FWXMachina on FPGA for L1 trigger

1. Classification with boosted decision trees
2. Regression with deep boosted decision trees
3. Anomaly detection with decision tree-based autoencoder
   ↳ https://indico.cern.ch/e/1283970/contributions/5554363/
   Lightning talk by Steve Roche

Tae Min Hong*
University of Pittsburgh

Ben Carlson
Westmont

Stephen Roche
Saint Louis University School of Medicine

Fast Machine Learning for Science
September 25, 2023
https://indico.cern.ch/e/1283970/contributions/5554356/

FWXMachina Papers

Classification

JINST — Published by IOP Publishing for Sissa Medialab
Received: April 9, 2021
Accepted: June 29, 2021
Published: August 4, 2021

Nanosecond machine learning event classification with boosted decision trees in FPGA for high energy physics

T.M. Hong,* B.T. Carlson, B.R. Eubanks, S.T. Racz, S.T. Roche, J. Stelzer and D.C. Stumpp
Department of Physics and Astronomy, University of Pittsburgh,
100 Allen Hall, 3941 O'Hara St., Pittsburgh, PA 15260, U.S.A.
E-mail: tmhong@pitt.edu

Abstract: We present a novel implementation of classification using the machine learning/artificial intelligence method called boosted decision trees (BDT) on field programmable gate arrays (FPGA). The firmware implementation of binary classification requiring 100 training trees with a maximum depth of 4 using four input variables gives a latency value of about 10 ns, independent of the clock speed from 100 to 320 MHz in our setup. The low timing values are achieved by restructuring the BDT layout and reconfiguring its parameters. The FPGA resource utilization is also kept low at a range from 0.01% to 0.2% in our setup. A software package called fwXmachina achieves this implementation. Our intended user is an expert in custom electronics-based trigger systems in high energy physics experiments or anyone that needs decisions at the lowest latency values for real-time event classification. Two problems from high energy physics are considered, in the separation of electrons vs. photons and in the selection of vector boson fusion-produced Higgs bosons vs. the rejection of the multijet processes.

Keywords: Digital electronic circuits; Trigger algorithms; Trigger concepts and systems (hardware and software); Data reduction methods

ArXiv ePrint: 2104.03408

*Corresponding author.

 https://doi.org/10.1088/1748-0221/16/08/P08016

2021 JINST 16 P08016

Hong et al., JINST **16**, P08016 (2021)
http://doi.org/10.1088/1748-0221/16/08/P08016

Regression + deep

JINST — Published by IOP Publishing for Sissa Medialab
Received: July 13, 2022
Accepted: August 23, 2022
Published: September 27, 2022

Nanosecond machine learning regression with deep boosted decision trees in FPGA for high energy physics

B.T. Carlson,[a,b] Q. Bayer,[b] T.M. Hong[b,*] and S.T. Roche[b]
[a] *Department of Physics and Engineering, Westmont College, 955 La Paz Road, Santa Barbara, CA 93108, U.S.A.*
[b] *Department of Physics and Astronomy, University of Pittsburgh, 100 Allen Hall, 3941 O'Hara St., Pittsburgh, PA 15260, U.S.A.*
E-mail: tmhong@pitt.edu

Abstract: We present a novel application of the machine learning / artificial intelligence method called boosted decision trees to estimate physical quantities on field programmable gate arrays (FPGA). The software package fwXmachina features a new architecture called parallel decision paths that allows for deep decision trees with arbitrary number of input variables. It also features a new optimization scheme to use different numbers of bits for each input variable, which produces optimal physics results and ultraefficient FPGA resource utilization. Problems in high energy physics of proton collisions at the Large Hadron Collider (LHC) are considered. Estimation of missing transverse momentum ($E_{\mathrm{T}}^{\mathrm{miss}}$) at the first level trigger system at the High Luminosity LHC (HL-LHC) experiments, with a simplified detector modeled by Delphes, is used to benchmark and characterize the firmware performance. The firmware implementation with a maximum depth of up to 10 using eight input variables of 16-bit precision gives a latency value of $\mathcal{O}(10)$ ns, independent of the clock speed, and $\mathcal{O}(0.1)$% of the available FPGA resources without using digital signal processors.

Keywords: Data reduction methods; Digital electronic circuits; Trigger algorithms; Trigger concepts and systems (hardware and software)

ArXiv ePrint: 2207.05602

*Corresponding author.

 https://doi.org/10.1088/1748-0221/17/09/P09039

2022 JINST 17 P09039

Carlson et al., JINST **17**, P09039 (2022)
http://doi.org/10.1088/1748-0221/17/09/P09039

Anomaly detection

arXiv:2304.03836v1 [hep-ex] 7 Apr 2023

PITT-PACC-2311

Nanosecond anomaly detection with decision trees for high energy physics and real-time application to exotic Higgs decays

S.T. Roche[a,b], Q. Bayer[b], B.T. Carlson[b,c], W.C. Ouligian[b], P. Serhiayenka[b], J. Stelzer[b], and T.M. Hong*[b]
[a]School of Medicine, Saint Louis University
[b]Department of Physics and Astronomy, University of Pittsburgh
[c]Department of Physics and Engineering, Westmont College

April 11, 2023

Abstract

We present a novel implementation of the artificial intelligence autoencoding algorithm, used as an ultrafast and ultraefficient anomaly detector, built with a forest of deep decision trees on FPGA, field programmable gate arrays. Scenarios at the Large Hadron Collider at CERN are considered, for which the autoencoder is trained using known physical processes of the Standard Model. The design is then deployed in real-time trigger systems for anomaly detection of new unknown physical processes, such as the detection of exotic Higgs decays, on events that fail conventional threshold-based algorithms. The inference is made within a latency value of 25 ns, the time between successive collisions at the Large Hadron Collider, at percent-level resource usage. Our method offers anomaly detection at the lowest latency values for edge AI users with tight resource constraints.

Keywords: Data processing methods, Data reduction methods, Digital electronic circuits, Trigger algorithms, and Trigger concepts and systems (hardware and software).

*Corresponding author, tmhong@pitt.edu

1

Roche et al., submitted for publication
https://arxiv.org/abs/2304.03836

FWX Machina

Outline

- ## Motivation

  Why fast
  Why small
  ML on FPGA .... *Decision Trees vs. Neural Networks*

- ## FPGA implementation

  Parallelize cuts .... *Classification of VBF Higgs vs. multijets*
  Parallelize terminal bins .... *Deep regression of missing energy*
  Tree-based autoencoder .... *Anomaly detection*
  ↳ Lightning talk by S. Roche
  https://indico.cern.ch/e/1283970/contributions/5554363/

- ## Results

  Physics scenarios .... *Topics listed above*
  Comparisons .... *vs. hls4ml family of tools*
  Where to find more info .... *http://fwx.pitt.edu*

FWX Machina

Why fast

Dataflow at LHC

- L1 Trigger

  Design: Custom boards w/, e.g., Xilinx Virtex US+

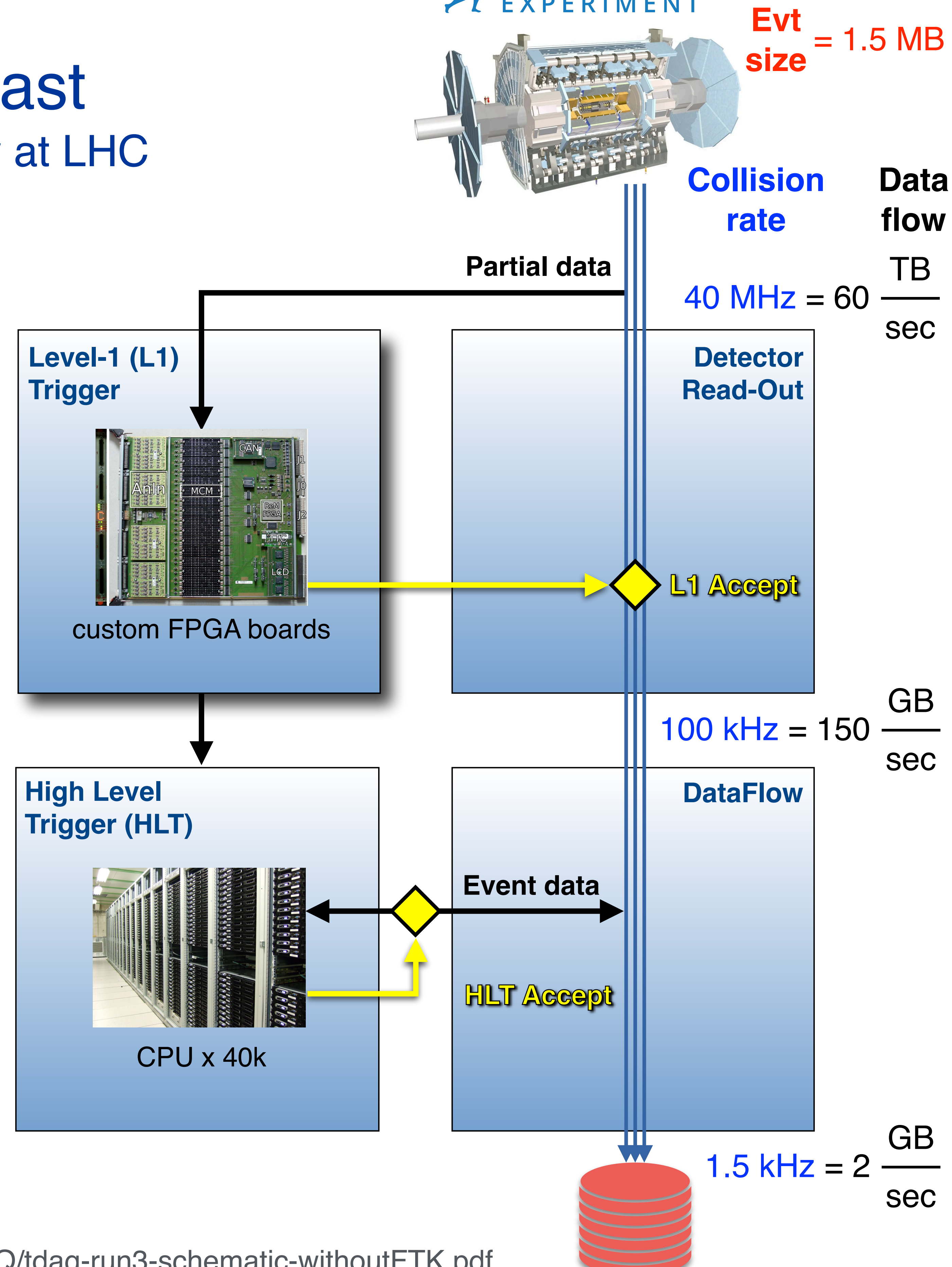
  Latency: Fixed at O(1) μs

  ML: Need fast firmware

- FWX

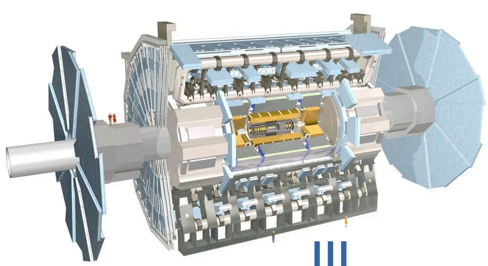
  Latency: O(10) ns FIFO

  Note: ML latency must fit between various pre- and post-processing, e.g., composite object creation from raw data



Modeled after https://cern.ch/twiki/pub/AtlasPublic/ApprovedPlotsDAQ/tdaq-run3-schematic-withoutFTK.pdf

Why small footprint
ATLAS for HL-LHC (2026)



- **L0 Trigger**
  - Design: Custom boards w/, e.g., Xilinx VP 1502
  - Avail.: 2M LUT, 7k DSP
  - ML: Need efficient fw
- **FWX**
  - Size: 1%-level footprint
  - Note: ML footprint must fit among various pre- and post-processing, e.g., composite object creation from raw data

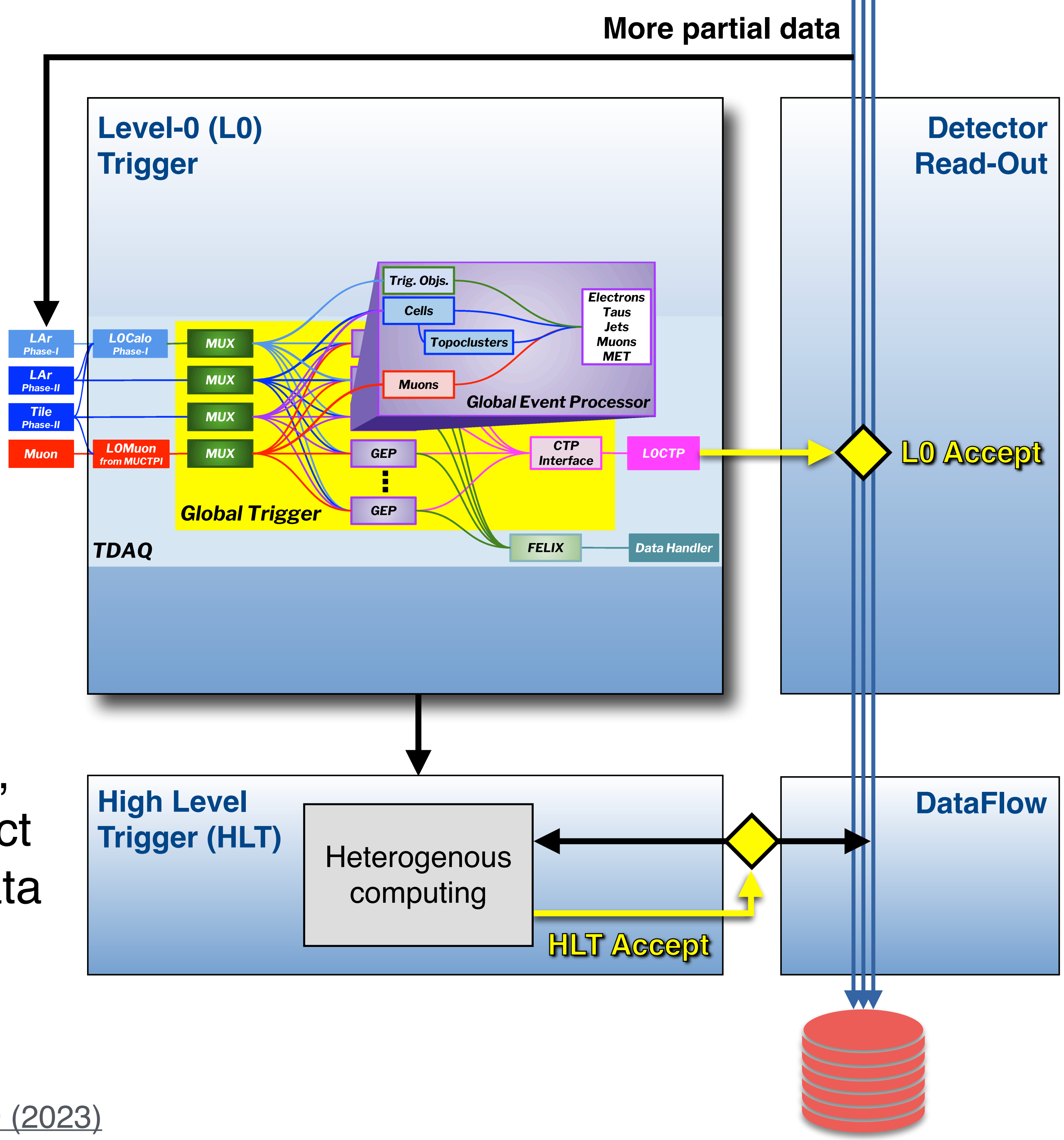


Diagram from Tang et al., IEEE Trans. on Nucl. Sci. 70, 9 (2023)

FWX Machina

ML on FPGA

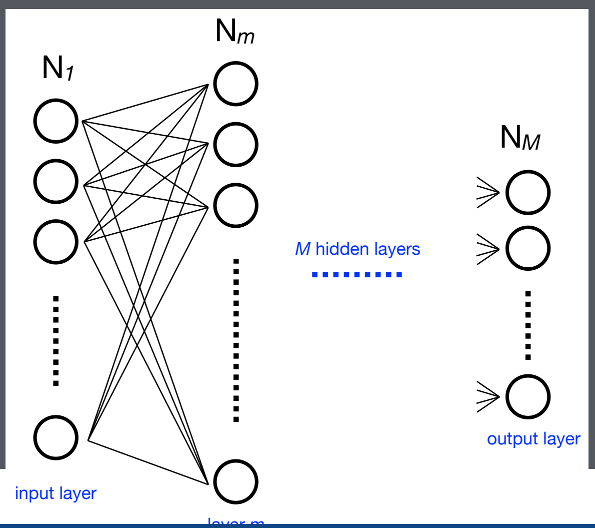
NN vs. DT

[1] Denby, Comp. Phys. Comm. 49-3, 429 (1988)
[2] Duarte et al., J. Instrum. 13, P07027 (2018)
[3] CMS Collaboration, Phys. Lett. B 716, 31 (2012)
[4] Summers et al., J. Instrum 15, P02056 (2020)
[5] Hong et al., J. Instrum. 16, P08016 (2021)
[6] Carlson et al., J. Instrum. 17, P09039 (2022)

- **Neural Network**

Popular	Been around HEP since the 80s[1]
Depth	Challenging, so ~3 on FPGA[2]
Score	$y = \Theta(\mathbb{M} \cdot x + b)$

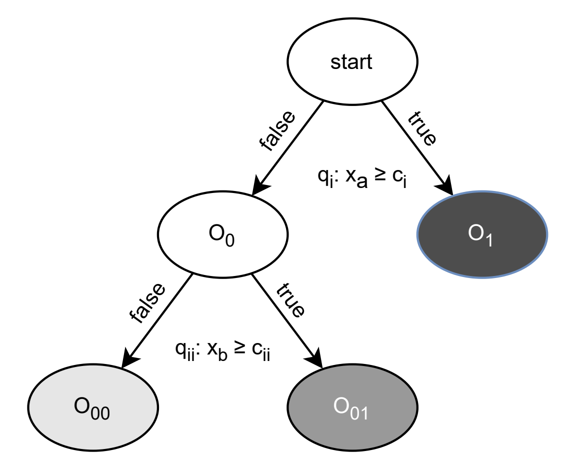
Activation *Multiplication*



- **Decision Tree**

Popular	Discovered the Higgs![3]
Depth	Challenging, so 4 to 8 on FPGA[4,5,6]
Score	$y = \Theta(x <$ threshold$)$

Step fn *Comparison*



- **FWX Decision Tree**

Physics	Comparable results vs. NN on FPGA
~~Float / fixed~~	Bit integer → bit shifts → efficient
Optimized	Parallelize → one step → low latency

FWX Machina

Design v1: Parallelize cuts

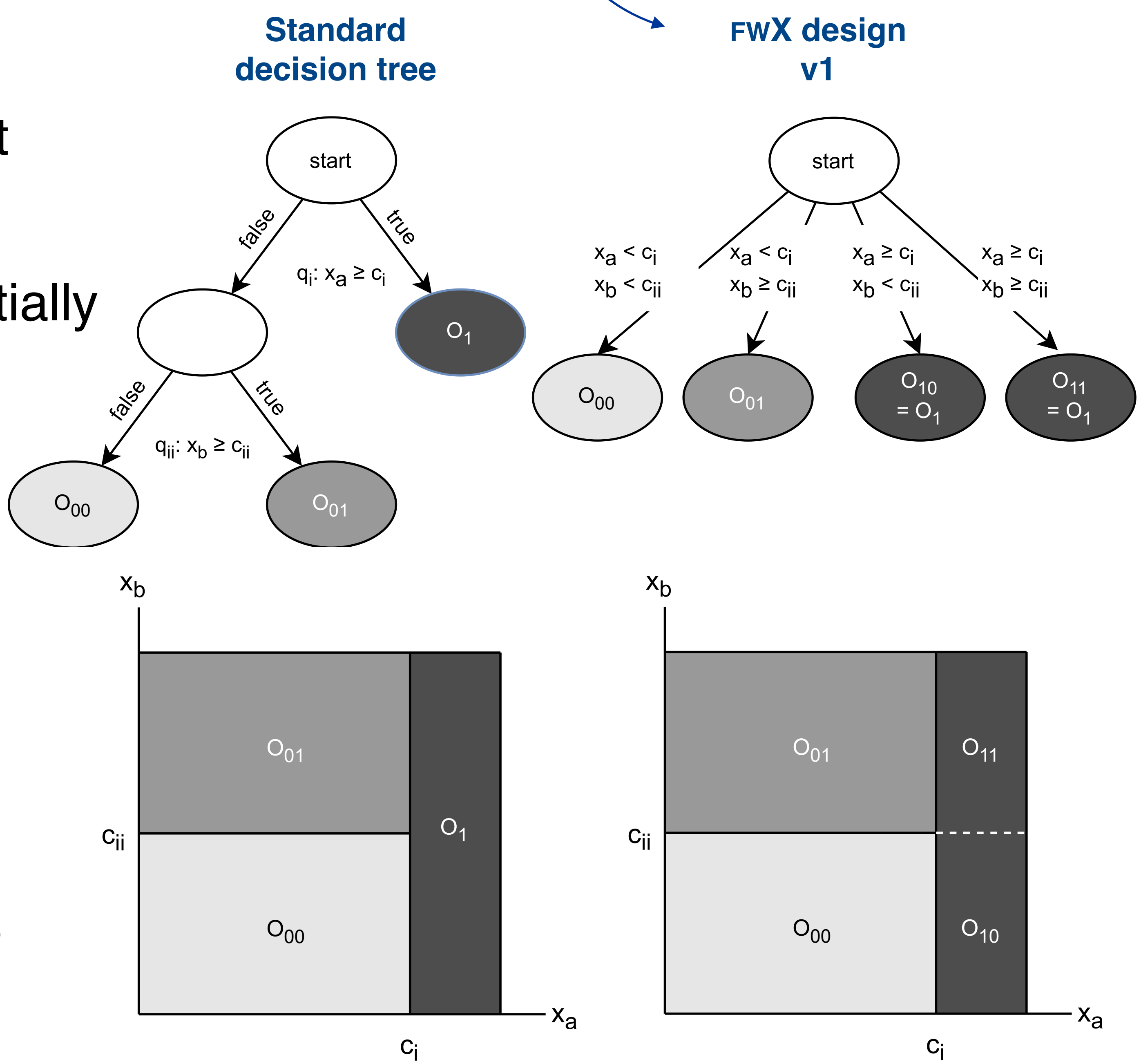
2 variable example

- General

Training	Use TMVA or equivalent
Design	Threshold comparisons
Challenge	Evaluate layers sequentially

- FWXv1

Key design	Evaluate cuts in parallel
Benefit	Each cut is indpd’t → Bin search on a grid → Bit shift to speed-up
Limitations	Does not scale well w/ tree depth & # variables
Follow-up	Led to v2 design



Hong et al., JINST 16, P08016 (2021)

Design v2: Parallelize terminal bins

Go deeper from 4 → 8

- Improve FWXv1

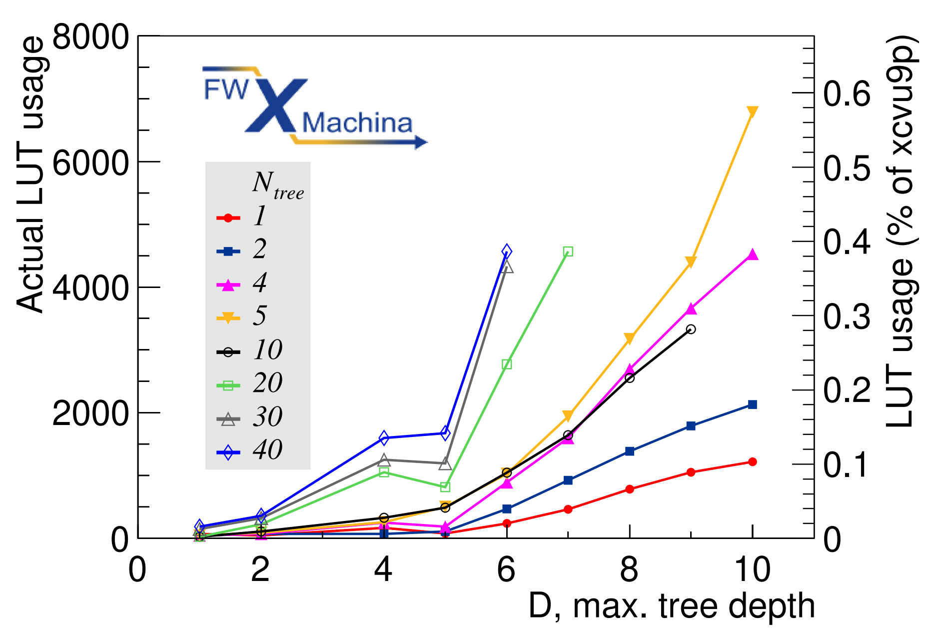
Challenge — Does not scale well w/ tree depth & # variables

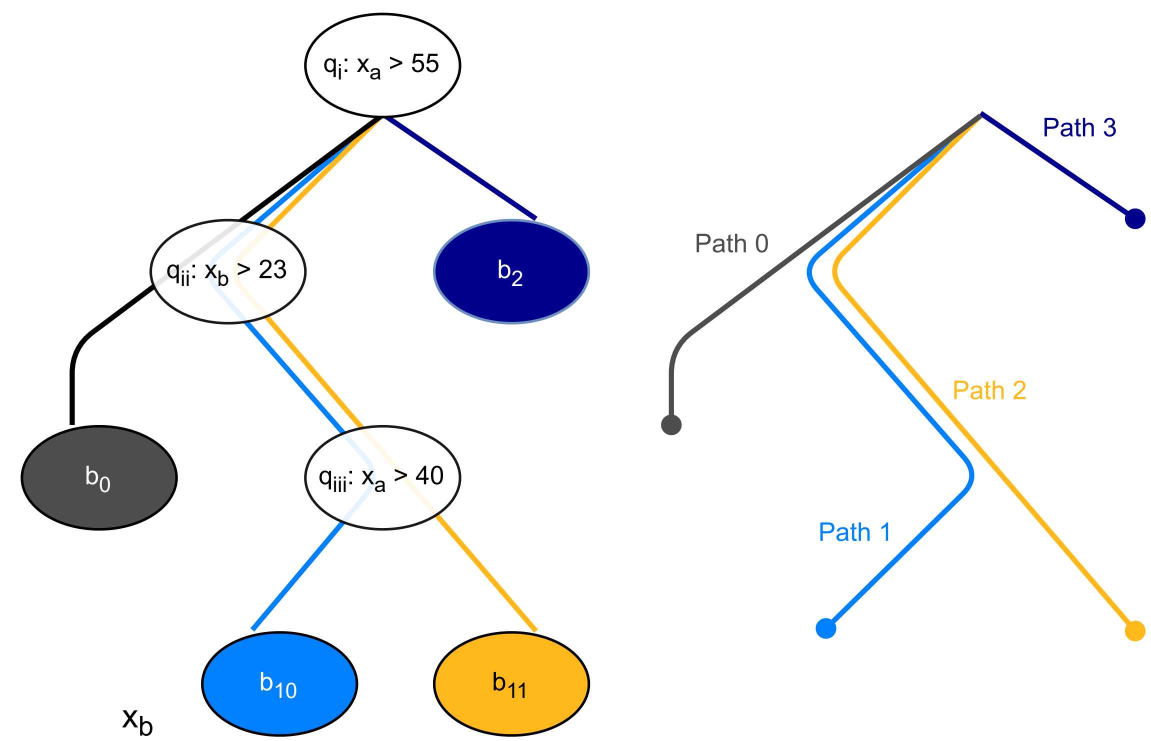
Cut redundancy 2^D

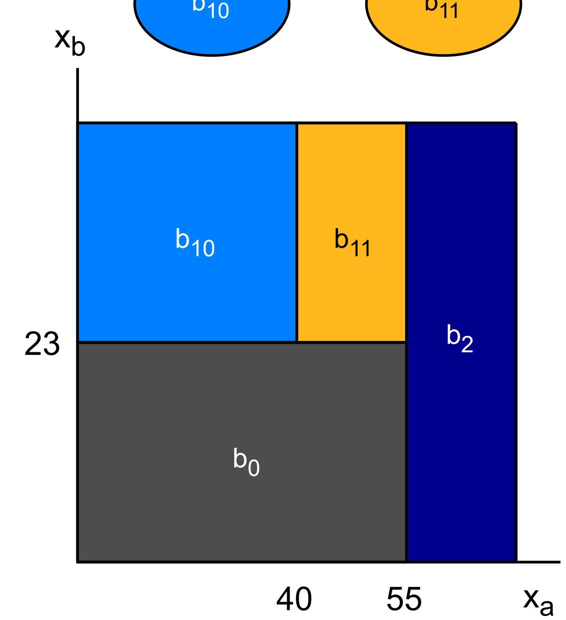
- FWXv2

Key design — Evaluate decision paths

Benefit — Softer scaling vs 2^D







Destination bin	Decision path
b_0	not(q_i) and not(q_{ii})
b_2	q_i
b_{10}	not(q_i) and q_{ii} and not(q_{iii})
b_{11}	not(q_i) and q_{ii} and q_{iii}

Carlson et al., JINST 17, P09039 (2022)

FWX Machina

Tree-based autoencoder

For anomaly detection

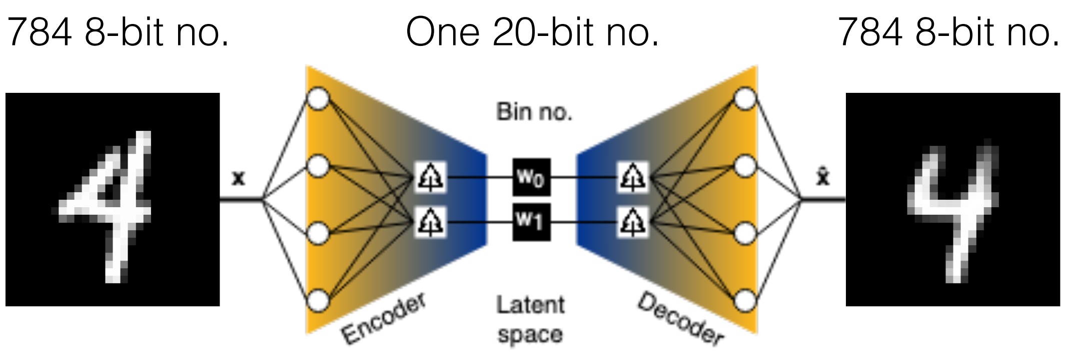
- **Autoencoder**

Design	Goal is to reproduce input
Challenge	Not many training methods
Re-use	FWX engine + distance

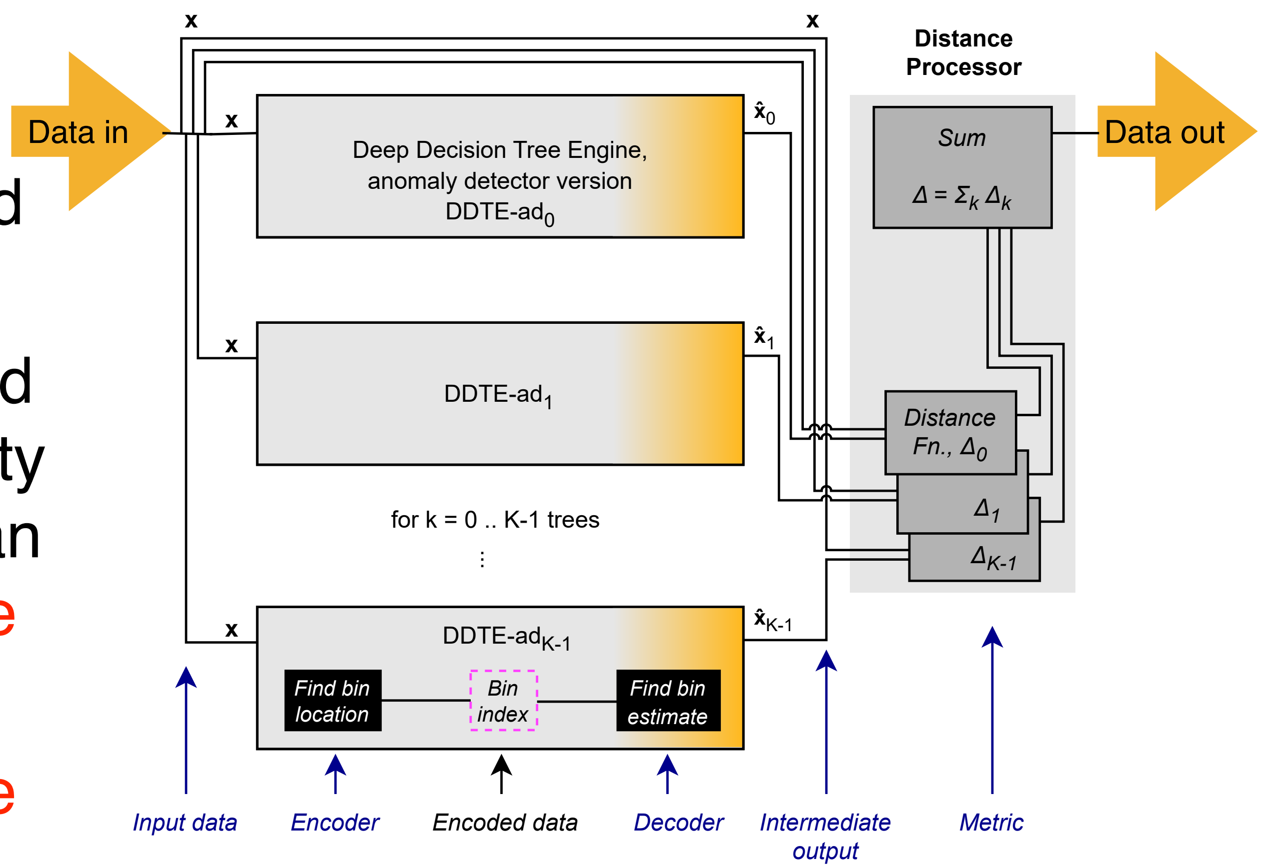
- **FWXAE**

Benefit	Latent data is retrievable, but can skip if speed desired → Direct input-to-output
Training	We created in-house method based on input-space density estimation by sample median ↳ lightning talk by S. Roche
Results	Comparison vs. hls4ml, see ↳ lightning talk by S. Roche

MNIST dataset
for data reconstruction



Firmware block diagram
shows the latent layer that can be skipped or not



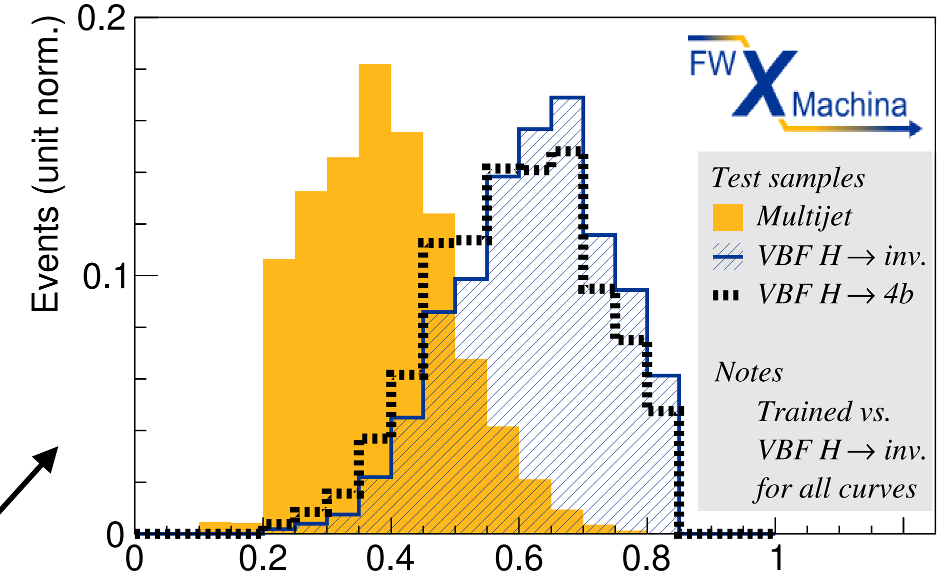
Roche et al., https://arxiv.org/abs/2304.03836

FWXMachina Results v1

VBF Higgs vs. Multijets

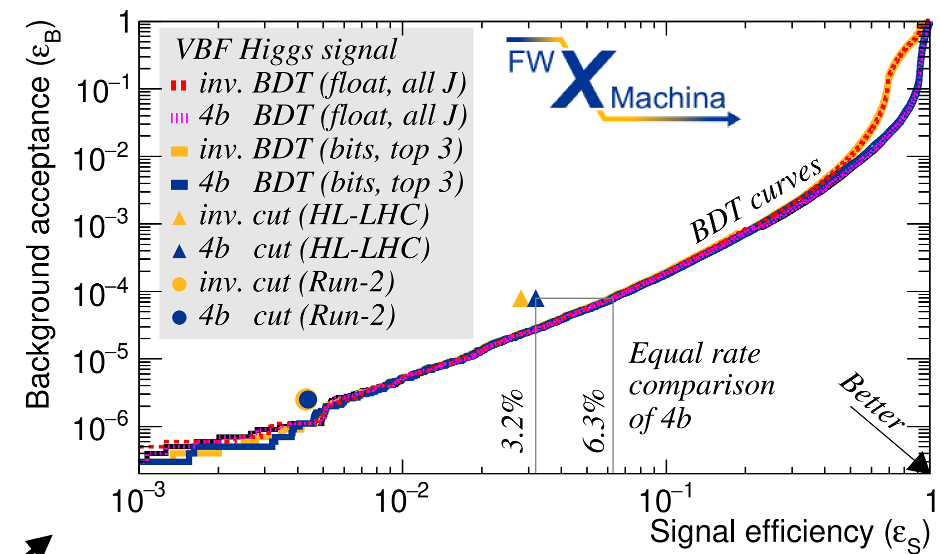
- **Setup**

Goal	Classification
Physics	VBF Higgs vs. Multijet
Inputs	5 variables re: 2 VBF jets
ML	BDT, 100 trees, 4 deep
Training	VBF Higgs → invisible
Test	VBF Higgs → 4b, works!



- **FWXv1**

Physics	2x better vs. ATLAS-inspired cuts
Timing	Latency of 16 ns (5 tick @330 MHz) Interval of 3 ns (1 tick @330 MHz)
FPGA usage	1% level or smaller



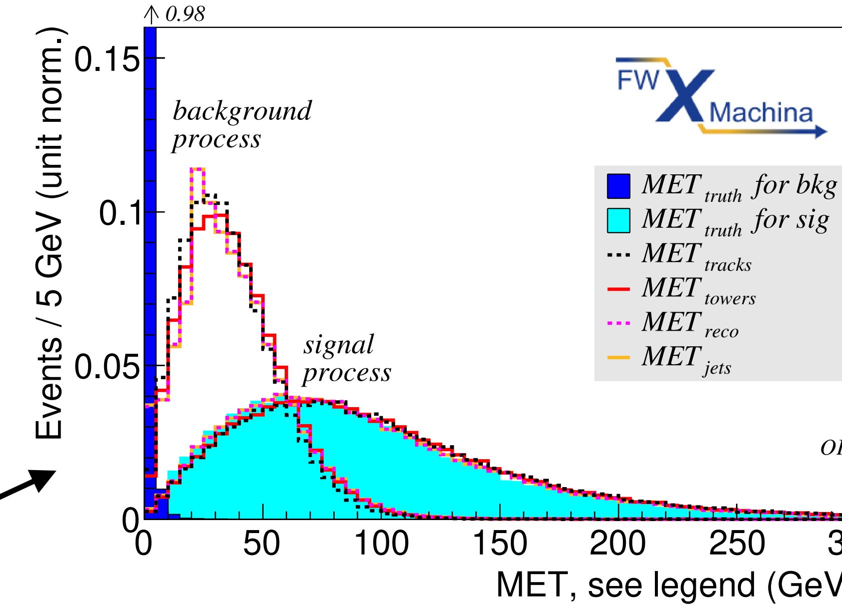
# bits for inputs	8
# bits for output	16
LUT	1%
Flip Flops	~ 0
BRAM	2%
DSP	0

Hong et al., JINST 16, P08016 (2021)

Results v2
Estimate MET

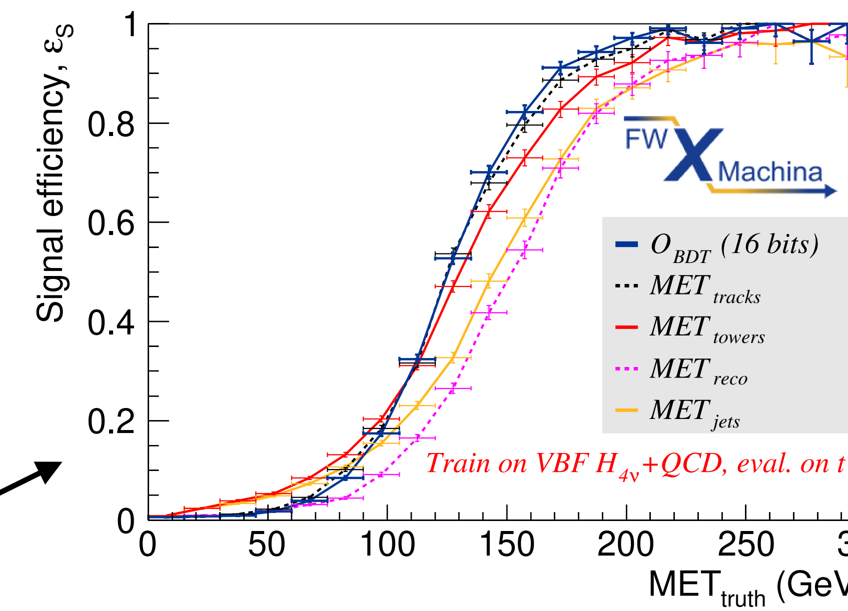
- **Setup**

Goal	Regression
Physics	Estimate MET at HL-LHC
Inputs	8 variables re: MET & rho
ML	BDT, 10 trees, 8 deep
Training	MET_{truth}



- **FWXv2**

Physics	Sharper turn-on curve
Timing	Latency of 65 ns (21 tick @330 MHz) Interval of 3 ns (1 tick @330 MHz)
FPGA usage	0.1% level or smaller

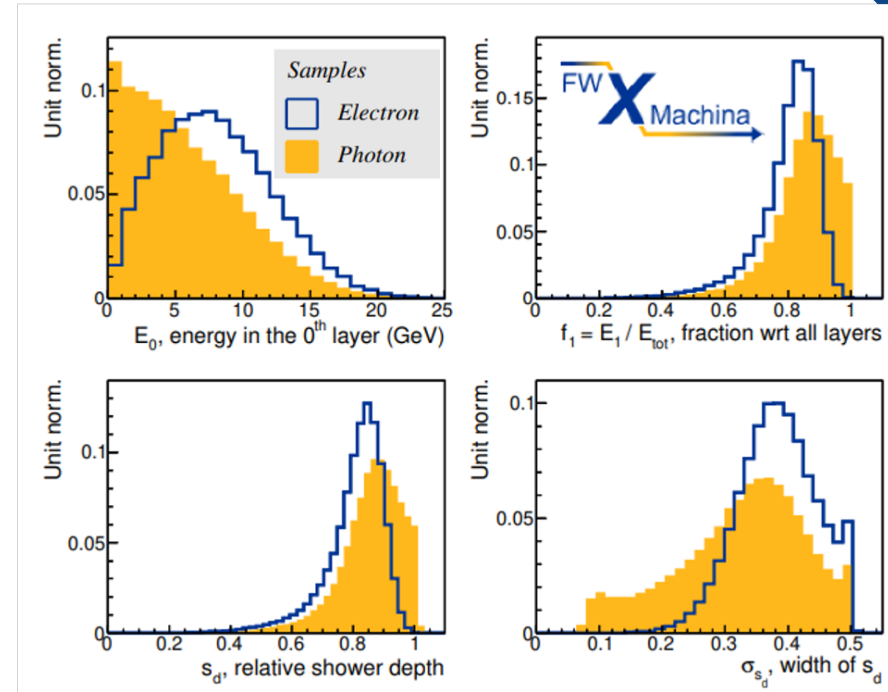


# bits for inputs	16
# bits for output	16
LUT	0.2%
Flip Flops	0.1%
BRAM	0.1%
DSP	0

Carlson et al., JINST 17, P09039 (2022)

FWX Machina Comparisons

vs. hls4ml family of tools

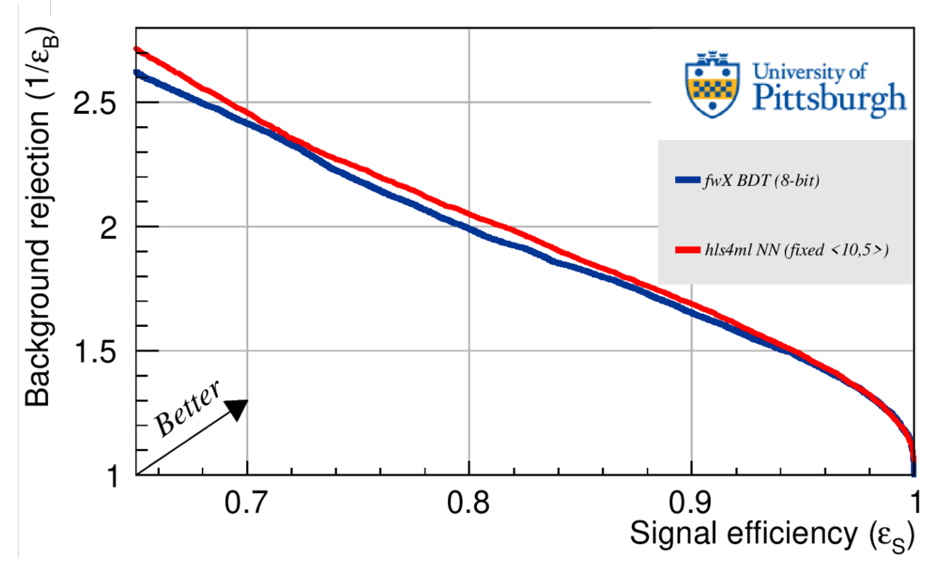


- **Setup**

Physics	4 variables for e vs. γ[1]
FWX BDT	100 trees, 4 deep
hls4ml BDT	” identical config for BDT
hls4ml NN	Out-of-the-box config

- **FWXv1**

vs. hls4ml NN	Comparable[2]
vs. hls4ml BDT	Same (since identical config)
Resource	< 1% for all methods
Latency	FWX’s parallel + no clocked operations



	FWX BDT v1	hls4ml NN	hls4ml conifer-BDT
# bits	〈 8 〉	〈 10, 5 〉	〈 10, 5 〉
LUT	0.06%	0.1%	0.3%
Flip Flops	0.01%	0.01%	0.1%
BRAM	0.1%	0.2%	0
DSP	0.03%	0.02%	0
Latency	10 ns	25 ns	47 ns
Interval	1 clock tick	1 clock tick	1 clock tick

- **FWXv2, FWXAE**

v2 vs. all	Need to do
AE vs. hls4ml	→ lightning talk S. Roche

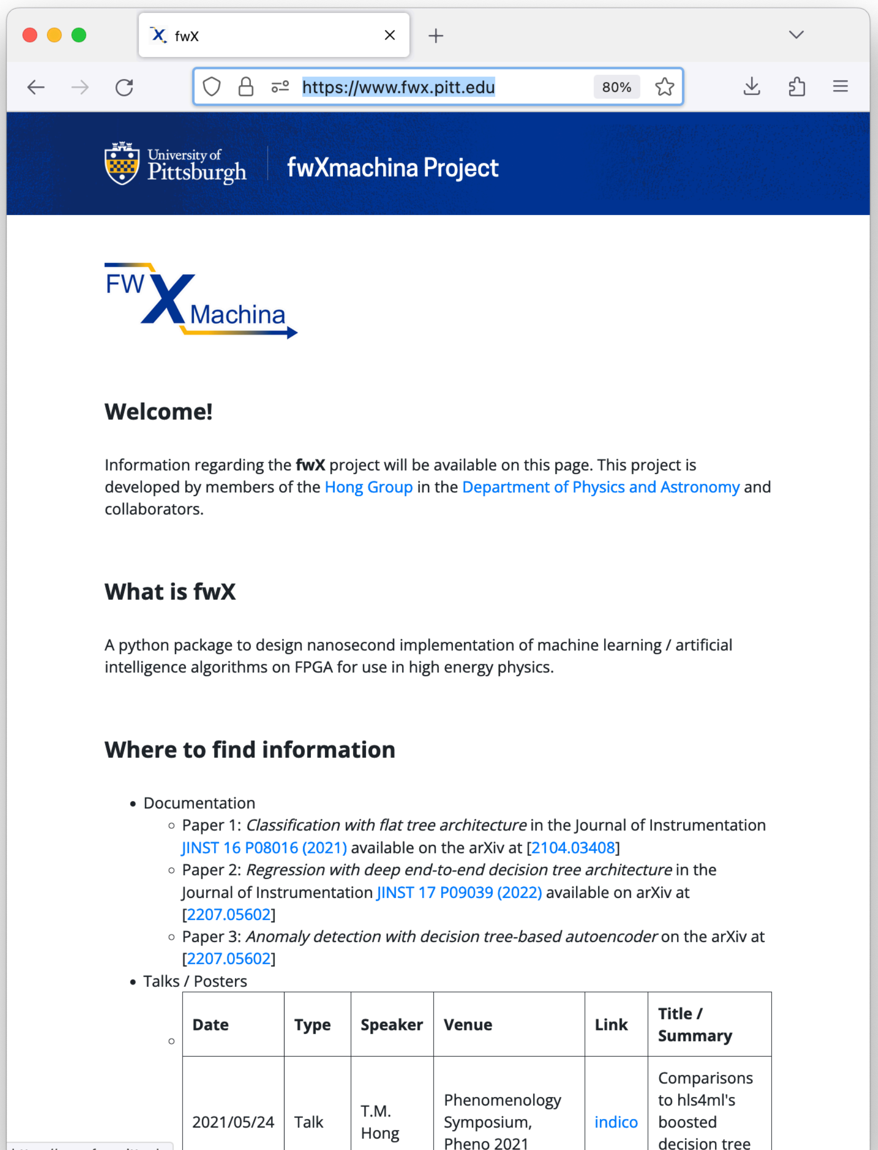
[1] Nachman et al., https://data.mendeley.com/datasets/kp3myh3v89/1
[2] Hong, PIKIMO 11, https://indico.cern.ch/event/1091676/contributions/4639362/

For more info

tmhong@pitt.edu

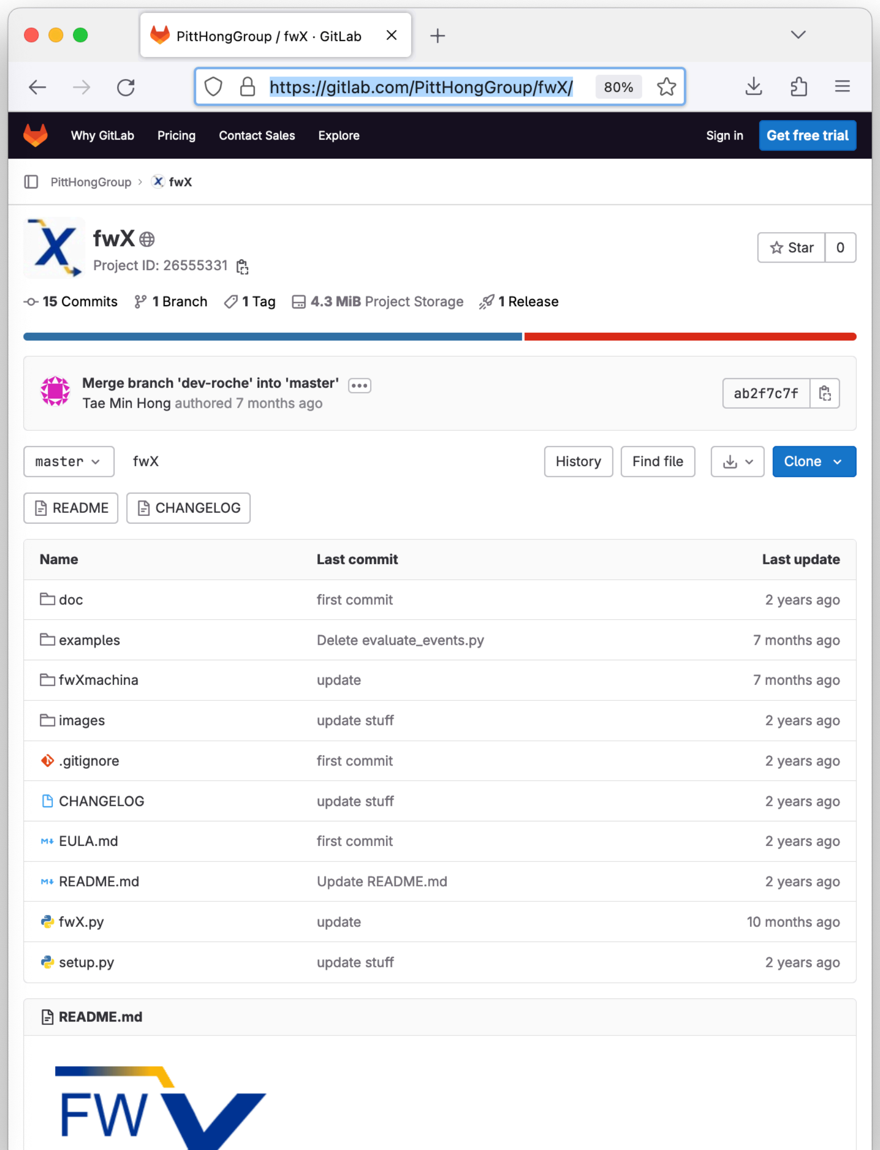
Webpage

http://fwx.pitt.edu



Code repository

gitlab.com/PittHongGroup/fwX/



Firmware testbench

d-scholarship.pitt.edu/44431/