Machine Learning for Heavy Flavor Identification

Cameron Dean

Massachusetts Institute of Technology

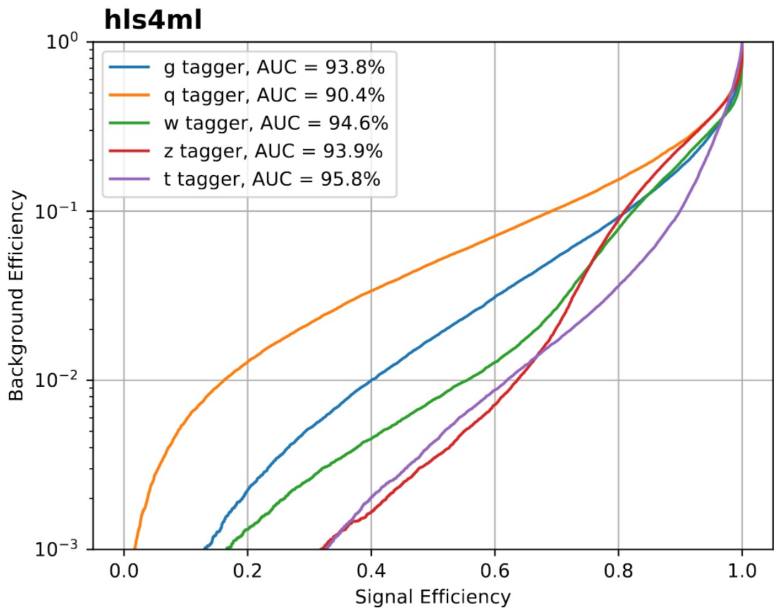
Second Workshop on Artificial Intelligence for the Electron Ion Collider

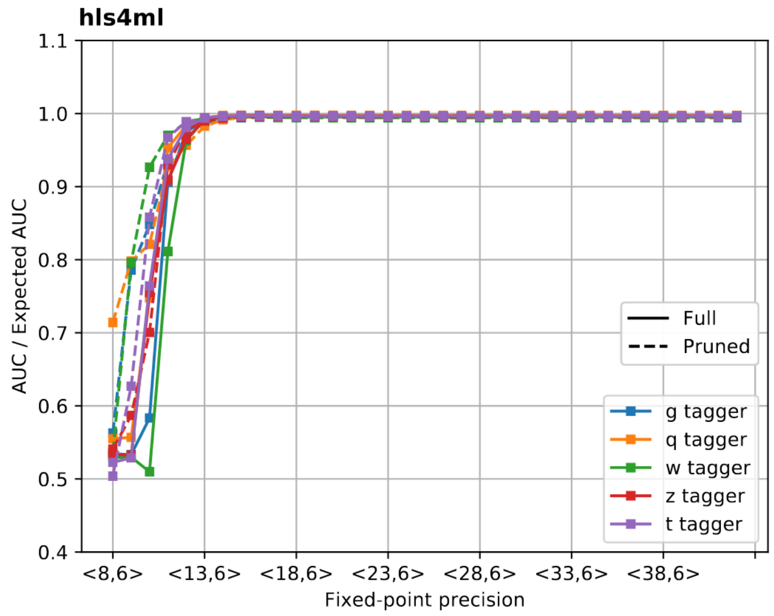
10/13/22

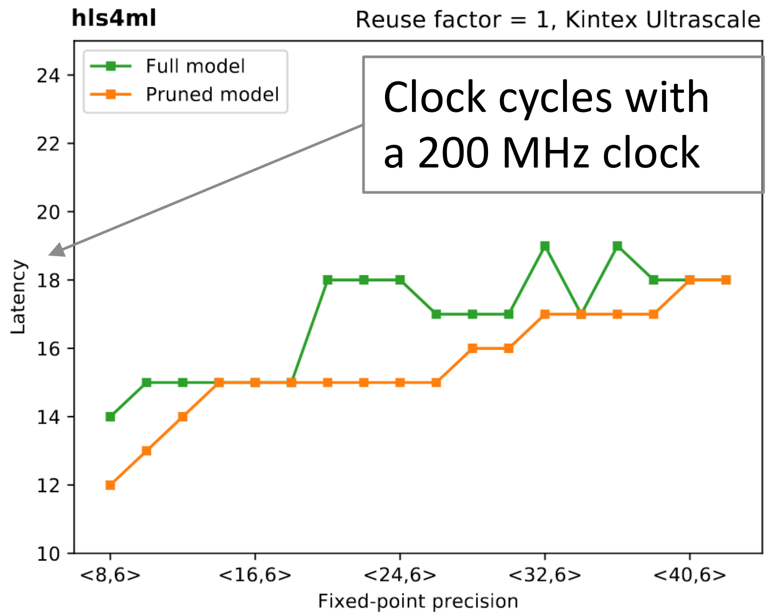
[JINST 13 P07027]

A brief flavor from the LHC

- ML is being applied to online selection at the LHC
  - CMS developed HF-jet taggers on FPGAs with 100ns latency



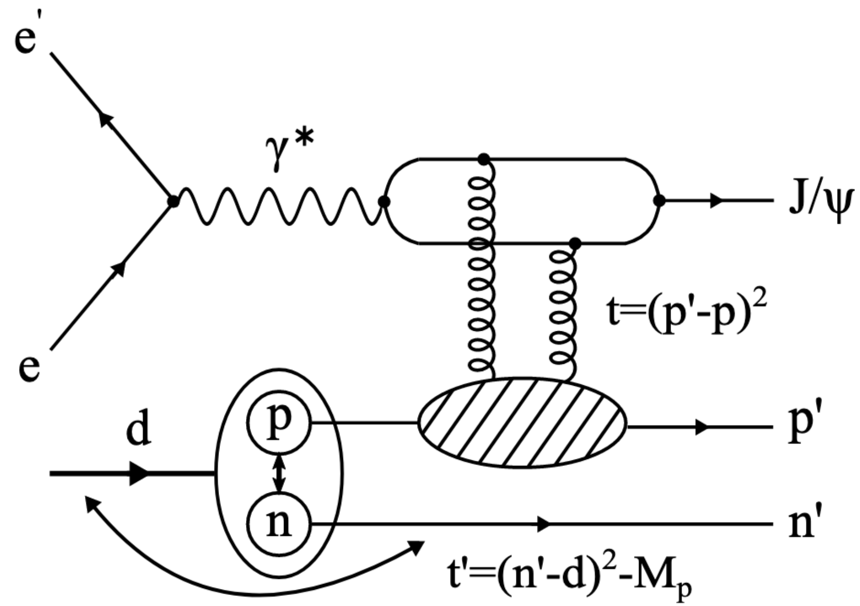


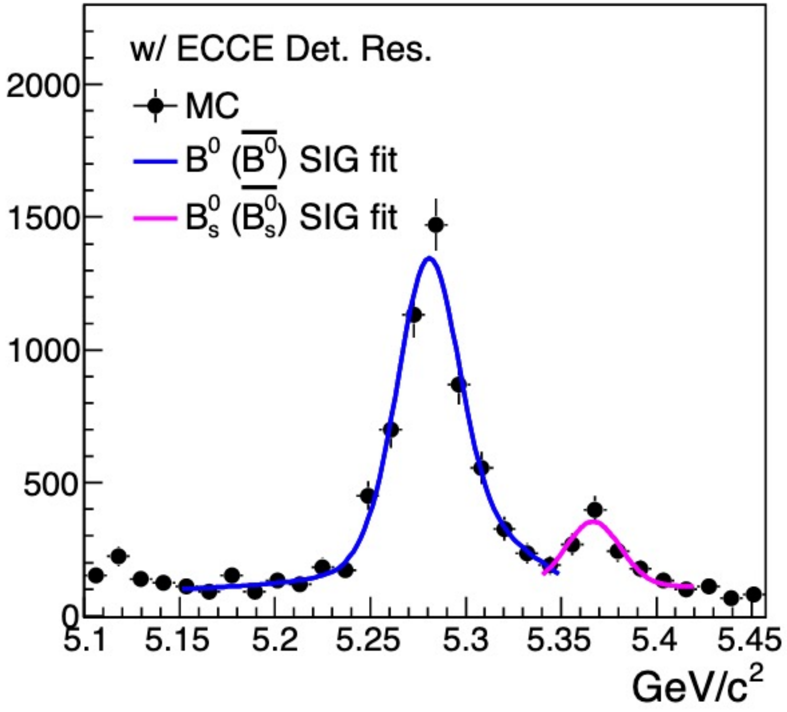


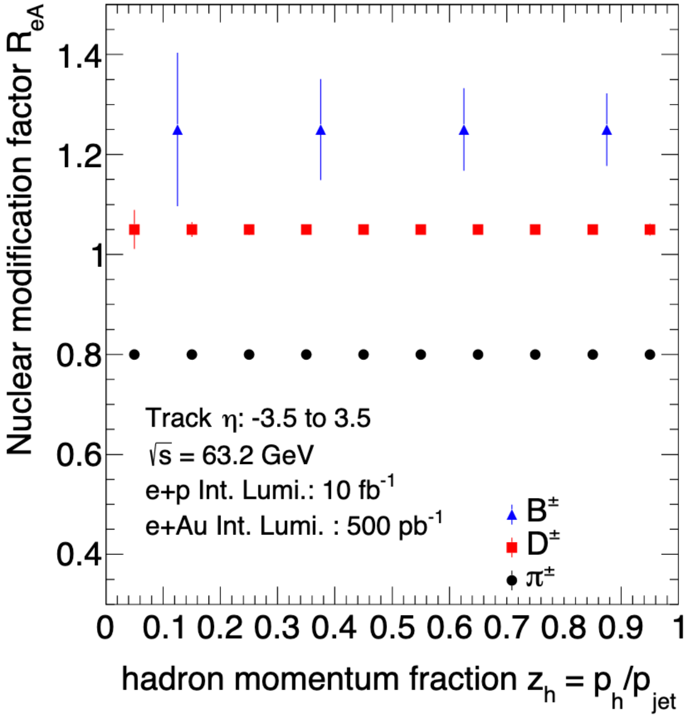
Heavy flavor at the EIC

[arXiv.2207.10632]
[arXiv:2103.05419]

- Why?
  - Main HF production is through photon-gluon processes
  - Good probe of gluon parton distribution function







The proposal

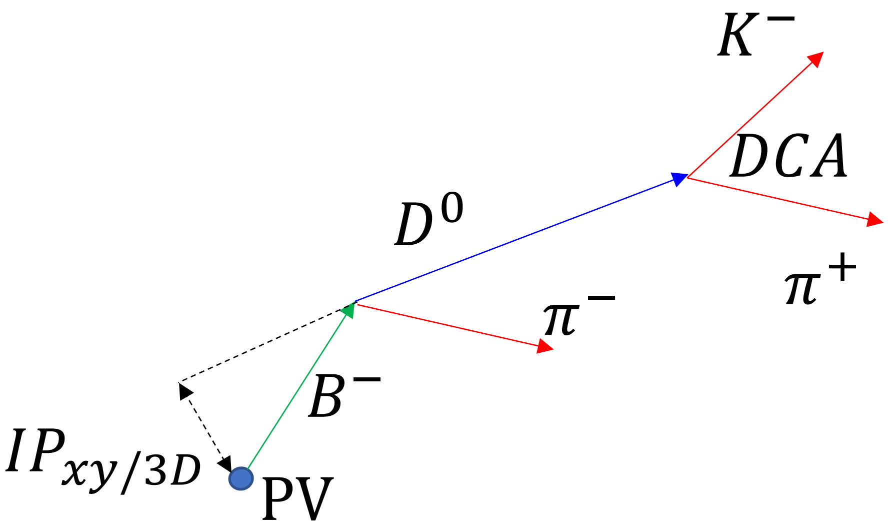
Intelligent experiments through real-time AI: Fast Data Processing and Autonomous Detector Control for sPHENIX and future EIC detectors

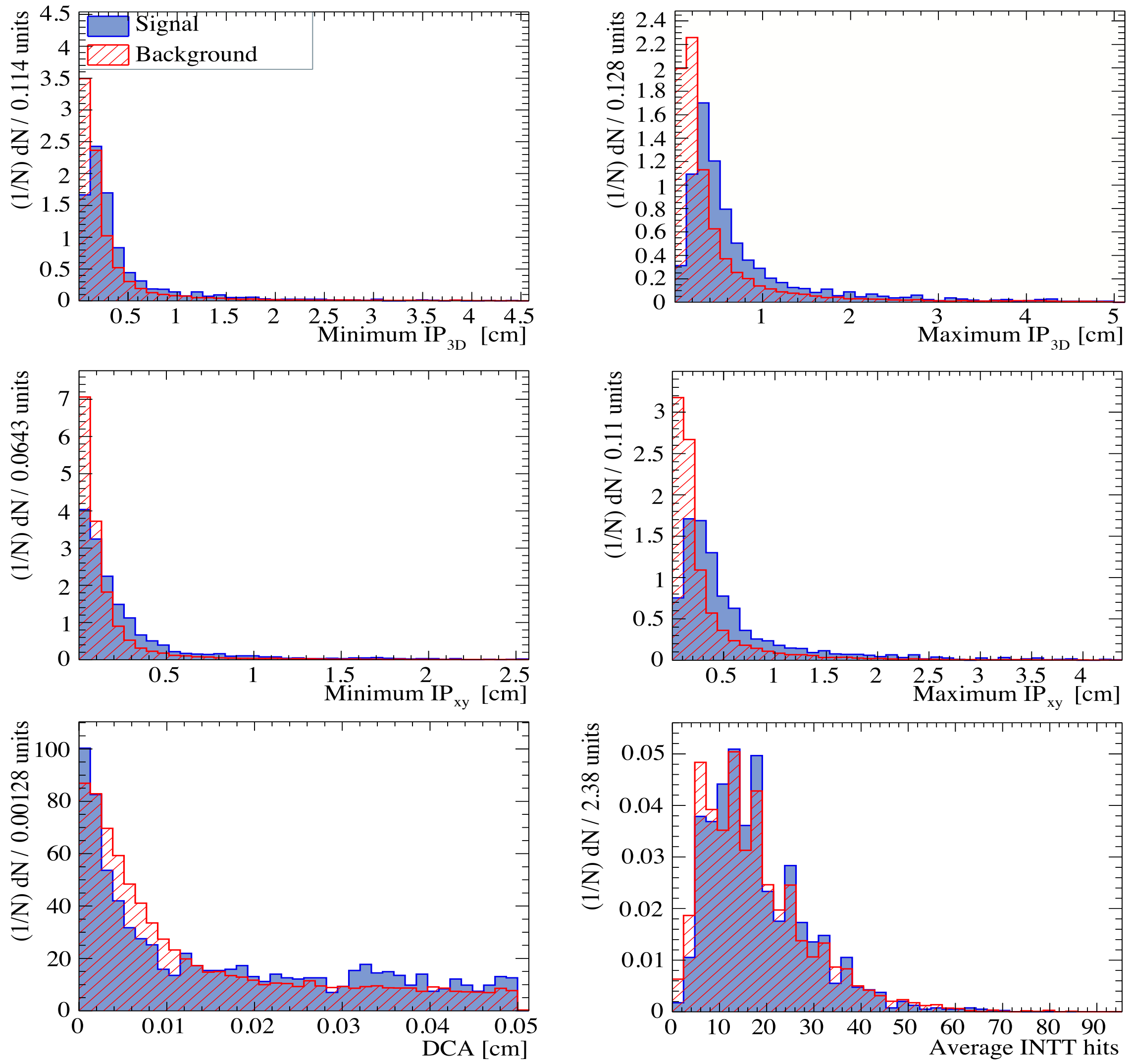
A proposal submitted to the DOE Office of Science
April 30, 2021

- Embed ML algorithms on FPGAs
- Stream trackers to FPGAs and determine if HF event is present through topology
- Monitor and update “beam-spot” in real time
- Send tag downstream to rest of detector
- Outcome announced 2nd December, 2021

Case study: AI HF selections

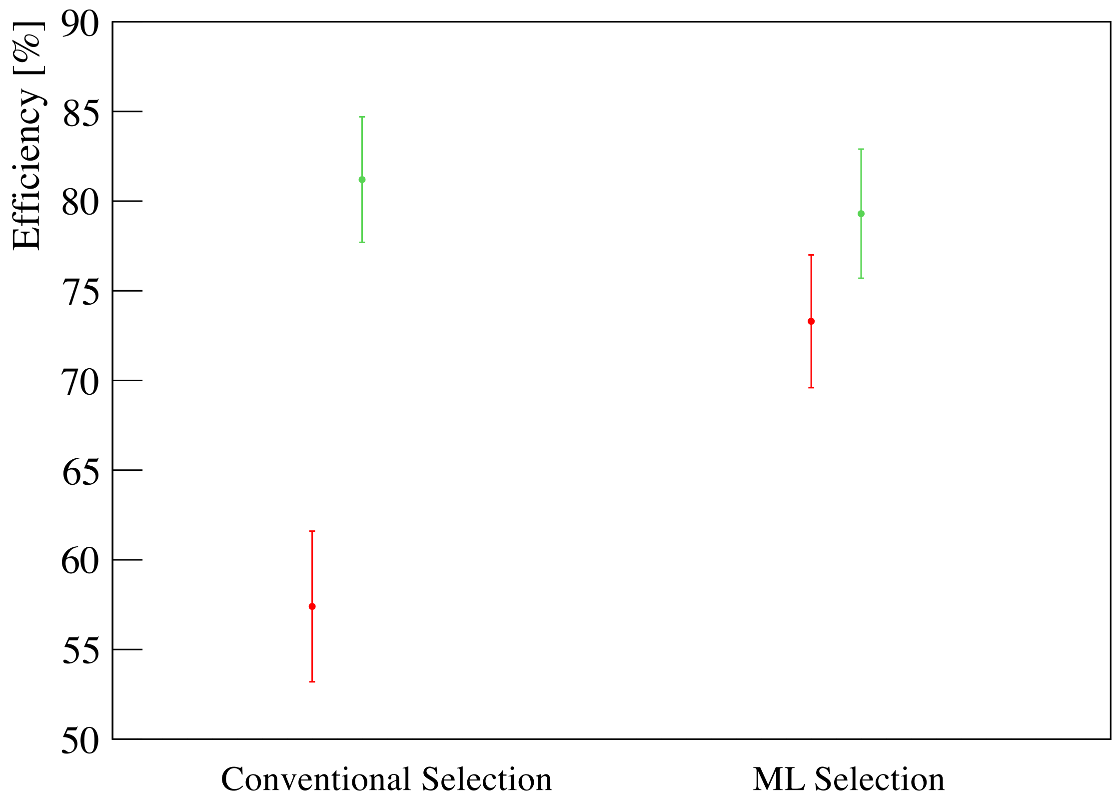
- Question: Is ML better for selecting HF decays over conventional selections?
- Challenge: Must run online, in FPGA. Hence variables must be "simple"





Case study: AI HF selections

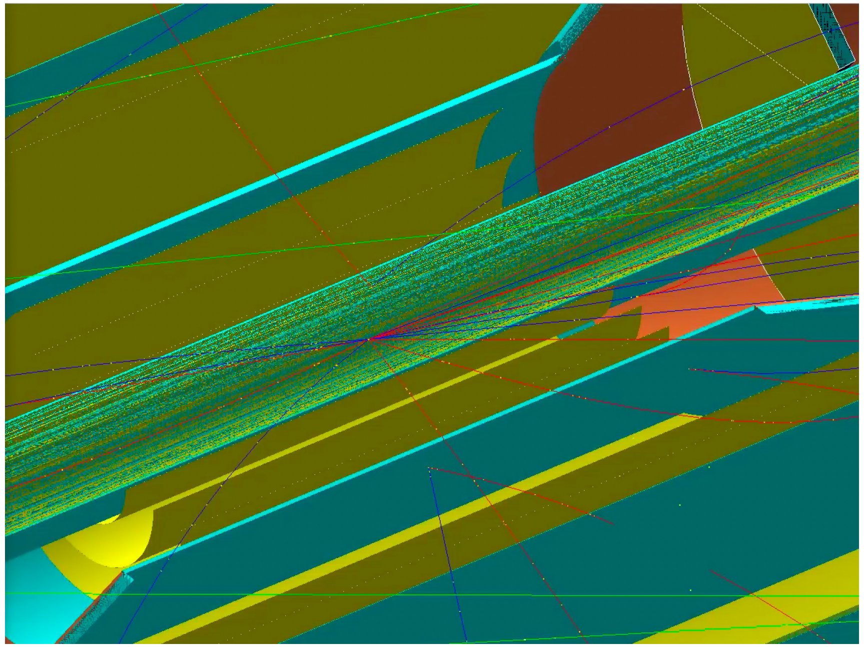
- Several algorithms trained using TMVA
  - Fast turnaround due to proposal time constraints
  - Algorithms used “out-of-the-box”, no optimizations
- Trained using samples with no HF signal and with $D^0 \rightarrow K^-\pi^+$ signal
- Selection tuned for approx. equal signal efficiency

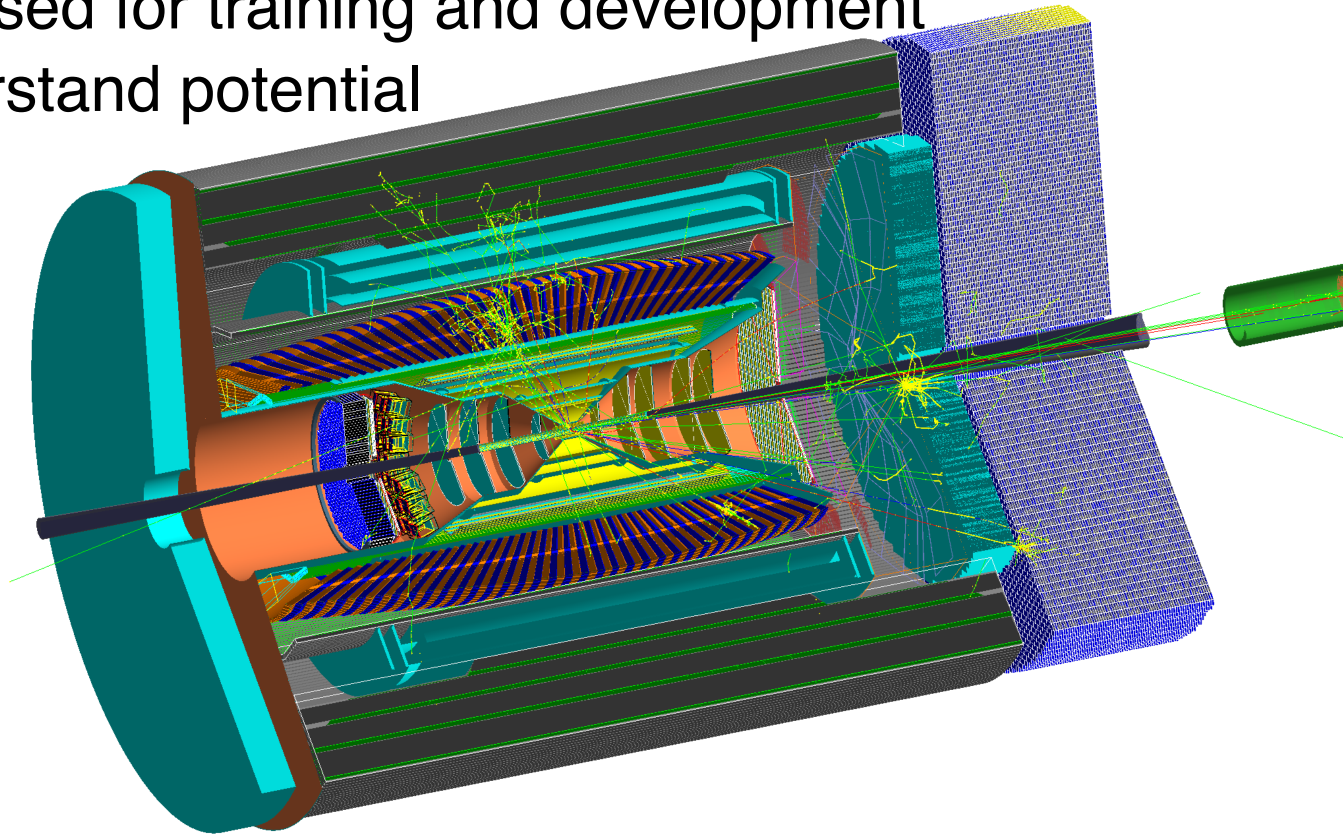


Green – The signal selection efficiency
Red – The background rejection efficiency

Simulating events

- EIC physics simulations progressed rapidly in 2021 and 2022
- No full EIC digitization yet
  - sPHENIX digitization can/will be used for training and development
  - We can use smeared hits to understand potential





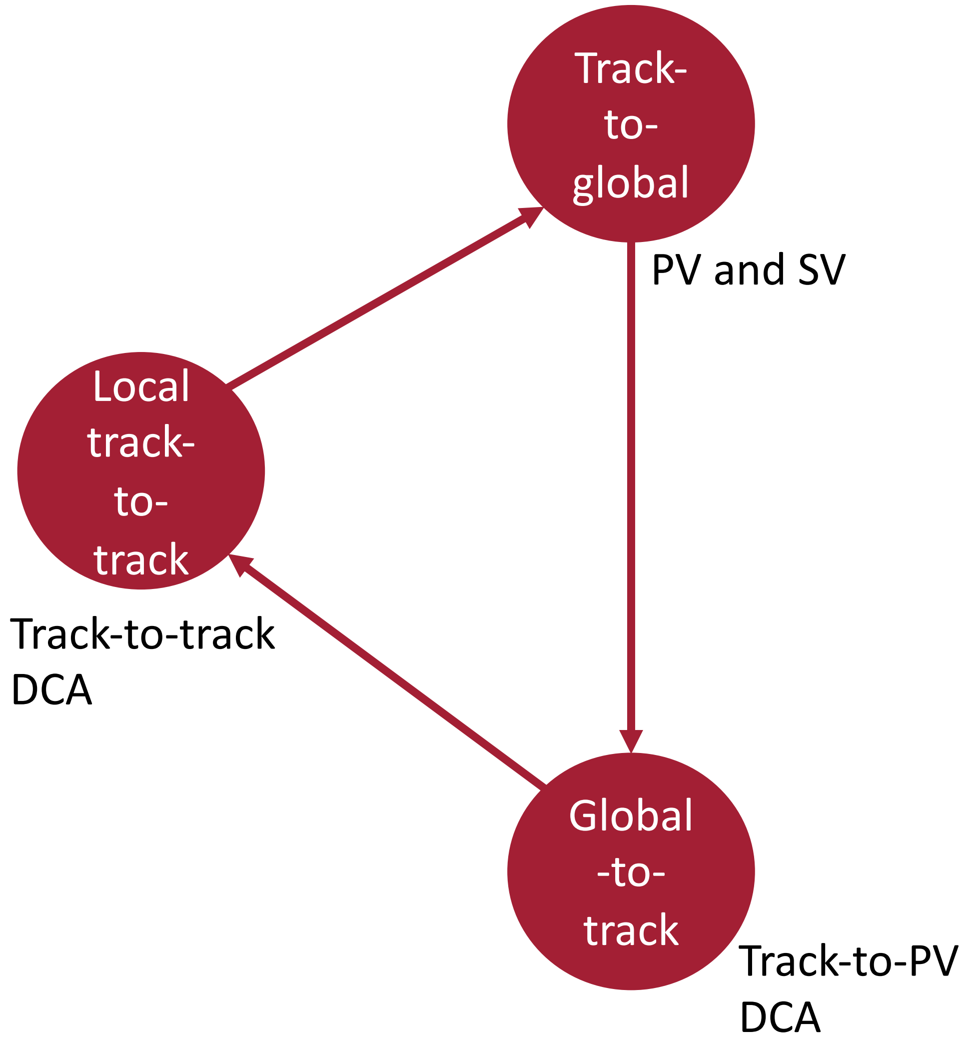
Constructing ML algorithms

- Aim to develop algorithms as Graph Neural Networks (GNN)
- Advantageous over Convolutional Neural Networks (CNN) by adding edge information
- Detector and physics knowledge will improve predictions
- Algorithms deployed at several points:

1. Fast tracking on FPGA
2. Topological separation of HF signals on FPGA
3. Beam-spot and anomaly detection on GPU
    - Part of feedback system to improve 1 & 2 plus inform detector operators

Feedback algorithms

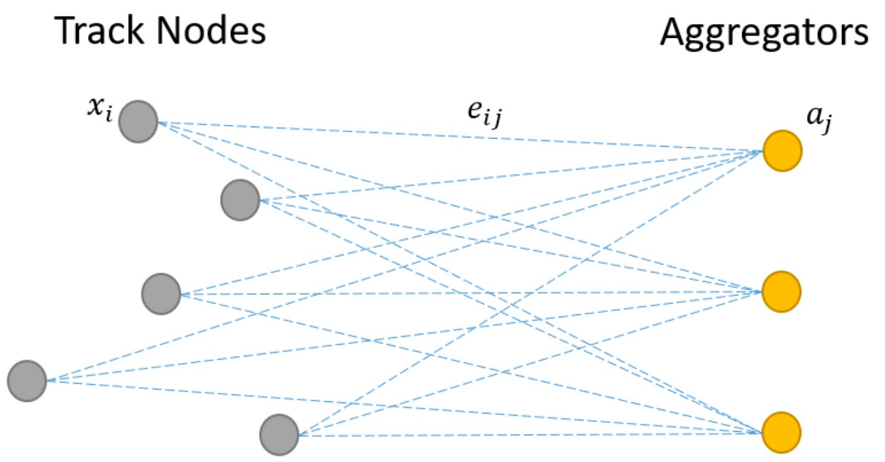
- We have been working on tracking algorithms using simulated signal and background events in the MVTX and INTT
- Used these models to feed into physics selection models to select interesting events
  - Models are bi-directional, local information is passed to global and global information is passed back to local to refine
- Initial trainings and models are developed on GPU
  - NVIDIA Titan RTX, A5000, and A6000
  - Will take the model and convert it to IP block for FPGA deployment
  - Models developed with PyTorch and PyTorch Geometric



GNN models

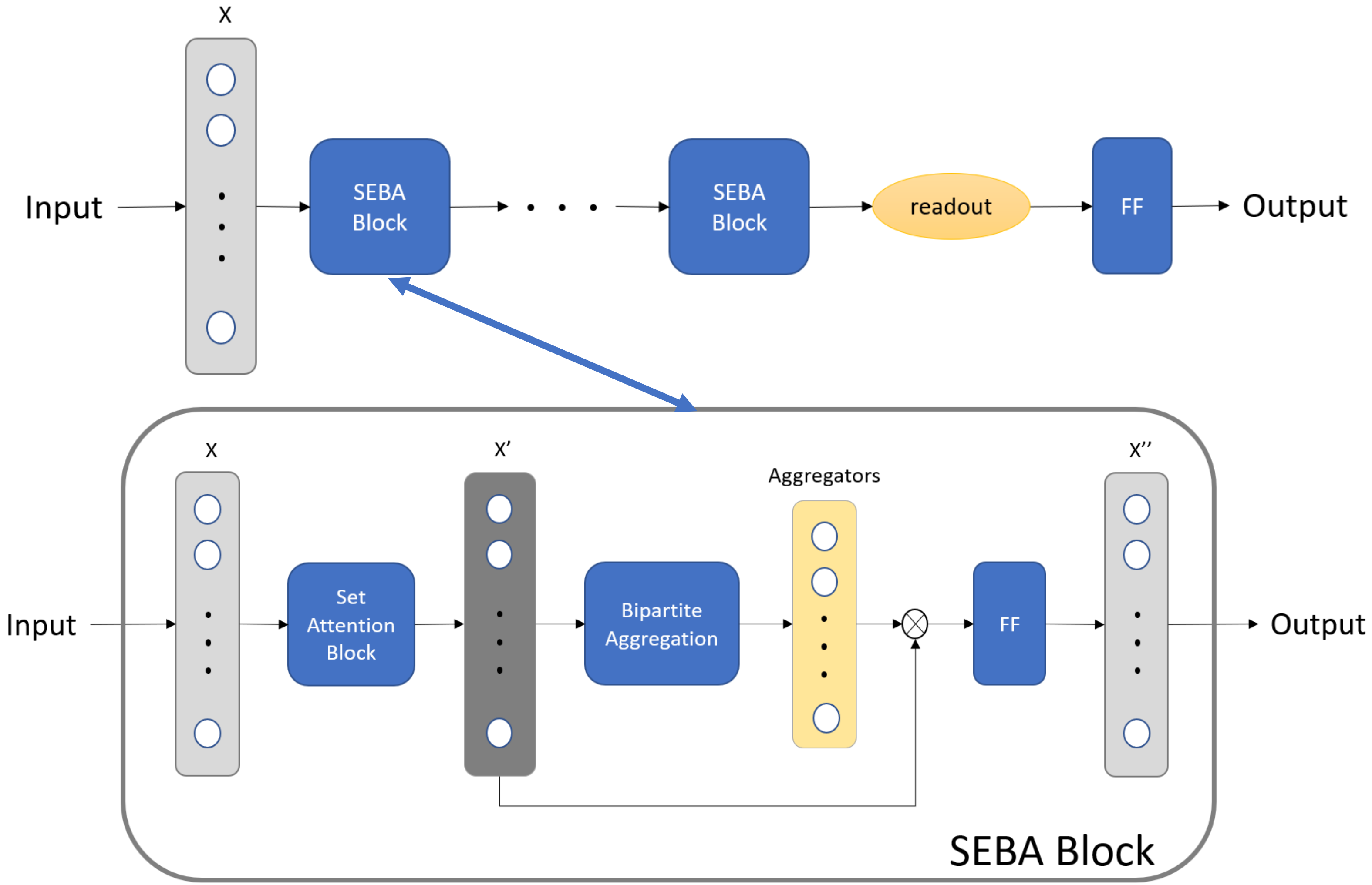
- Track input vectors
  1. 5 hits (MVTX + INTT)
  2. Length of each segment: $L = |\overrightarrow{x_{i+1}} - \overrightarrow{x_i}|$
  3. Angle between segments
  4. Total length of segments
- Aggregators
  1. Primary vertex
  2. Secondary vertex
- Data matrix (X) is:

$X \in \mathcal{R}^{nd}$ where n is the number of tracks and d is the track vector dim.



$e_{ij} = s_{ij} x_i$ is track-aggregator messages
s_{ij} is the weight

GNNs with set transformers



The cycle

1. Track information is initially defined
2. This is relayed to all primary and secondary vertex information
3. Weights are assigned to each link
4. The PV and SV information go through a feedforward NN
5. This updates the track information

pT estimation (sPHENIX)

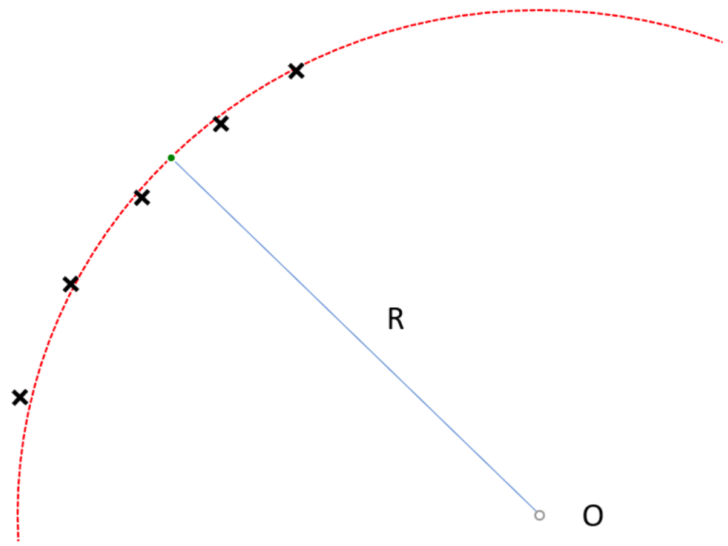
- pT is a good observable to discriminate signal and background
- Inner tracker measurement arm is too small for sPHENIX momentum measurement
  - We add information from TPC
- NJIT team developed algorithms to estimate pT based on least-squares method to produce a best fit circle

A circle is represented by the following formula: $x^2 + y^2 + \beta_1 x + \beta_2 y + \beta_3 = 0$. Given a track of k_T hits $T = \{(x_1, y_1), (x_2, y_2), ..., (x_{k_T}, y_{k_T})\}$, we define a linear system that consists of k_T equations for these hits and attempt to derive the circle's coefficients $\beta = [\beta_1, \beta_2, \beta_3]^T$. To get the best circle approximation, we use the least-squares (LS) optimization to solve the linear regression equation and extract the β coefficients:

$$\beta = (A^T A)^{-1} A^T B.$$

Here $A = \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ \dots & & \\ x_{k_T} & y_{k_T} & 1 \end{bmatrix}$, $B = [-x_1^2 - y_1^2, -x_2^2 - y_2^2, ..., -x_{k_T}^2 - y_{k_T}^2]^T$. With the optimized coefficients for the fitted circle, the circle radius is as follows:

$$R = \frac{1}{2}\sqrt{\beta_1^2 + \beta_2^2 - 4\beta_3}. \tag{3}$$



pT estimation (sPHENIX)

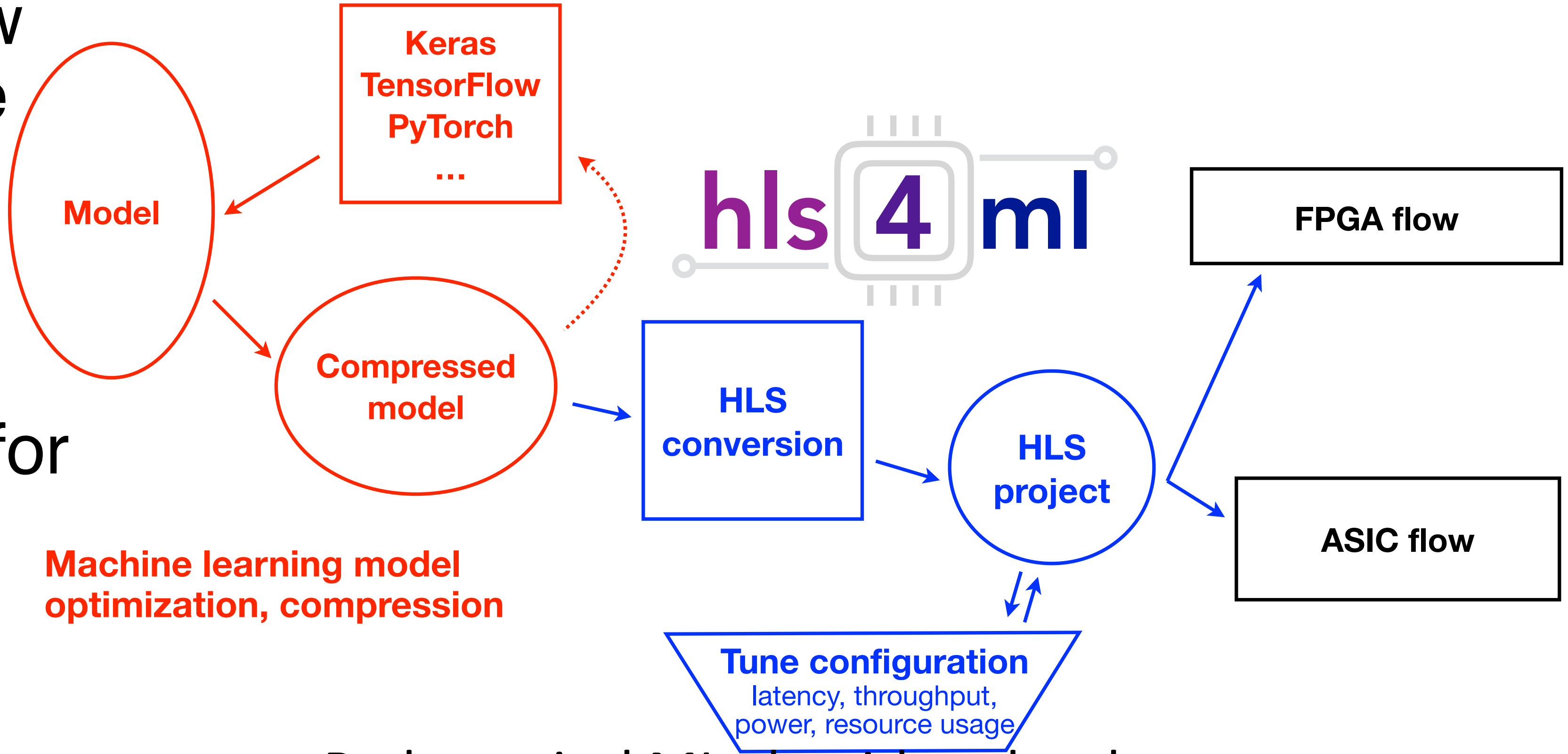
- A feed-forward neural net is used to predict the pT
- First results ~15% improvement in tracking with pT estimation

	with LS-radius			without radius		
Model	#Parameters	Accuracy	AUC	#Parameters	Accuracy	AUC
Set Transformer	300,802	84.17%	90.61%	300,418	69.80%	76.25%
GarNet	284,210	90.14%	96.56%	284,066	75.06%	82.03%
PN+SAGPool	780,934	86.25%	92.91%	780,678	69.22%	77.18%
BGN-ST	355,042	**92.18%**	**97.68%**	354,786	**76.45%**	**83.61%**

	LS		MLP	
Hidden dim	Accuracy	AUC	Accuracy	AUC
32	91.52%	97.33%	91.48%	97.31%
64	92.18%	97.68%	92.23%	97.73%
128	**92.44%**	**97.82%**	**92.49%**	**97.86%**

Translating to firmware

- Algorithms must have low latency and resource use
- *hls4ml* translates NN algorithms into high level synthesis
- Also generates IP cores for easy implementation

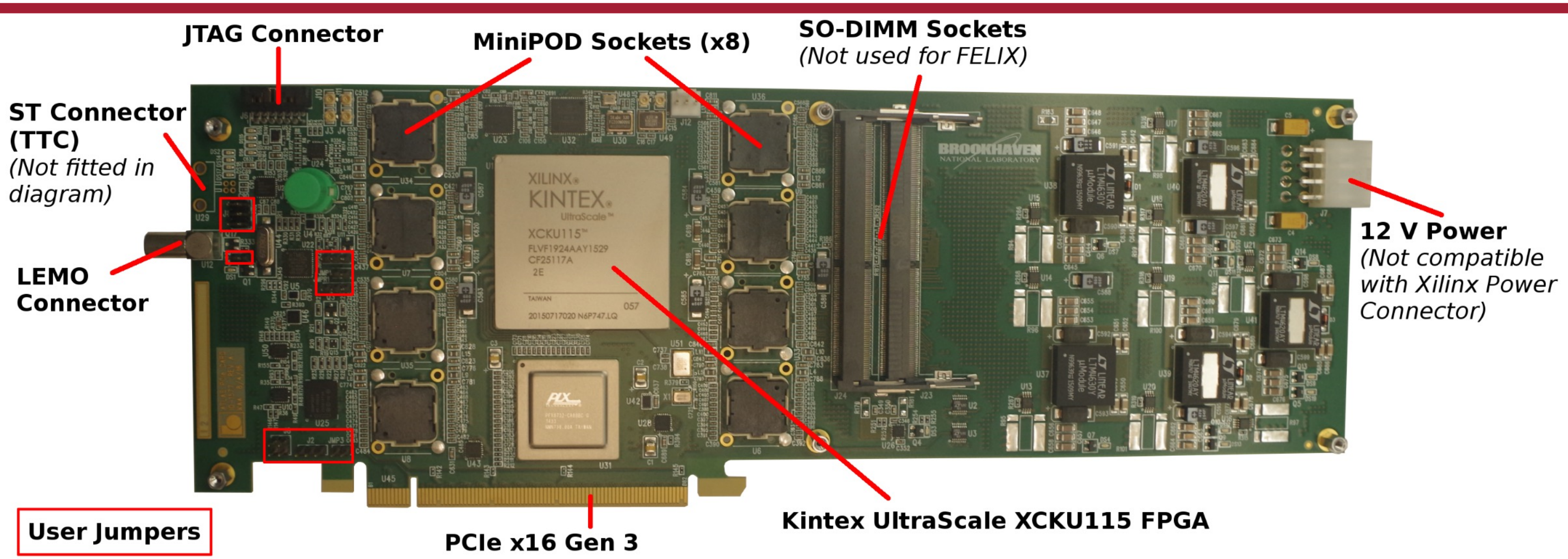


Red – typical ML algorithm development stages

Blue – HLS conversion to IP

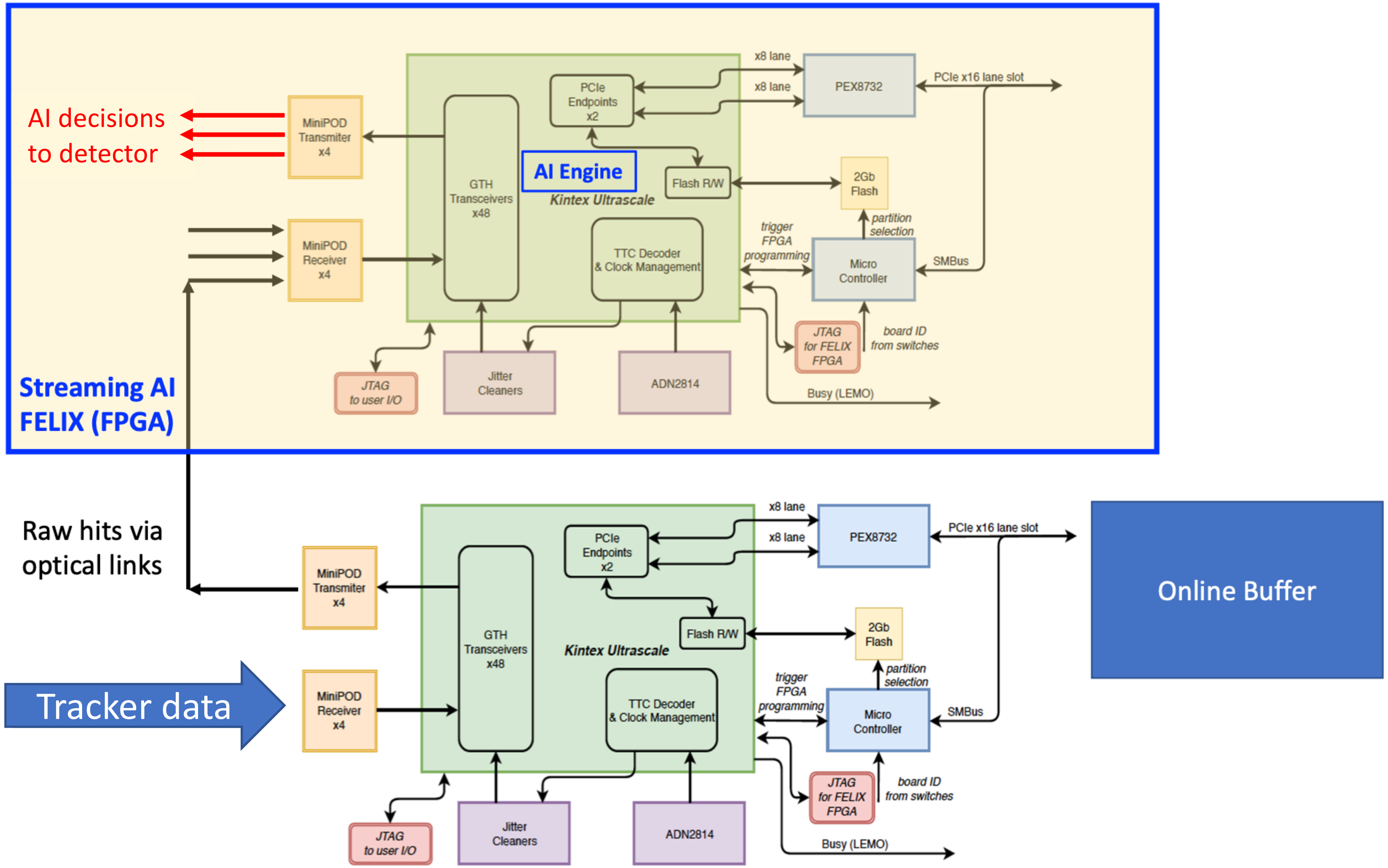
Black – typical implementation onto chips

Realizing in firmware

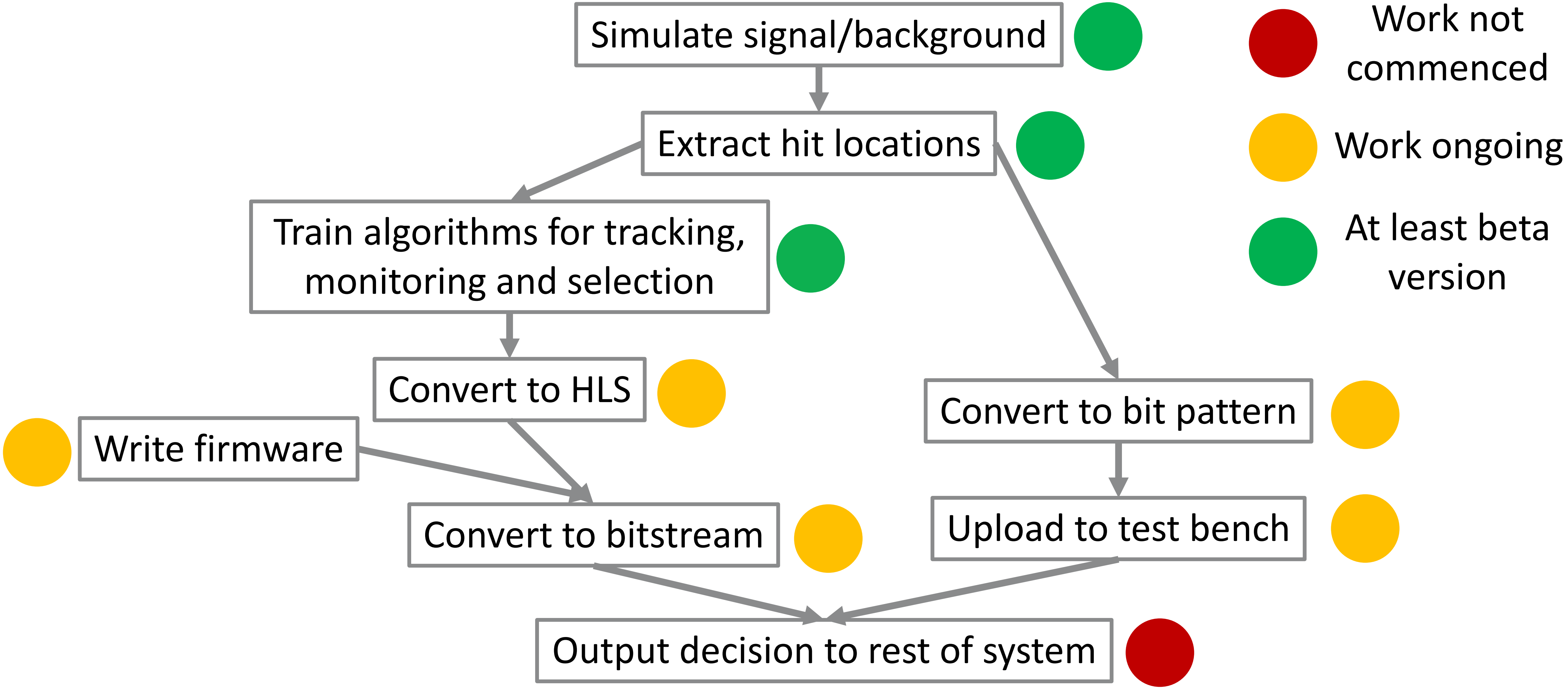


- Decision hardware is currently a BNL-711 FELIX board
  - Current experiments deploy an BNL-712
  - BNL-711 has more on-board memory for buffering
- Team can successfully transfer data from BNL-712 to KC-705 evaluation board
- Current work on reducing resource usage in BNL-711 firmware

Realizing in firmware



Workflow


Simulate signal/background
Extract hit locations
Train algorithms for tracking, monitoring and selection
Convert to HLS
Write firmware
Convert to bitstream
Convert to bit pattern
Upload to test bench
Output decision to rest of system
Work not commenced
Work ongoing
At least beta version

Predicted timeline

2021
- Project started
- Initial simulations constructed
- First data for algorithm training

2022
- SRO development
- Fast tracking algorithms in place
- GPU feedback machine design
- Initial bitstream synthesis

2023
- Refine interface between system and detectors
- Improve algorithms with latest data stream and commissioning info

2024
- Deploy device at sPHENIX

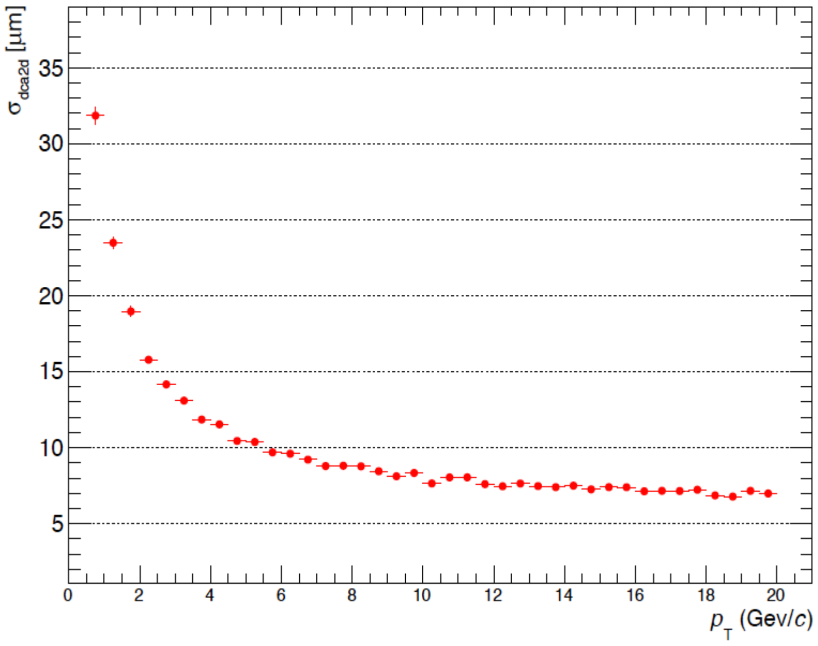
(after 2024)
- Design updated system
- Take advantage of new technology if required

2032
- Deploy device at EIC

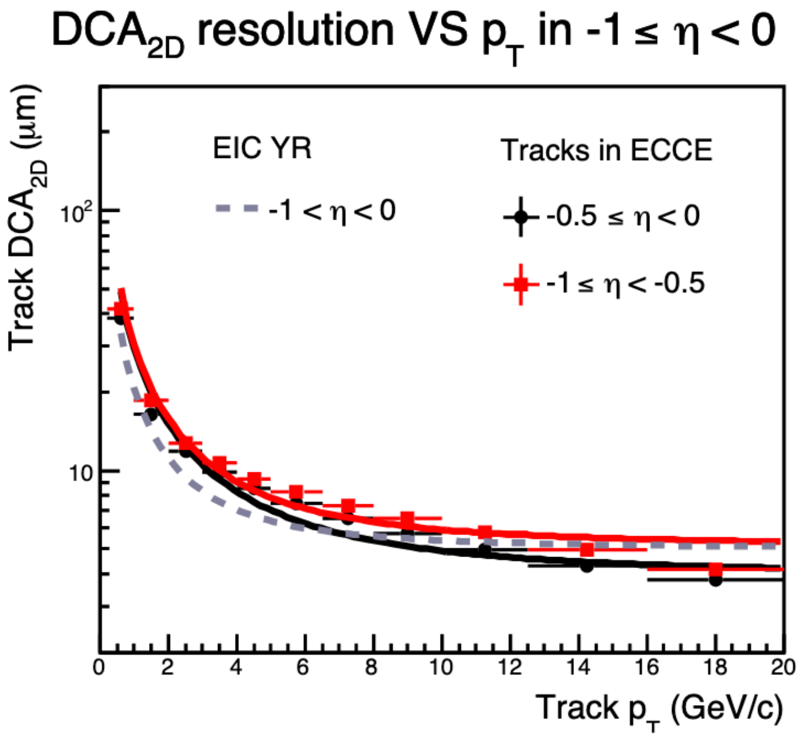
Backup

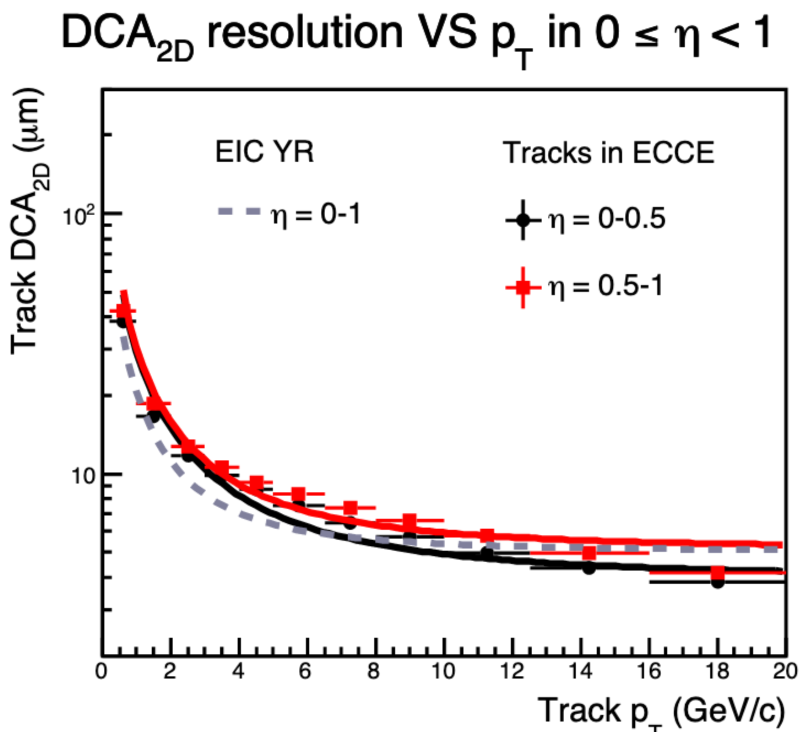
DCA resolution

sPHENIX



ECCE

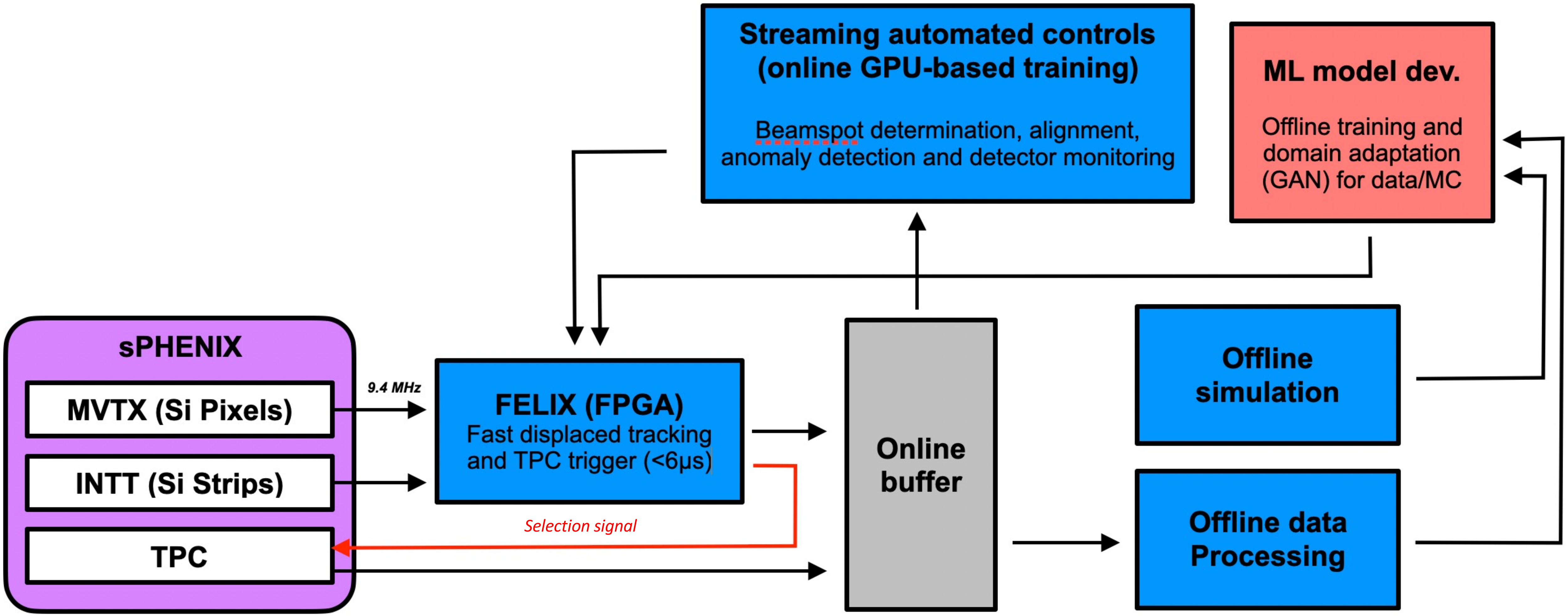




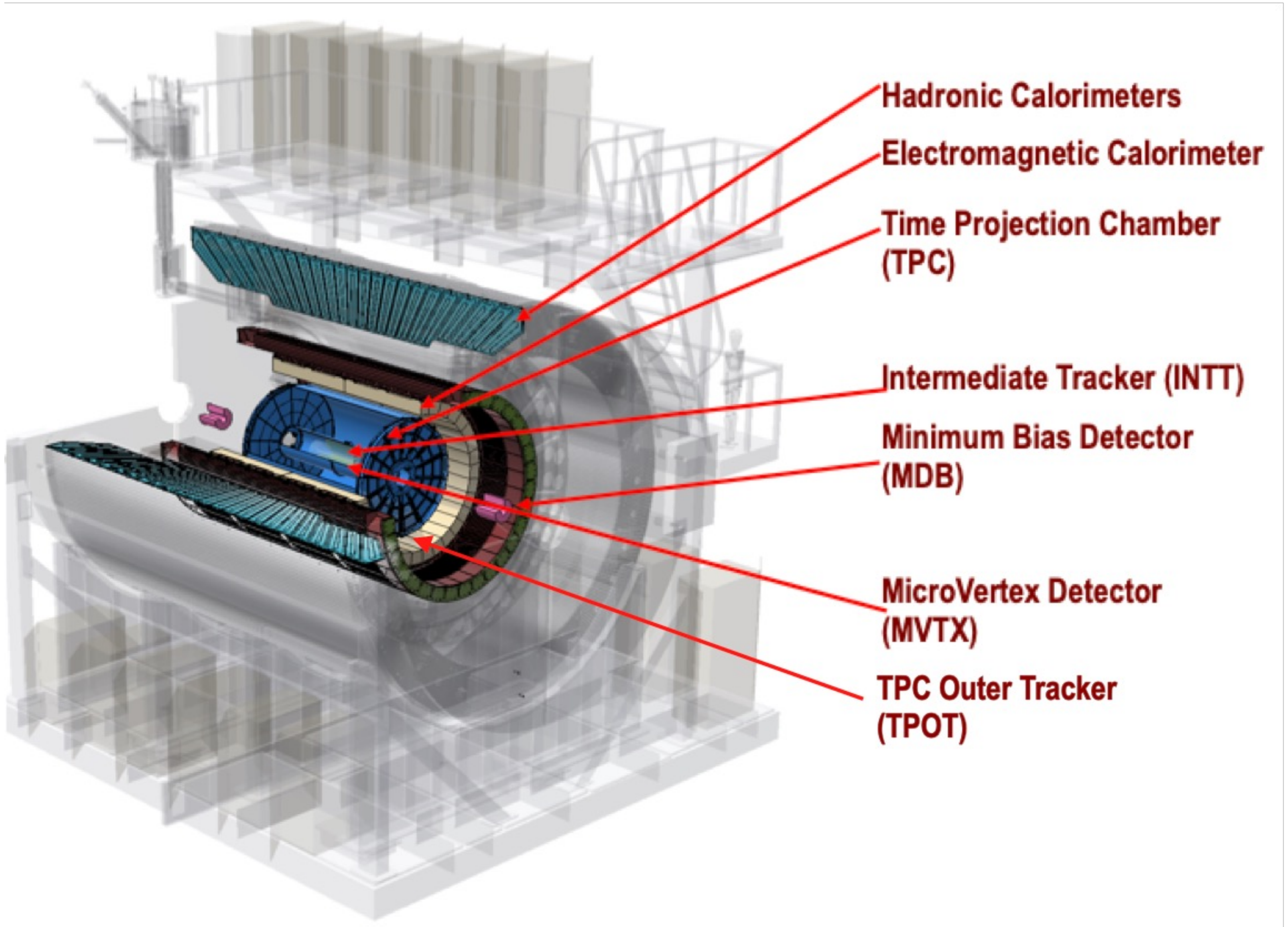
sPHENIX σ_{DCA} = 17 μm at 2 GeV, 7 μm at 10 GeV

ECCE σ_{DCA} = 11 μm at 2 GeV, 5 μm at 10 GeV

Overcoming with AI



sPHENIX

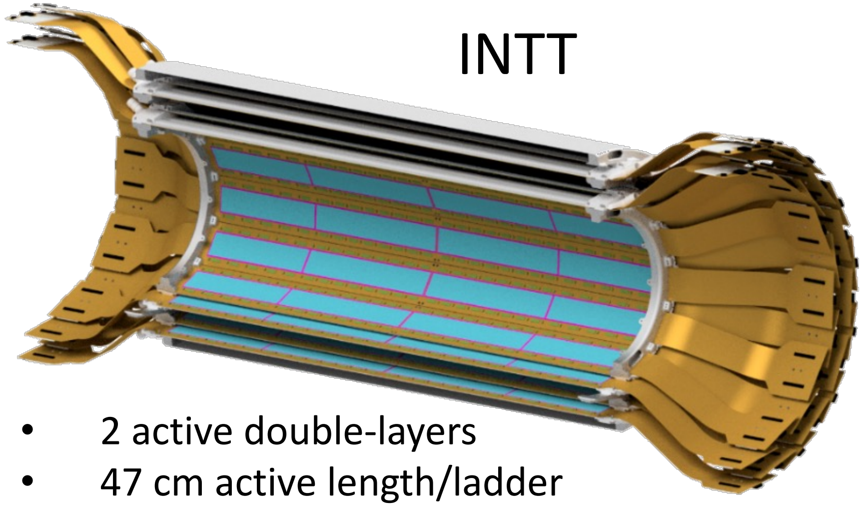


First run year	2023
$\sqrt{s_{NN}}$ [GeV]	200
Trigger Rate [kHz]	15
Magnetic Field [T]	1.4
First active point [cm]	2.5
Outer radius [cm]	270
$\lvert\eta\rvert$	$\leqslant$1.1
$\lvert z_{vtx}\rvert$ [cm]	10
N(AuAu) collisions*	1.43x10^{11}

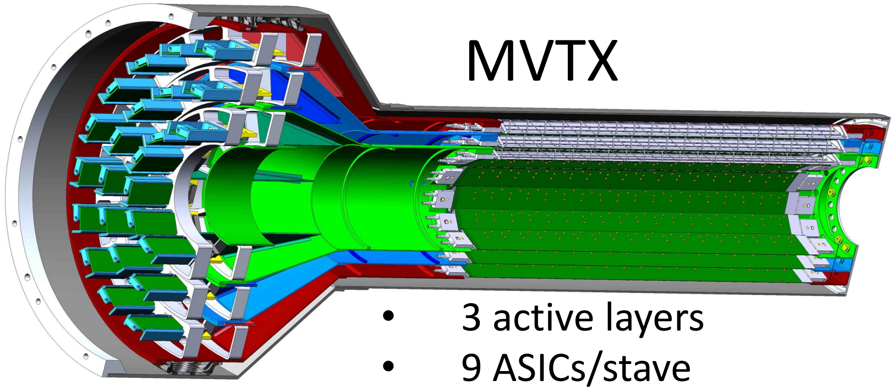
* In 3 years of running

Tracking at sPHENIX

- Tracking consists of 3 sub-detectors:
  - Pixel Vertex Detector (MVTX)
  - Intermediate Silicon Tracker (INTT)
  - Time Projection Chamber (TPC)
- MVTX and INTT are both capable of streaming readout
- Combined tracking to r = 10.3 cm



- 2 active double-layers
- 47 cm active length/ladder
- Silicon strip detector



- 3 active layers
- 9 ASICs/stave
- 27 cm active length/stave
- Pixel detector

sPHENIX HF constraints

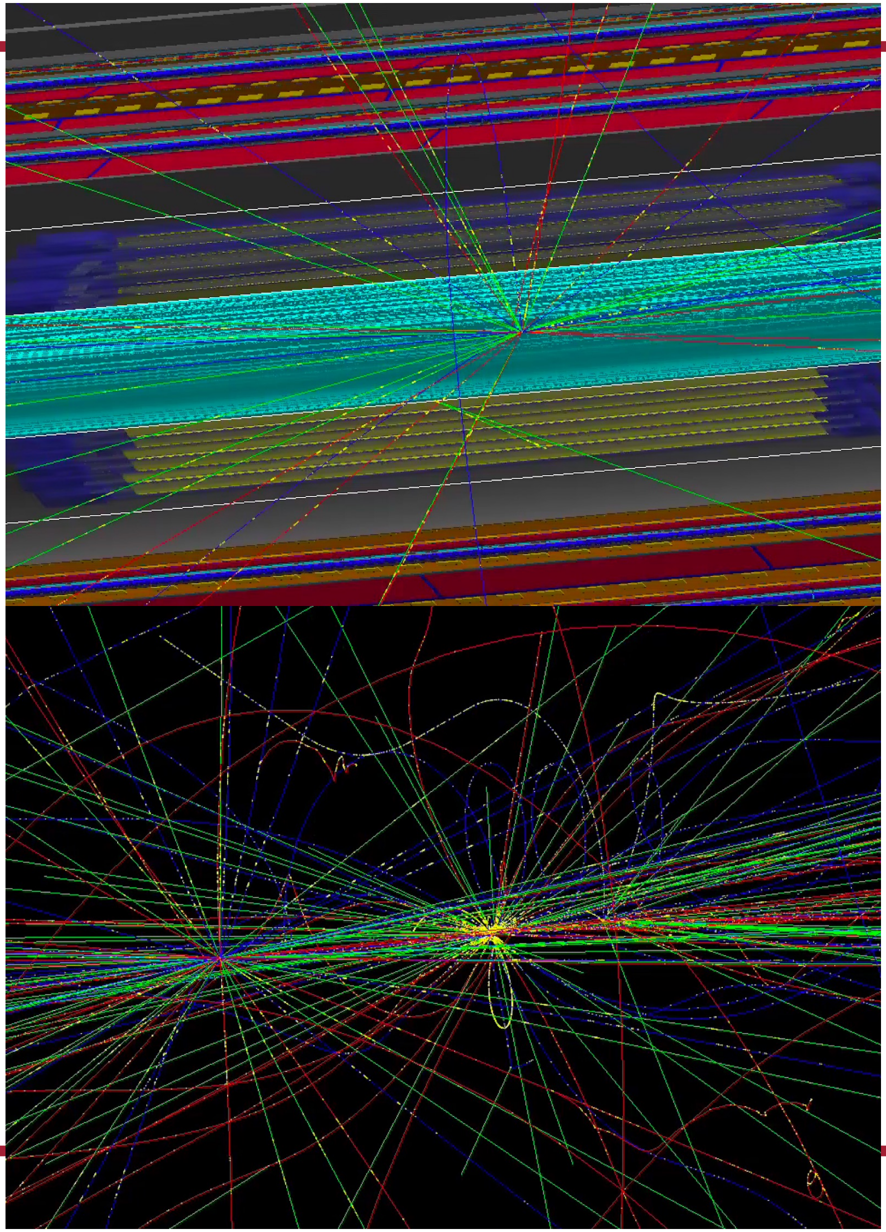
- sPHENIX has great tracking and calorimetry
- However, limited by calorimetry backend readout rate (15kHz) in triggered mode
- RHIC pp rate is ~10 MHz
- Plan: Use tracker SRO to recover some heavy flavor physics potential

Year	Species	$\sqrt{s_{NN}}$ [GeV]	Cryo Weeks	Physics Weeks	Rec. Lum. $\|z\| < 10$ cm	Samp. Lum. $\|z\| < 10$ cm
2023	Au+Au	200	24 (28)	9 (13)	3.7 (5.7) nb^{-1}	4.5 (6.9) nb^{-1}
2024	$p^{\uparrow}p^{\uparrow}$	200	24 (28)	12 (16)	0.3 (0.4) pb^{-1} [5 kHz] 4.5 (6.2) pb^{-1} [10%-*str*]	45 (62) pb^{-1}
2024	$p^{\uparrow}$+Au	200	–	5	0.003 pb^{-1} [5 kHz] 0.01 pb^{-1} [10%-*str*]	0.11 pb^{-1}
2025	Au+Au	200	24 (28)	20.5 (24.5)	13 (15) nb^{-1}	21 (25) nb^{-1}

sPHENIX beam-use proposal. 5 kHz refers to final rate with triggered readout, 10%-str refers to 10% streaming readout

Simulating events (sPHENIX)

- Can already simulate any number of signal and background events with full digitization
- Package developed to extract raw hit information
- Work progressing to use this for algorithm training and bit pattern conversion



Top - $D^0 \rightarrow K^-\pi^+$ simulation at sPHENIX. Beam pipe is in turquois, MVTX is in olive and INTT is in red
Bottom – typical simulated *pp* event at sPHENIX.
Three collisions can clearly be observed