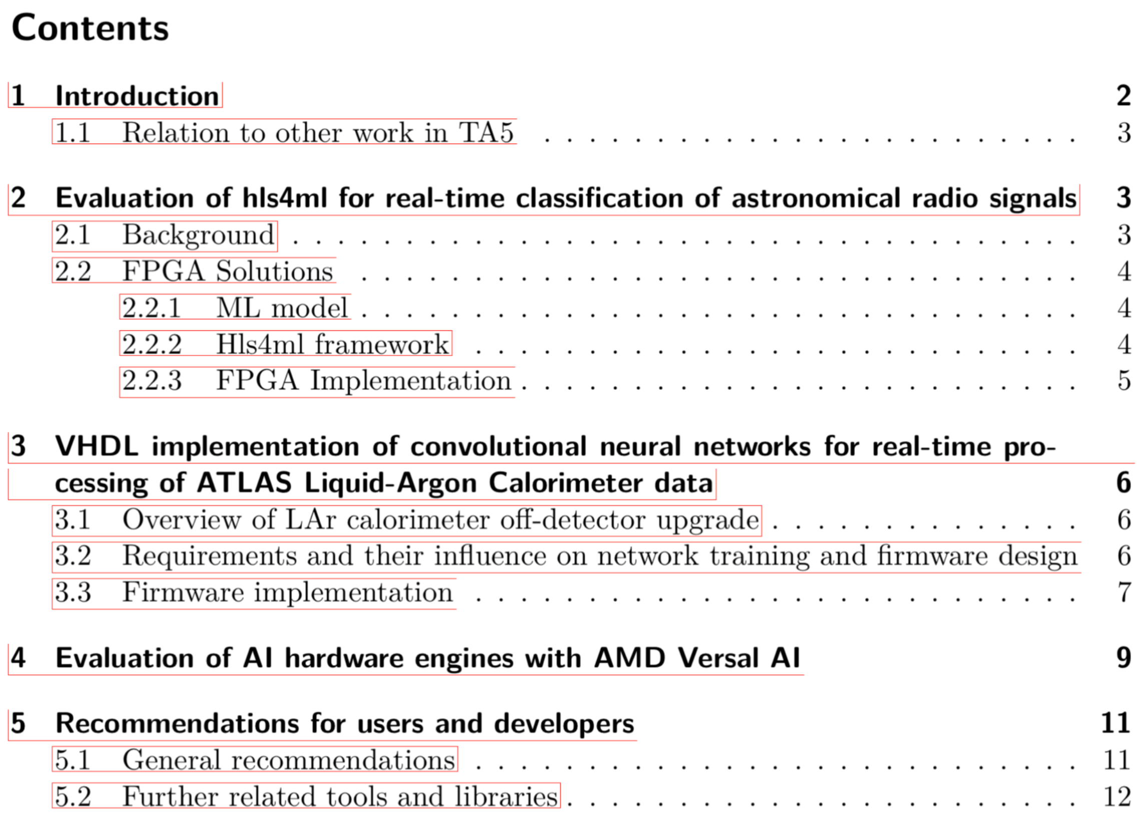


TA5 WP2-4 report
Generic Tools for Artificial Neural Network Implementation on Field Programmable Gate Arrays

Christian Kahra[4], Ramesh Karuppusamy[1], Andrei Kazantzev[1], Yunpeng Men[1], Andreas Redelbach[3], Christian Schmitt[4], Arno Straessner[2], and Johann C. Voigt[2]

[1]Max-Planck-Institut für Radioastronomie Bonn
[2]Technische Universität Dresden
[3]Frankfurt Institute for Advanced Studies, Universität Frankfurt
[4]Johannes Gutenberg-Universität Mainz

June 23, 2025

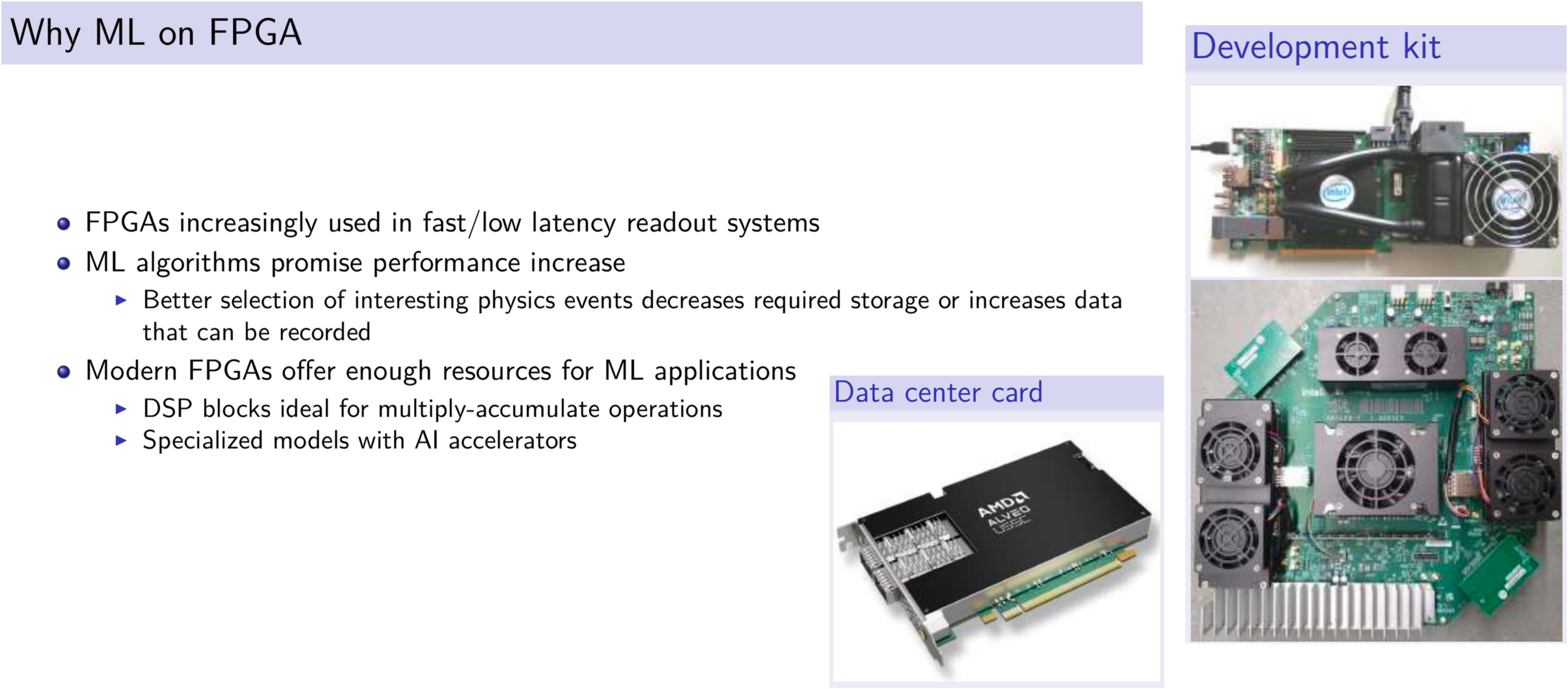


Contents

- → Evaluated different tools to implement neural network inference on FPGA for different applications: hls4ml, direct VHDL implementation, specialized hardware (Versal AI engines)
- → Dynamic field with high dependence on vendor tools and high degree of specialization
- → Document with experiences and **generalized recommendations**

ML on FPGAs – presentation

Presentation slides available:
https://events.fairmat-nfdi.eu/event/32/contributions/418/attachments/45/105/2025_07_03_ML_on_FPGA_PUNCH4NFDI.pdf



Why ML on FPGA

- FPGAs increasingly used in fast/low latency readout systems
- ML algorithms promise performance increase
  - Better selection of interesting physics events decreases required storage or increases data that can be recorded
- Modern FPGAs offer enough resources for ML applications
  - DSP blocks ideal for multiply-accumulate operations
  - Specialized models with AI accelerators

Data center card

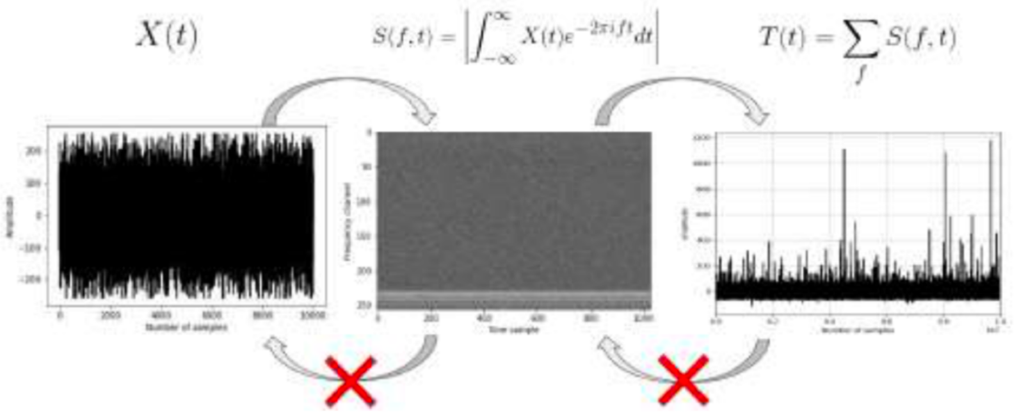
Development kit

Presentation by J. Voigt (TU Dresden)

ML on FPGAs – presentation: Pulsar data analysis

Analysis of pulsar data from radio telescope

Data formats



$X(t)$

$S(f,t) = \left| \int_{-\infty}^{\infty} X(t) e^{-2\pi i f t} dt \right|$

$T(t) = \sum_{f} S(f,t)$

Example data set (21 min):

Baseband data (1.6 TB)

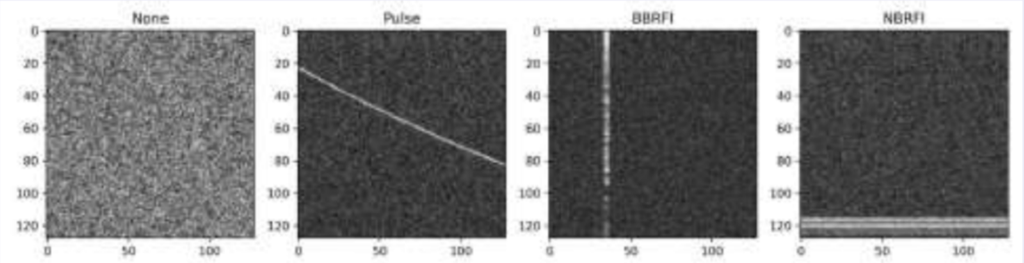
Spectrogram (3 GB)

Time series (49 MB)

- Goal: Only store baseband data when signal is detected
- Use minimal pre-processing
    - Train ANN to classify spectrograms

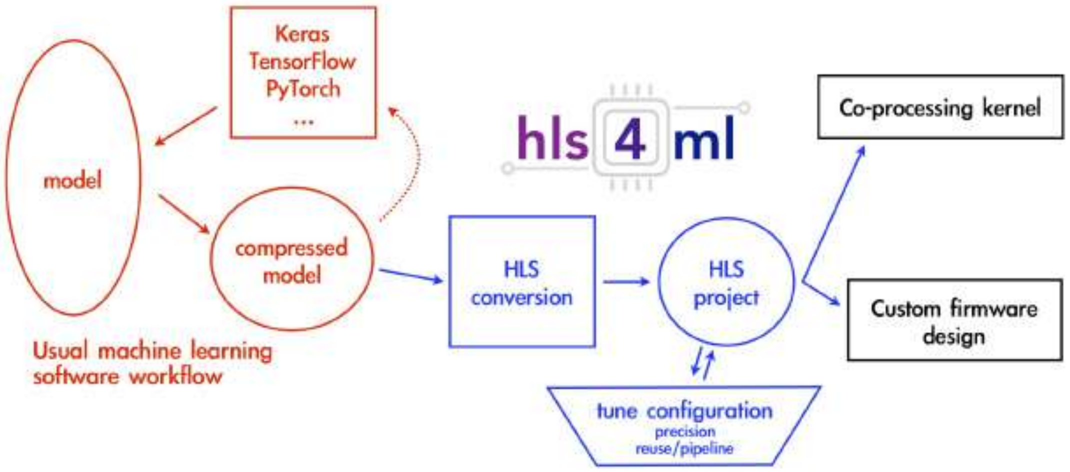
https://indico.desy.de/event/45348/contributions/173482/ [5]

8 classes of signal and RFI



Presentation by J. Voigt (TU Dresden)

ML on FPGAs – presentation: HLS4ML



Automatic conversion from trained models to HLS (High Level Synthesis) code

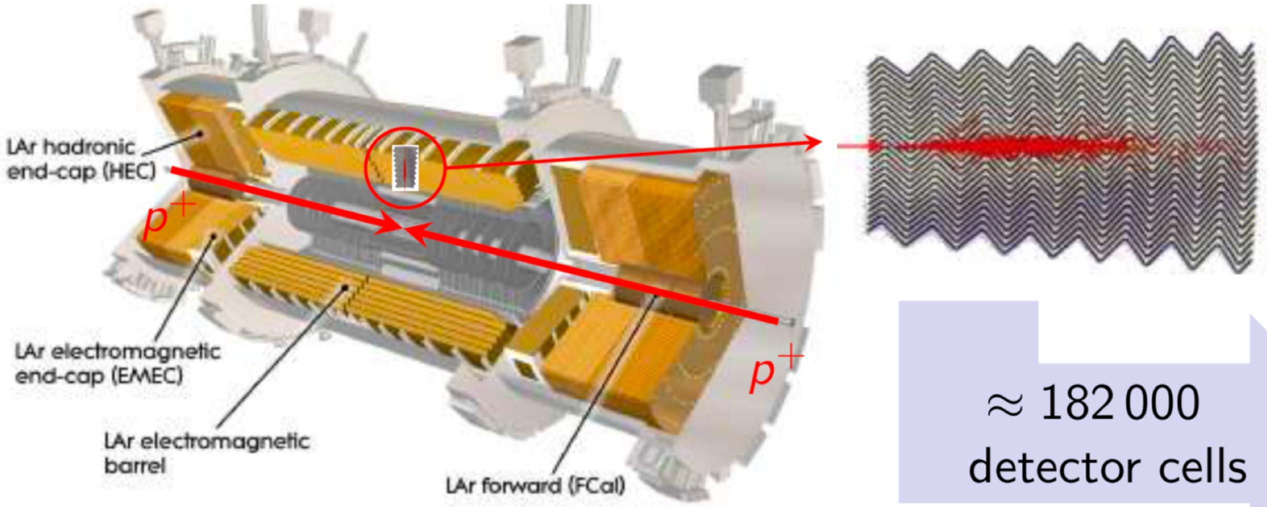
- Using FPGA as accelerator:
  - Host CPU manages the process
  - Shared High Bandwidth Memory (HBM) to transfer data to FPGA
  - AI kernel on FPGA (and data stream management)
  - Transfer results back via HBM
- AI kernel generated using hls4ml
- Tested with simplified example network: 2D CNN with 4572 parameters, results match expectation (can be tested out at `https://github.com/ypmen/punch_workshop` [7])
- Latency estimate ≈ 1 ms with 72 MHz maximum clock frequency

Presentation by J. Voigt (TU Dresden)

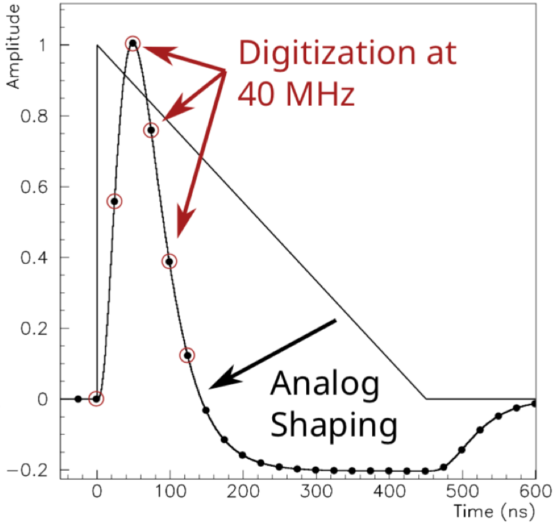
ML on FPGAs – presentation: ATLAS high trigger rates

ATLAS LAr calorimeter

- Upgraded Large Hadron Collider will provide ≈ 140 proton-proton collisions per bunch crossing (BC) ≙ every 25 ns ≙ 40 MHz
- ATLAS experiment detects collision products using layered sub-detectors
- Higher pileup and higher trigger rate require replacement of LAr calorimeter electronics
- 235 Tbit s^{-1} data stream from LAr calorimeters alone

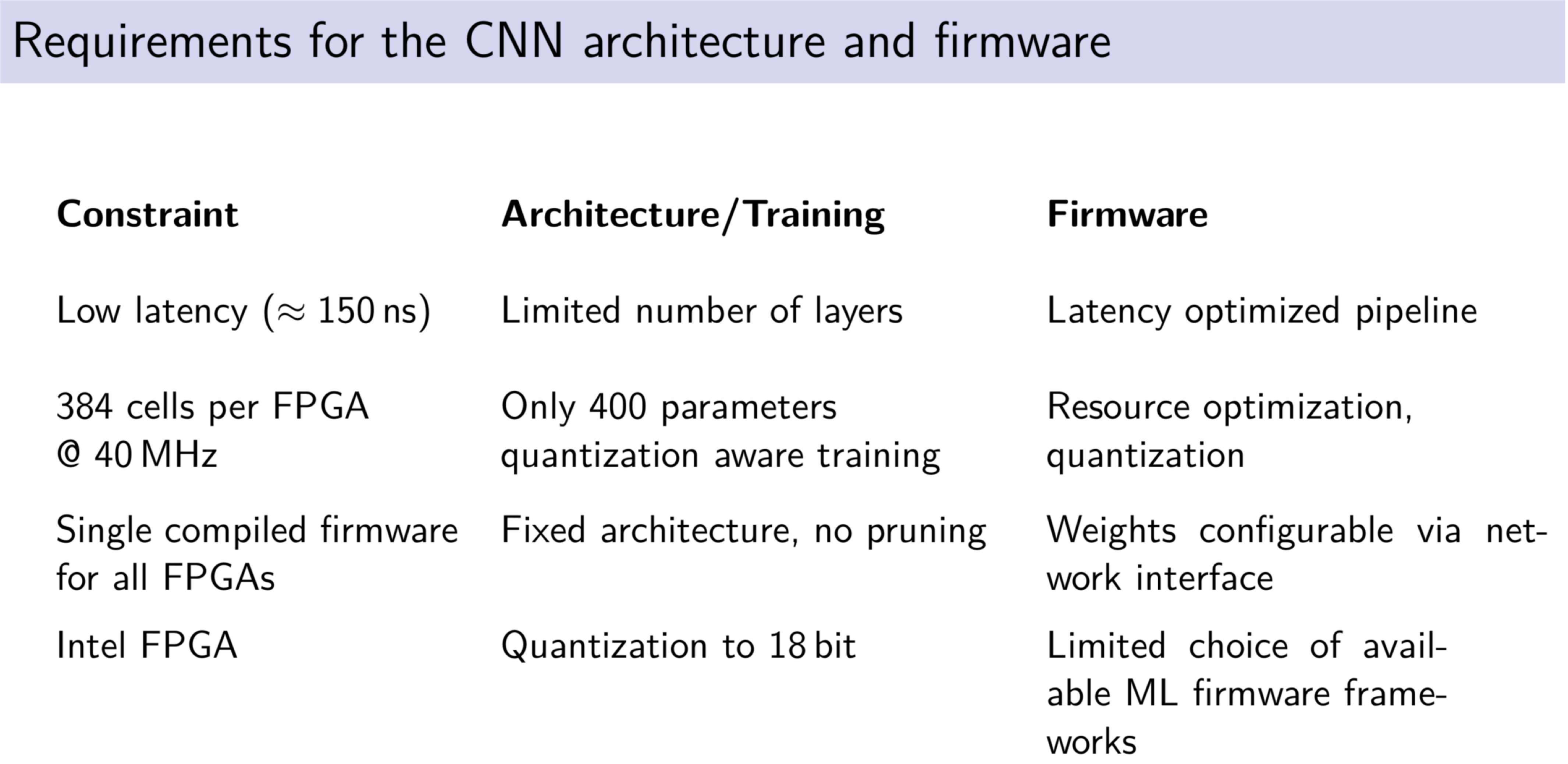

LAr hadronic end-cap (HEC)
p^+
LAr electromagnetic end-cap (EMEC)
LAr electromagnetic barrel
LAr forward (FCal)
p^+

≈ 182 000 detector cells



Presentation by J. Voigt (TU Dresden)

ML on FPGAs – presentation: Complex requirements

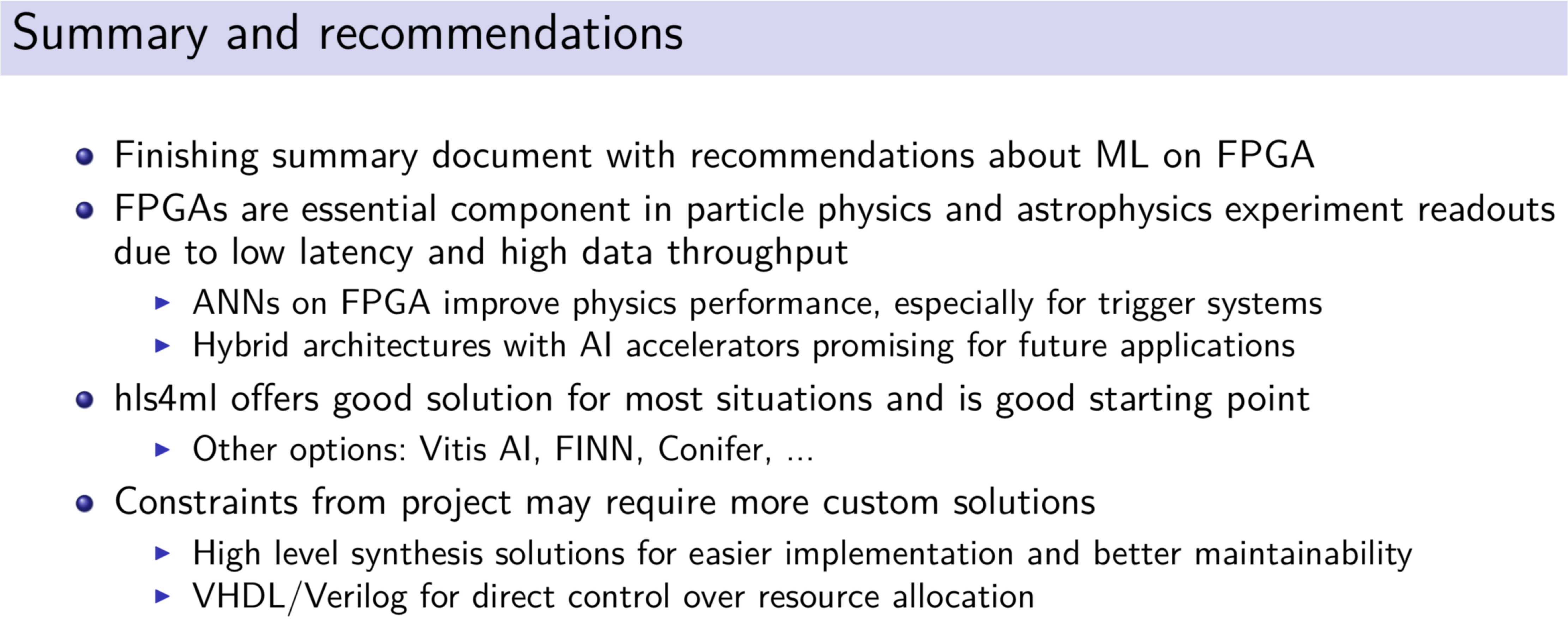


Requirements for the CNN architecture and firmware

Constraint	Architecture/Training	Firmware
Low latency ($\approx$ 150 ns)	Limited number of layers	Latency optimized pipeline
384 cells per FPGA @ 40 MHz	Only 400 parameters quantization aware training	Resource optimization, quantization
Single compiled firmware for all FPGAs	Fixed architecture, no pruning	Weights configurable via network interface
Intel FPGA	Quantization to 18 bit	Limited choice of available ML firmware frameworks

Presentation by J. Voigt (TU Dresden)

ML on FPGAs – presentation: Recommendations



Summary and recommendations

- Finishing summary document with recommendations about ML on FPGA
- FPGAs are essential component in particle physics and astrophysics experiment readouts due to low latency and high data throughput
    - ANNs on FPGA improve physics performance, especially for trigger systems
    - Hybrid architectures with AI accelerators promising for future applications
- hls4ml offers good solution for most situations and is good starting point
    - Other options: Vitis AI, FINN, Conifer, ...
- Constraints from project may require more custom solutions
    - High level synthesis solutions for easier implementation and better maintainability
    - VHDL/Verilog for direct control over resource allocation

Presentation by J. Voigt (TU Dresden)

Efficient caching - developments

Prototype setup for PUNCH users:
Improvements for storing frequently accessed files closer to the compute environment, thereby improving data locality and throughput

Analyses on distributed data, repeated file accesses also for validations in time-critical workflows

Deliverable: D-TA5-WP4-3 (31 Mar 2025): Caching strategies for processing a set of benchmark files with the evaluated efficiencies and latencies

XCache addresses I/O bottlenecks in data-intensive workflows by reducing latency for repeated file access.
It acts as a caching proxy for **XRootD-based** data sources

Apptainer (formerly Singularity) provides a containerized execution layer to encapsulate XCache environment, ensuring:

- Dependency isolation
- Environment reproducibility
- Portability across HPC and cluster nodes

→ enabling deployment of complex caching setups without polluting the host system or requiring root privileges.

Work by G. Dange (FIAS)

Efficient caching - developments

Created **container with definition file**
Used apptainer to build xcache.sif and xrootd.def to compile and install XRootD and XCache from source with installed dependencies – complete setup using xrooot.def script
Writeable mode for apptainer

Initial tests successful:
Validating XCache behaviour with CERN Open Data (public CMS data sets from eospublic.cern.ch)
→ Run xrdfs localhost query stats to retrieve key metrics: cache.hits, cache.misses, and cache size used
→ Acceleration of file access observed

Performance results and benchmarks – until September

Future work: Adding authentication plugins and secure configurations

Work by G. Dange (FIAS)

Other results and outlook

Forwarding/promoting recent results of TA5 (next meeting July 17)

More deliverable documents to be prepared

Activities for many deliverables for the remaining time of PUNCH 1.0 – critical is work for WP5

Paper on classical theory of optimal detection in context of radio astronomy in preparation

Towards PUNCH 2.0: Possible collaboration with NFDI4DS on Fast ML?