Machine Learning Advances in High Energy Physics

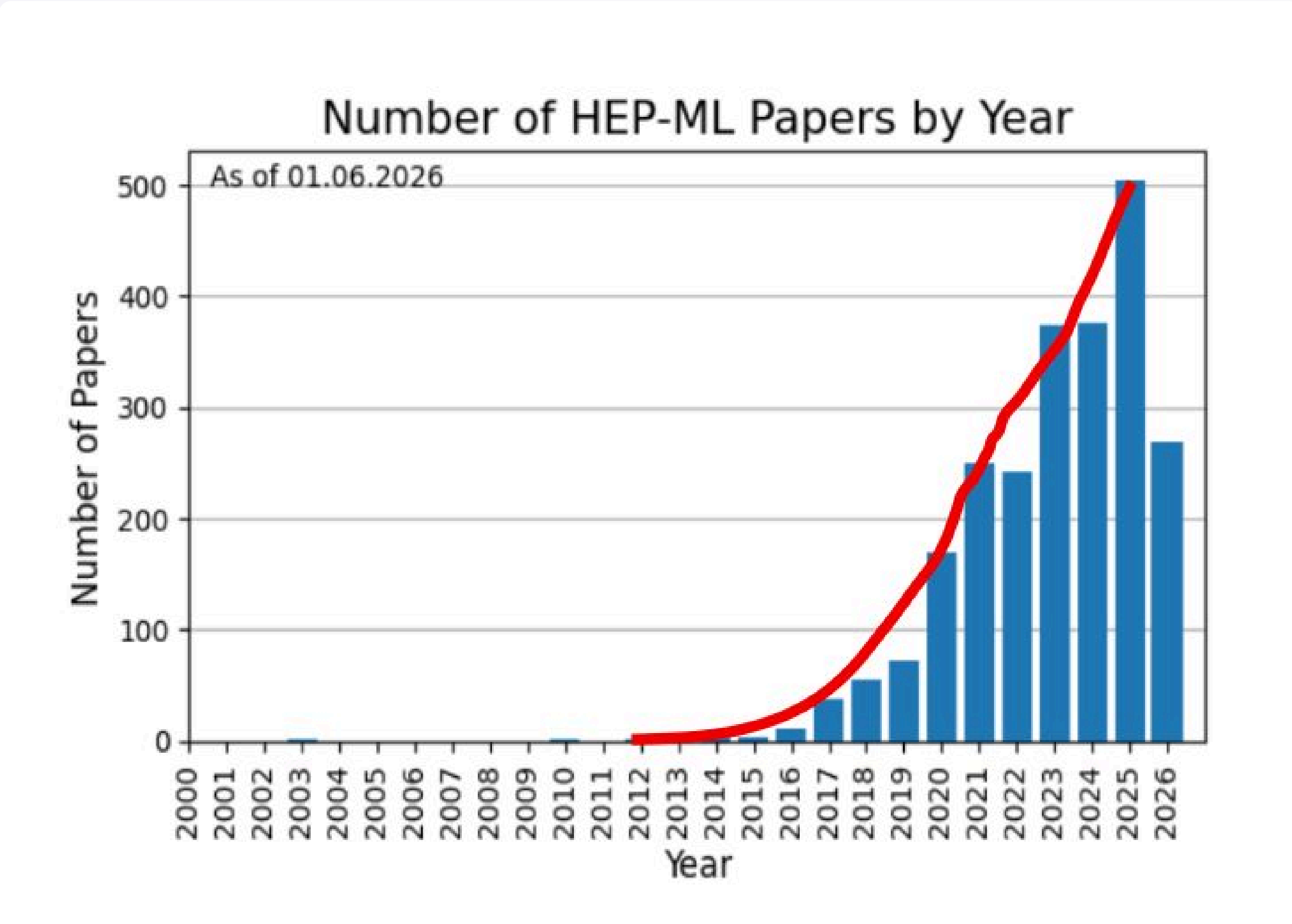
From real-time triggers to autonomous analysis — how deep learning is transforming every stage of the HEP pipeline.

Dmitry Matvienko

HEP ML living review

Real deployments and published results from 2000-2026

Modern machine learning techniques, including deep learning, are rapidly being applied, adapted, and developed for high energy physics

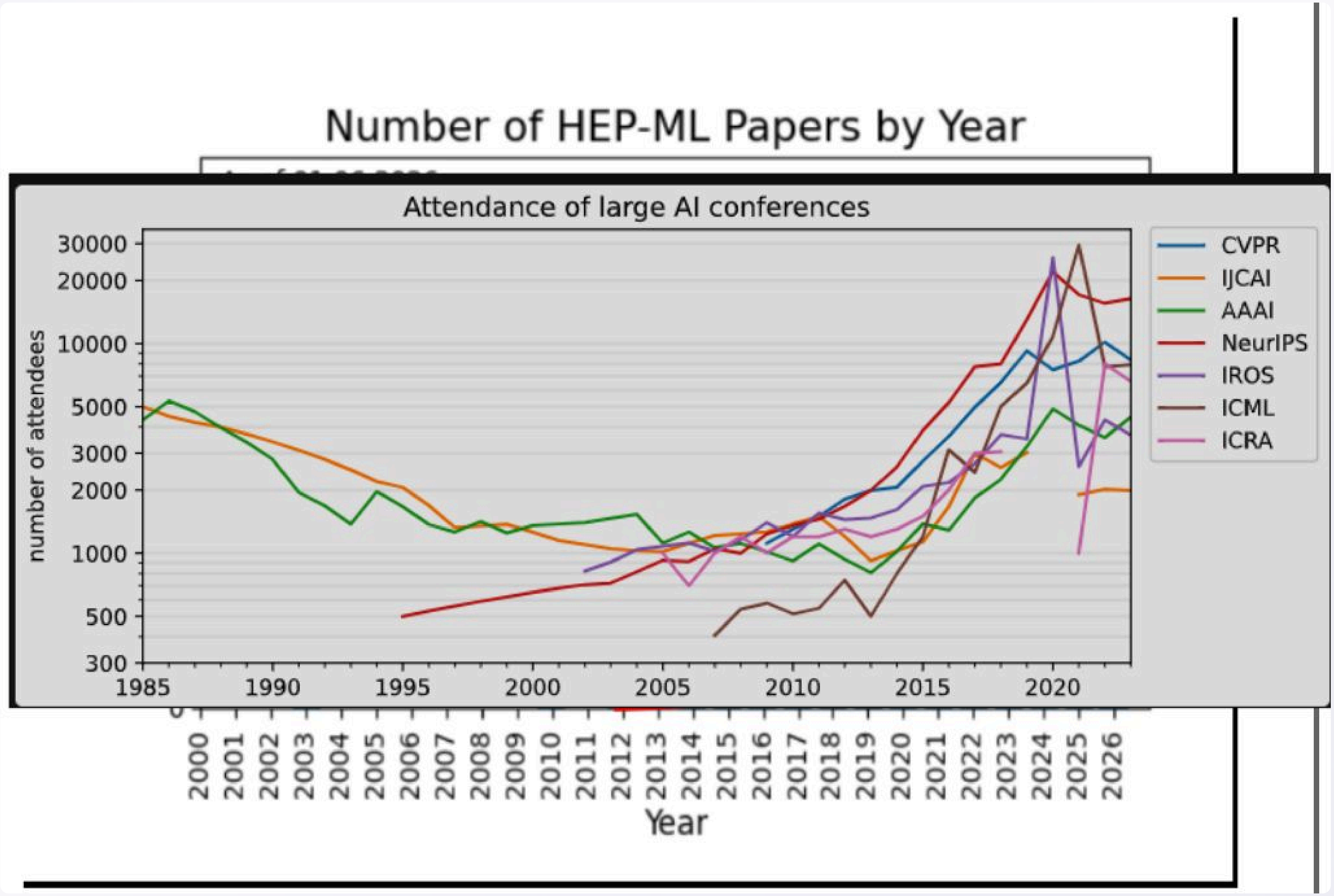


- *Classification / regression*
- *Generative models*
- *Anomaly detection*
- *Foundation models / LLM*
- *Physics-informed NN*
- *Kolmogorov-Arnold Networks*
- *Formal Theory and ML*
- *Uncertainty Quantification*
- *Simulation-based Inference*
- *Reviews*

HEP ML living review

Real deployments and published results from 2000-2026

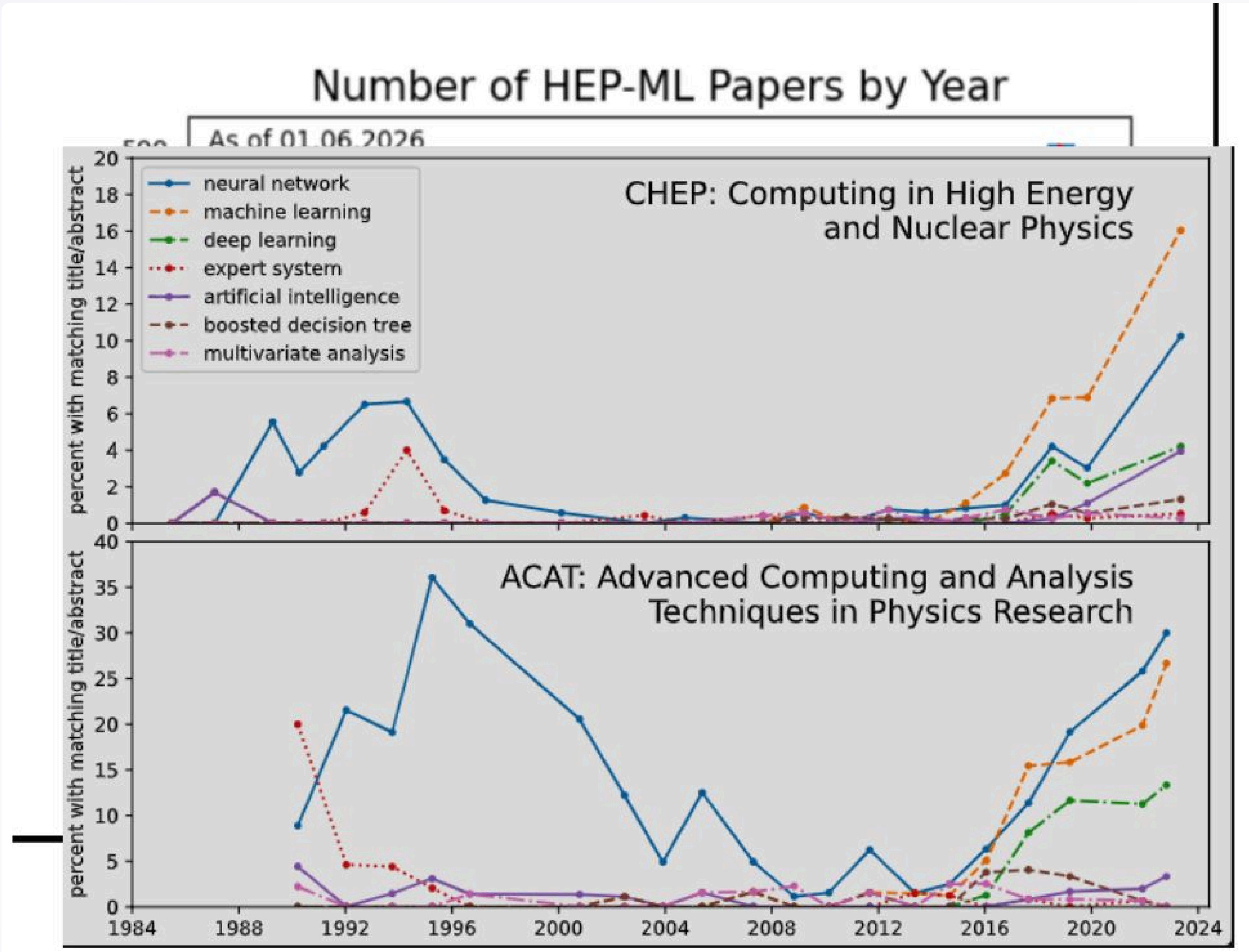
Modern machine learning techniques, including deep learning, are rapidly being applied, adapted, and developed for high energy physics



HEP ML living review

Real deployments and published results from 2000-2026

Modern machine learning techniques, including deep learning, are rapidly being applied, adapted, and developed for high energy physics



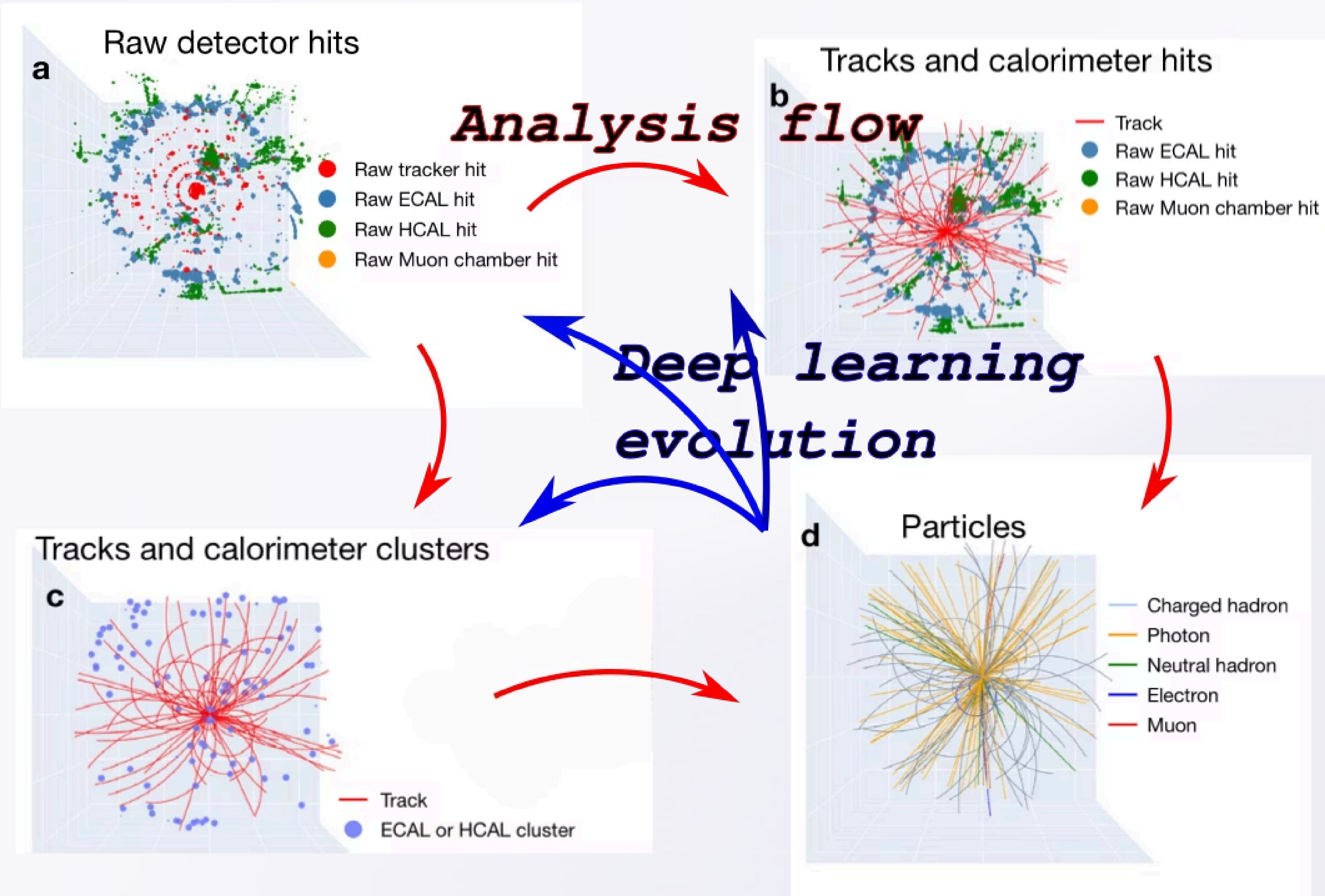
ICHEP 2026

Artificial Intelligence, Machine Learning and Quantum Computing in HEP

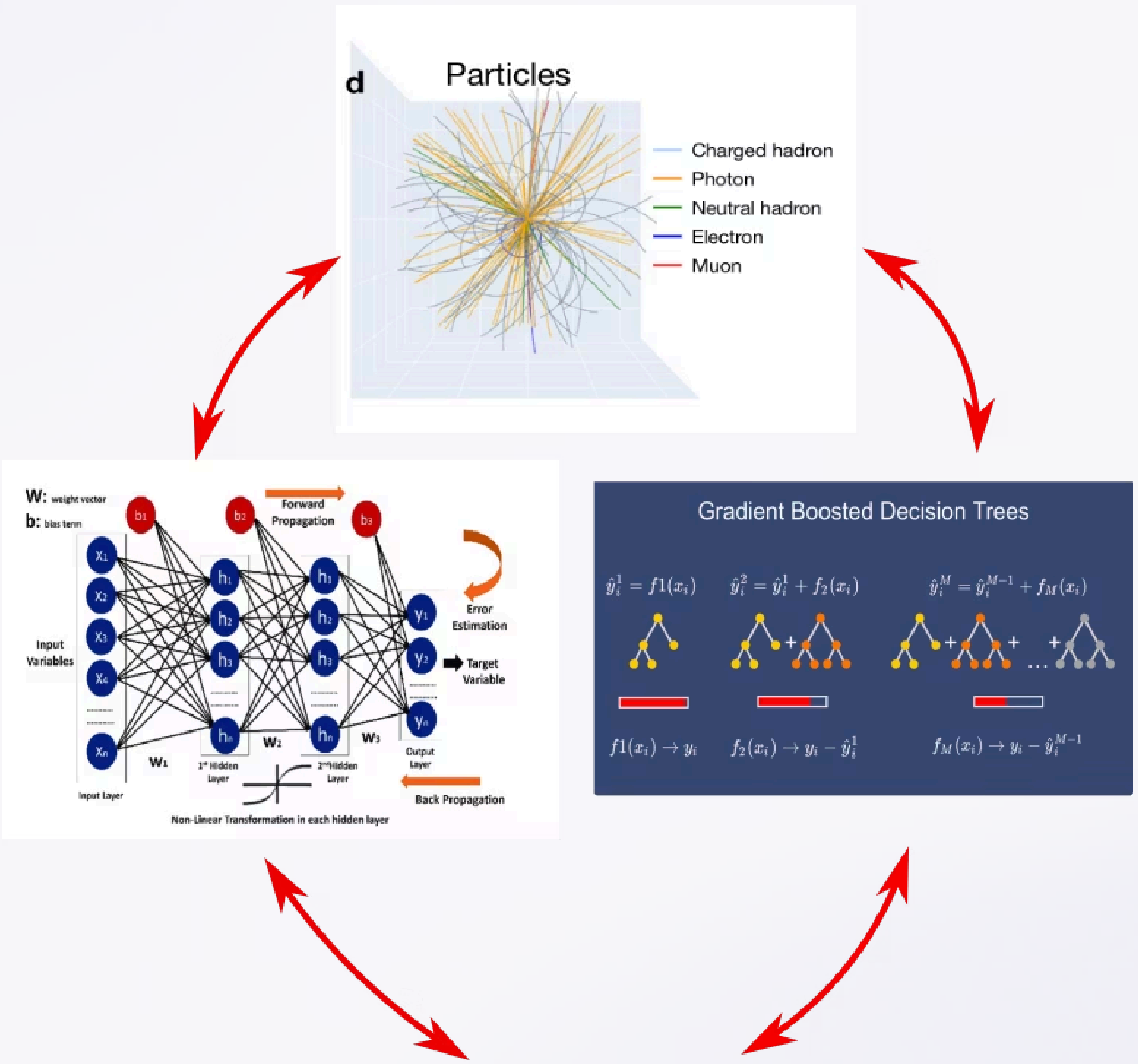
45 parallel and 1 plenary talks

The section is dedicated to addressing computer, networking and software challenges related to the highly demanding needs of the HEP experiments.

Deep learning progress from particles to raw hits



Deep learning progress from particles to raw hits




```
– Signal / Bkg
– Particle ID
– Jet tagging
```

Classical ML (2010's)

Particles as data objects with vector of features: momentum, angles, charge, ...

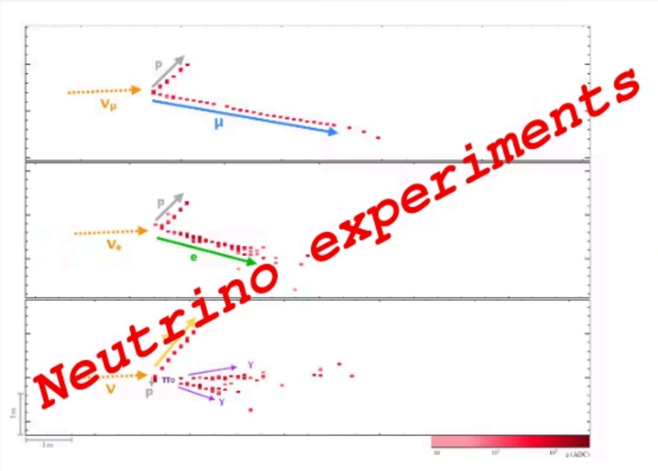
- Classifier training to separate signal vs background

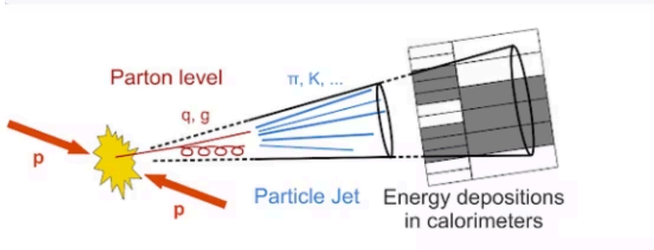
- Particle ID network pion / kaon / proton / ... separation based on energy loss in various detector layers

- Jet tagging, ex. b-tagging jet from b quark. NN trains on aggregating properties of jet (mass, number of tracks in jet, particle vertex displacement)

Cons: real data are not tabular

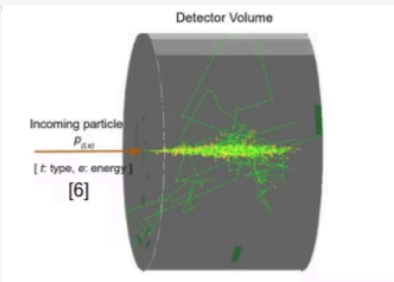
Deep learning progress from particles to raw hits

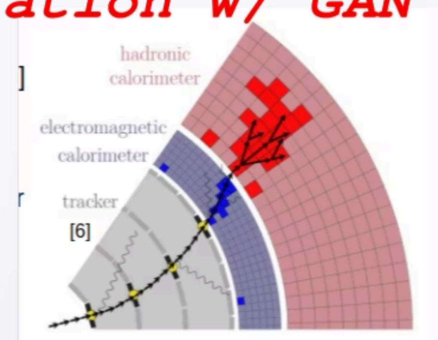




Jet substructure

Fast simulation w/ GAN





Convolutional NN (2015-)

Tracks / Clusters as image objects :

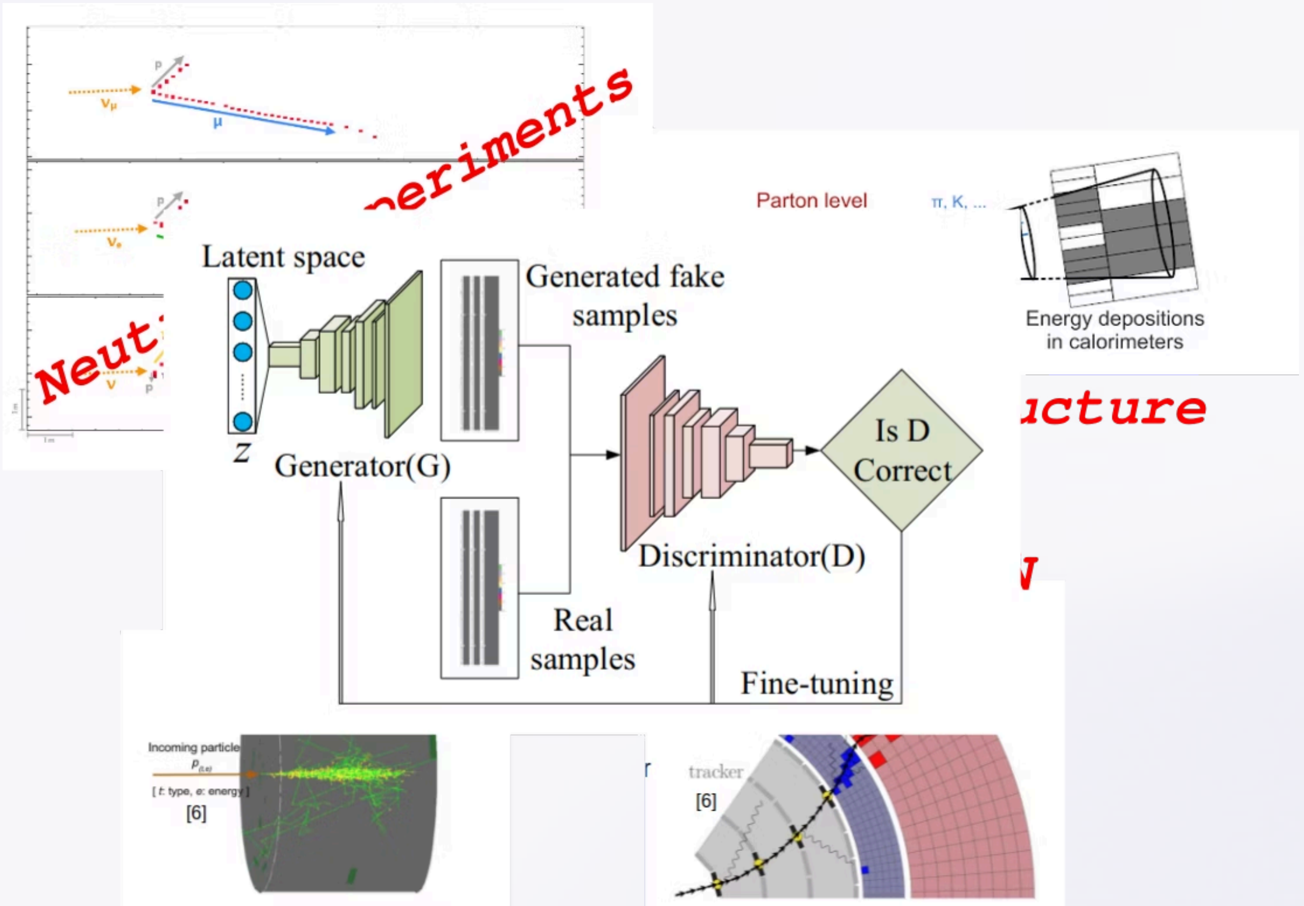
- Neutrino experiments (Nova, MicroBoone, IceCube, DUNE) use CNN to identify neutrino type

- Jet images as calorimeter pixels to determine jet substructure

- Fast simulation w/ generative models where Convolutional layers are broadly used in discriminator

Cons: sparsity

Deep learning progress from particles to raw hits



Convolutional NN (2015-)

Tracks / Clusters as image objects :

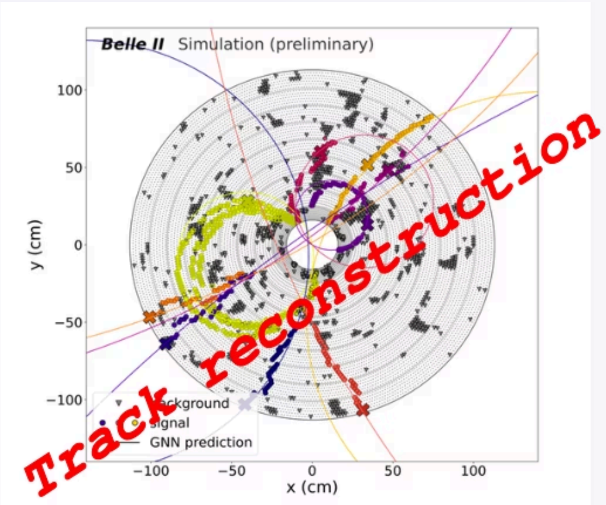
- Neutrino experiments (Nova, MicroBoone, IceCube, DUNE) use CNN to identify neutrino type

- Jet images as calorimeter pixels to determine jet substructure

- Fast simulation w/ generative models where Convolutional layers are broadly used in discriminator

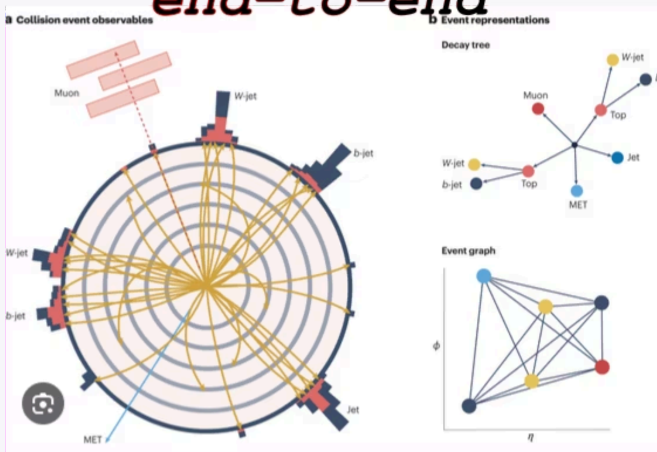
Cons: sparsity

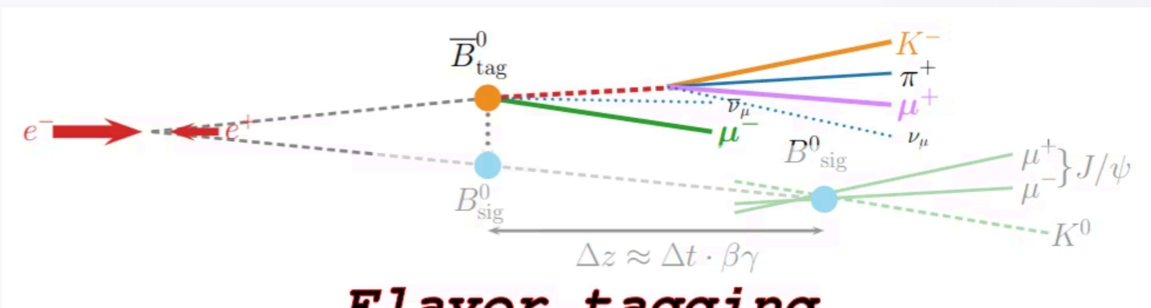
Deep learning progress from particles to raw hits



Track reconstruction

Particle flow
end-to-end





Flavor tagging

Graph NN (2020-)

Raw hits as graph nodes connected to edges

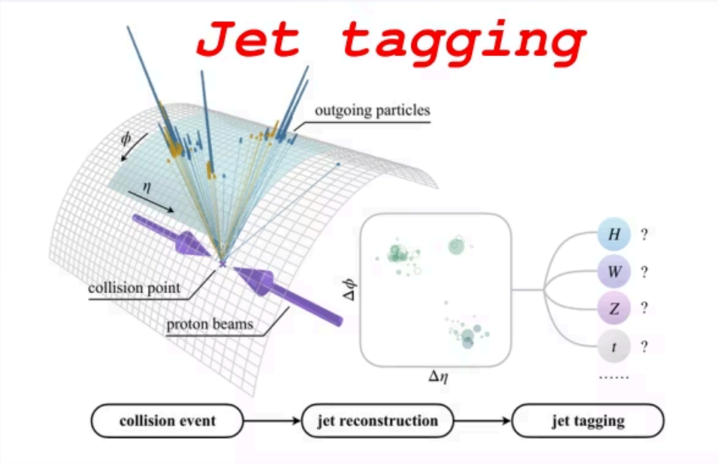
- Track reconstruction in drift chamber

- Flavor tagging where particle in ROE is graph node

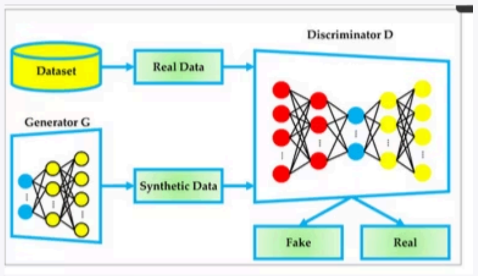
- Particle flow conception to combine all hits into unified list of particles

Cons: over-smoothing, pair connections

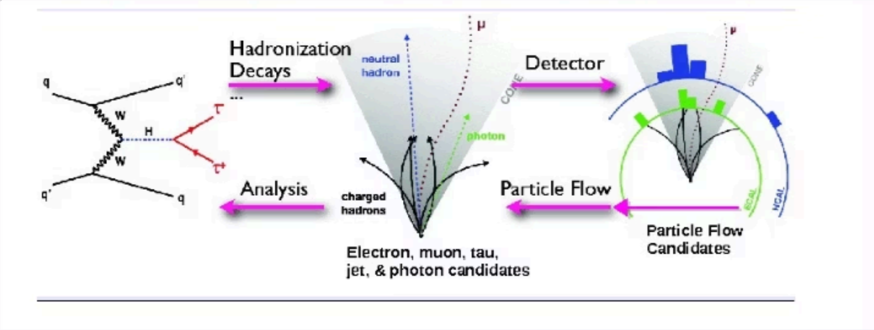
Deep learning progress from particles to raw hits



Fast simulation



Particle flow end-to-end



Transformers (2022-)

Raw hits as objects where self-attention is applied

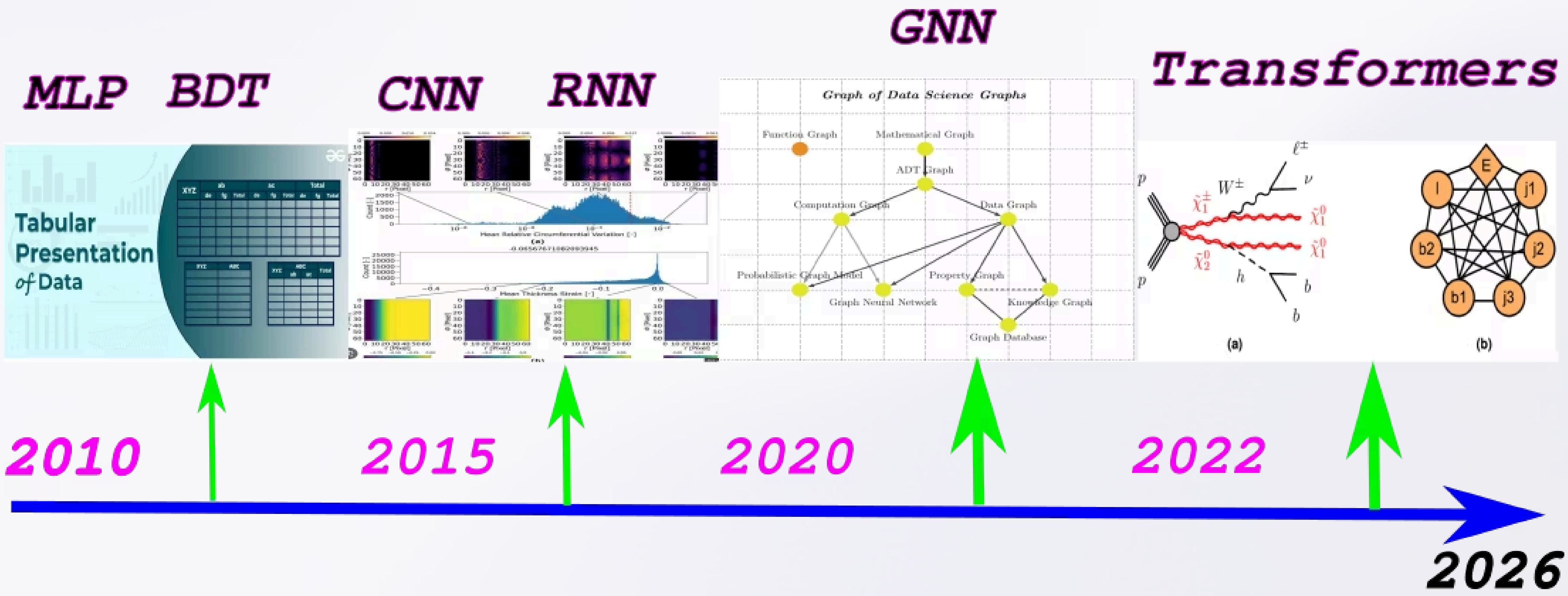
- Jet tagging where attention mechanism is integrated into particle list

- Fast simulation w/ transformer to improve global physical correlations and strength discriminator

- Particle flow conception where all hits form global sequence

Cons: $O(N^2)$

Deep learning progress from particles to raw hits



NN architecture is not just a playground

Not only NN architecture but other tunings are needed to improve performance / interpretability / ...

- Frequentist / Probabilistic → various views on the problem

- Problem motivated regularizations of the loss function, change the concept from likelihood to heuristic.

- Gradient descent adaptations / BackProp / ...

- NN compactification to improve interpretability

Towards a probabistic approach

Towards a probabilistic approach

Working in a **probabilistic framework** has a number of advantages

- Probabilities are **well defined mathematical objects** that we know how to manipulate
- **Bayesian inference**: learn a posterior probability, given a model and priors,

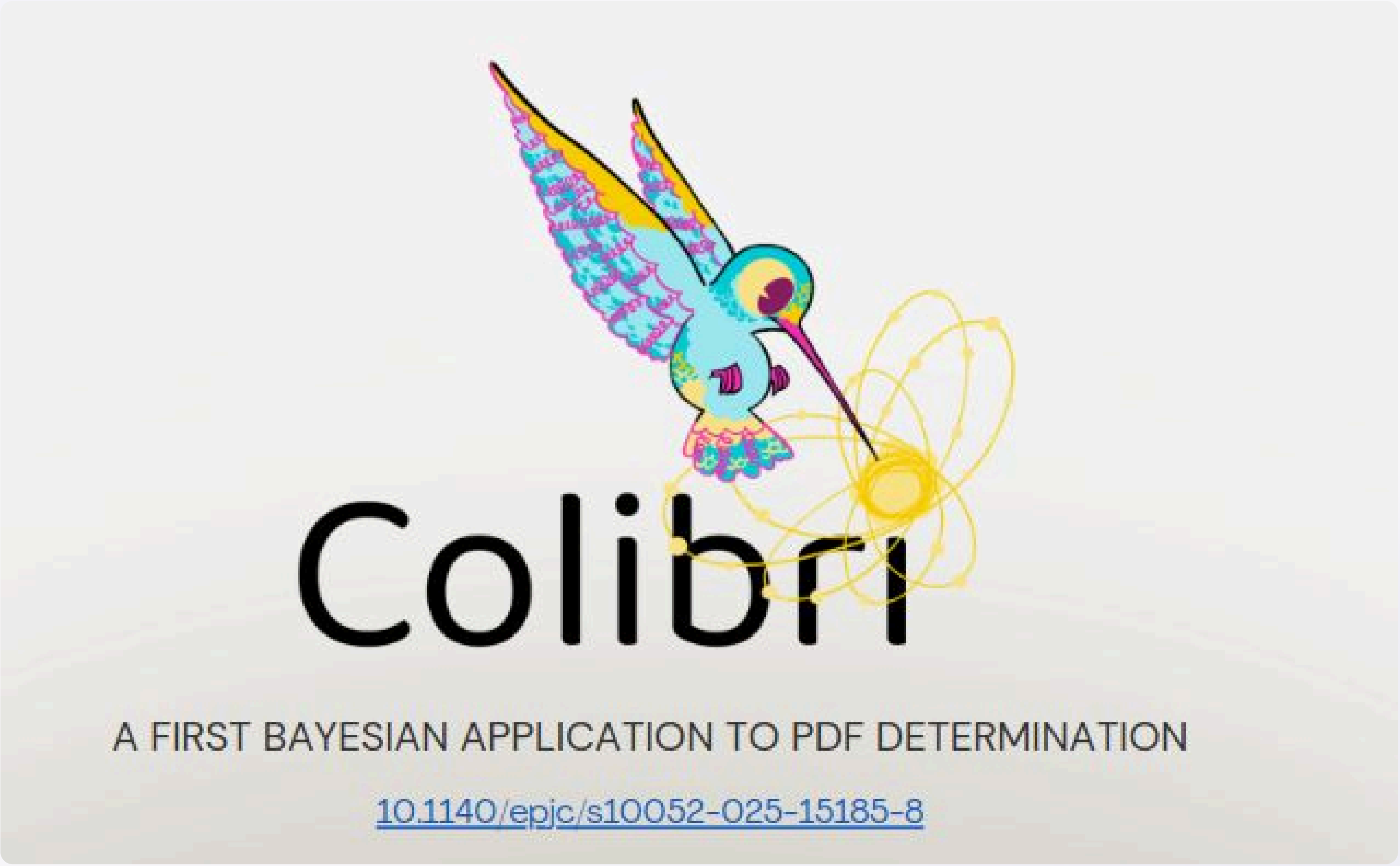
$$p(\boldsymbol{\theta} \mid \mathbf{X}) = \frac{p(\mathbf{X} \mid \boldsymbol{\theta})\; p(\boldsymbol{\theta})}{p(\mathbf{X})}$$

$\boldsymbol{\theta}$: parameters (latent variables)

$\mathbf{X}$: data (observed variables)

- Allows for **unsupervised learning**
- **Classification tasks**: assignment probabilities $p(\text{class}|x)$ (calibrated by definition!)
- **Generative sampling**: sample datasets given posterior
  - Useful for model validation and uncertainty estimation
- How to: build a model? choose the prior? learn the posterior?

Colibri: A new tool for fast-flying PDF fits



- Colibri is a modular tool to calculate and fit parton distribution functions.

- PDF model from polynomial to feed-forward NN

- Live points: set of PDF models w/ prior parameters

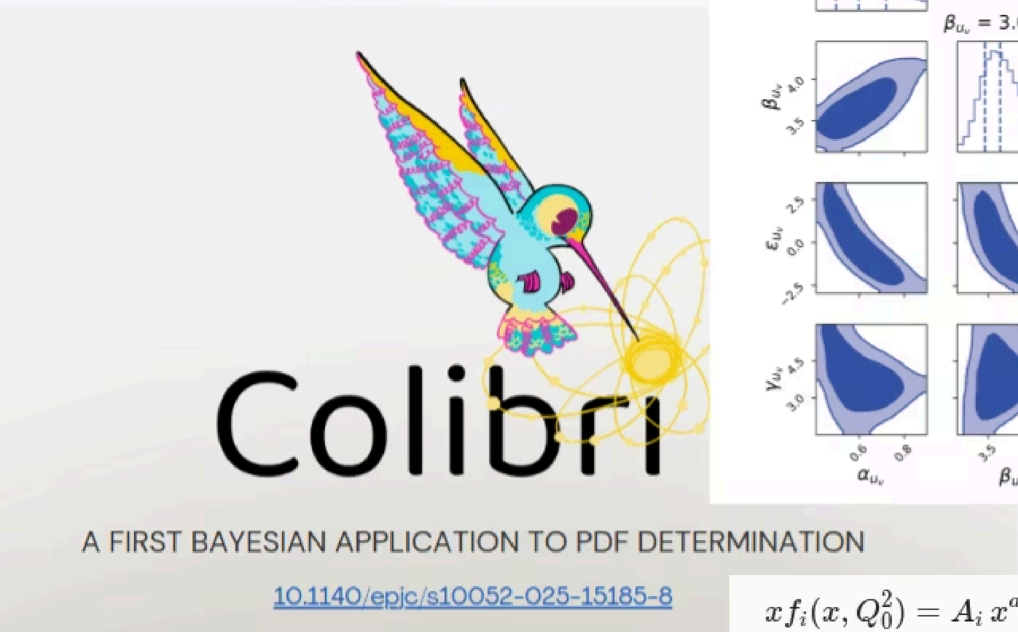
- Likelihood step: compare each of PDF model w/ data

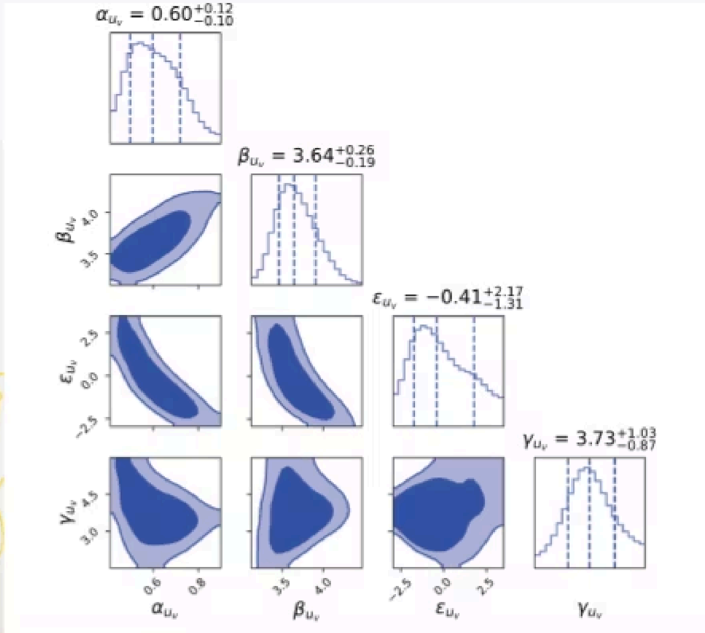
- Bayes step: remove the worst model and produce the new one better

- Posterior as set of weights from model parameters

Prons: honest errors, model selection, speed acceleration

Colibri: A new tool for fast-flying PDF fits





$$xf_i(x, Q_0^2) = A_i\, x^{a_i}(1-x)^{b_i}(1+\epsilon_i\sqrt{x}+\gamma_i x)$$

- Colibri is a modular tool to calculate and fit parton distribution functions.
- PDF model from polynomial to feed-forward NN
- Live points: set of PDF models w/ prior parameters
- Likelihood step: compare each of PDF model w/ data
- Bayes step: remove the worst model and produce the new one better
- Posterior as set of weights from model parameters

Prons: honest errors, model selection, speed acceleration

Split-N-Fit: differentiable maximum likelihood

(NeurIPS 2025)

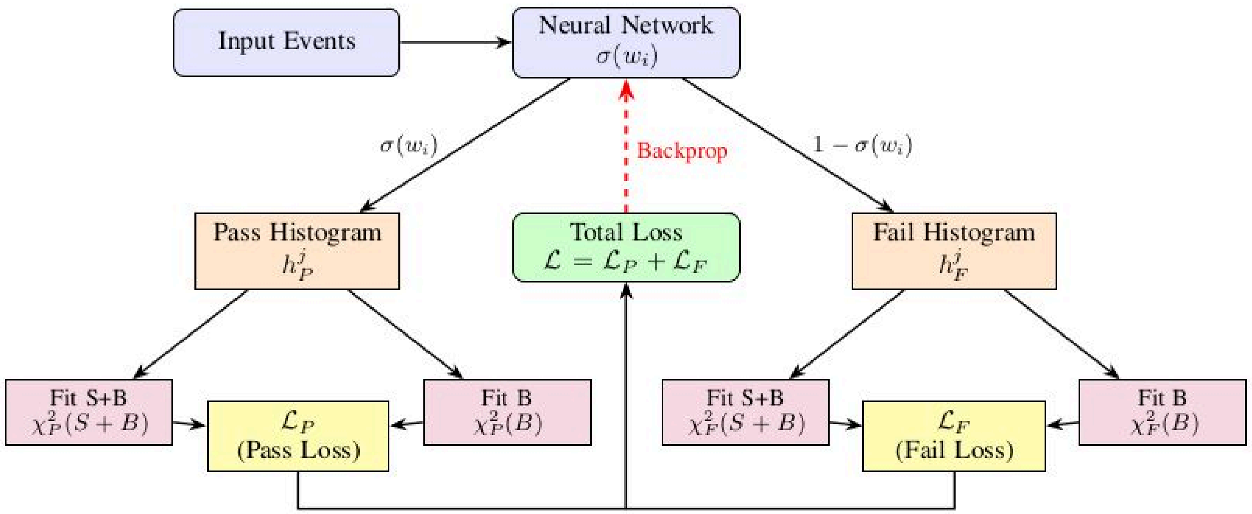


Figure 1: Schematic overview of the Split-N-Fit method. Events are split into pass and fail histograms. Both S+B and B functions are fit to each histogram, yielding χ^2 values that define individual loss terms $\mathcal{L}_P$ and $\mathcal{L}_F$, which combine to form the total differentiable loss.

- ML trained on simulations suffers from modeling uncertainties

- Split-N-Fit allows fine-tuning on data replacing cross entropy loss function with χ^2 fit based

- Split step: each event is split between two histograms: pass and fail

- Fit step: maximum likelihood fit of a function to the histogram yields in both the pass and fail categories

Split-N-Fit: differentiable maximum likelihood

(NeurIPS 2025)

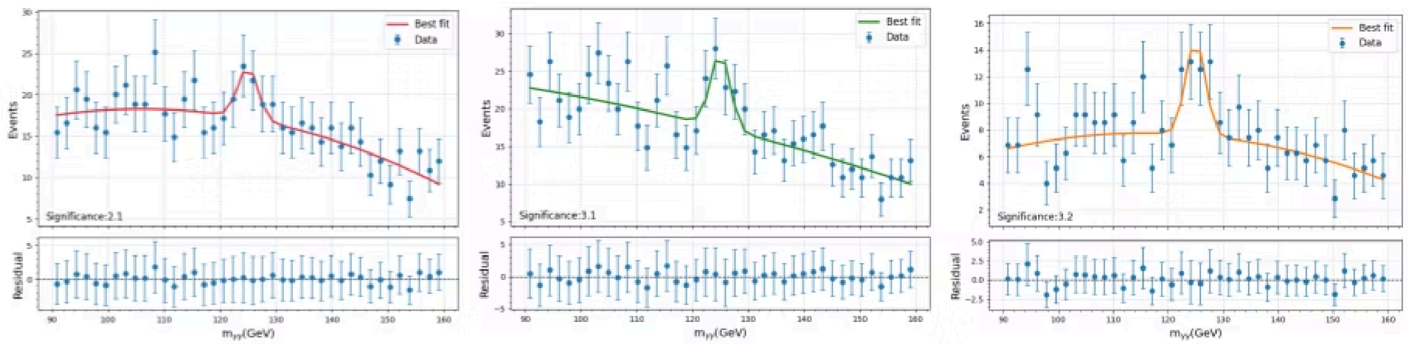


Figure 2: Visualization of the resulting di-Higgs signal in the fitted di-photon mass for (left) selecting events with an MLP trained on the incorrect background, (center), the same incorrect MLP as the left diagram fine-tuned with Split-N-Fit, and (right), an MLP trained with the correct background.

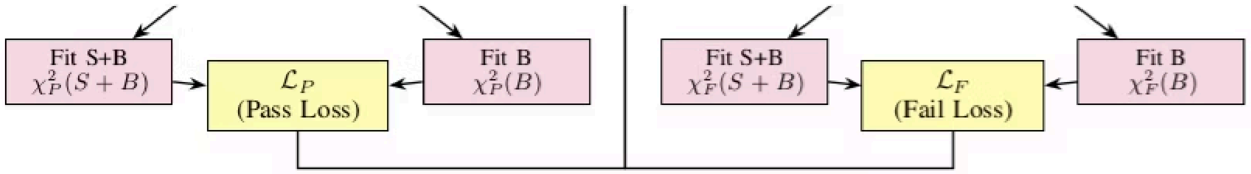


Figure 1: Schematic overview of the Split-N-Fit method. Events are split into pass and fail histograms. Both S+B and B functions are fit to each histogram, yielding χ^2 values that define individual loss terms $\mathcal{L}_P$ and $\mathcal{L}_F$, which combine to form the total differentiable loss.

- ML trained on simulations suffers from modeling uncertainties

- Split-N-Fit allows fine-tuning on data replacing cross entropy loss function with χ^2 fit based

- Split step: each event is split between two histograms: pass and fail

- Fit step: maximum likelihood fit of a function to the histogram yields in both the pass and fail categories

Belle II track reconstruction

(arXiv:2411.13596, arXiv:2602.10897)

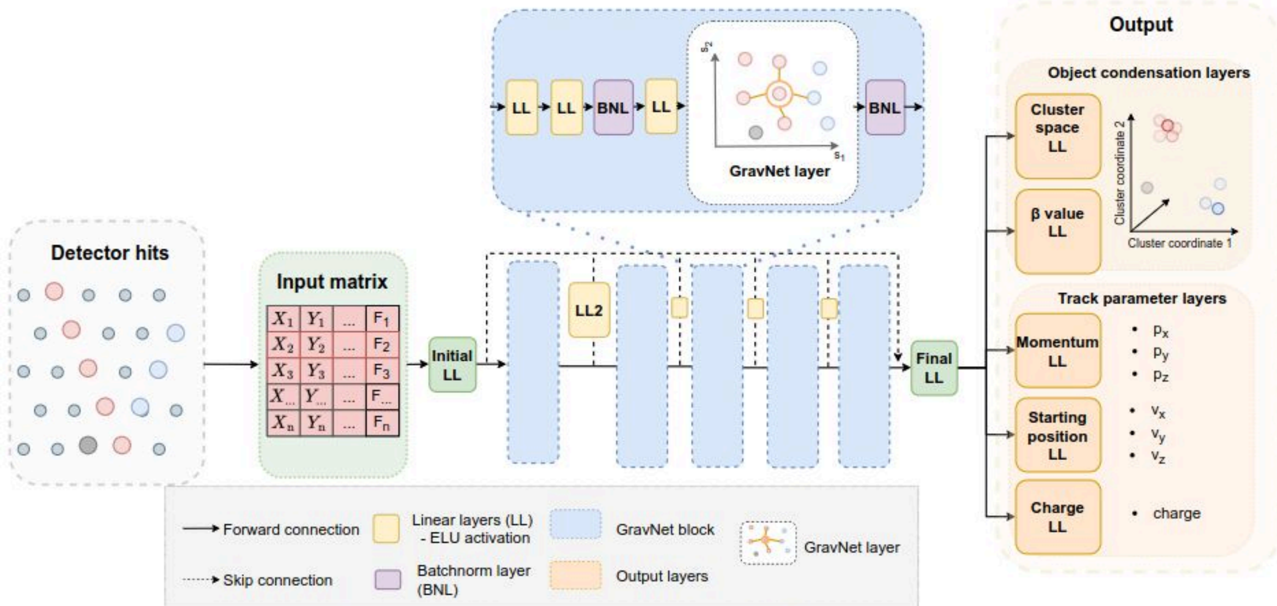
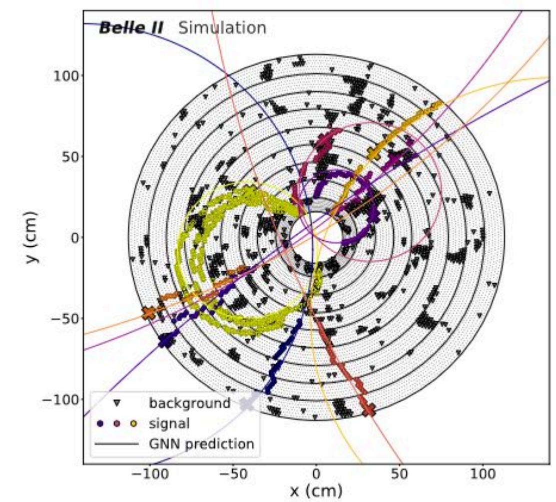


Fig. 2: An illustration of the GNN architecture.



CDC AI Tracking uses detector hits as inputs and simultaneously predicts the number of track candidates in an event

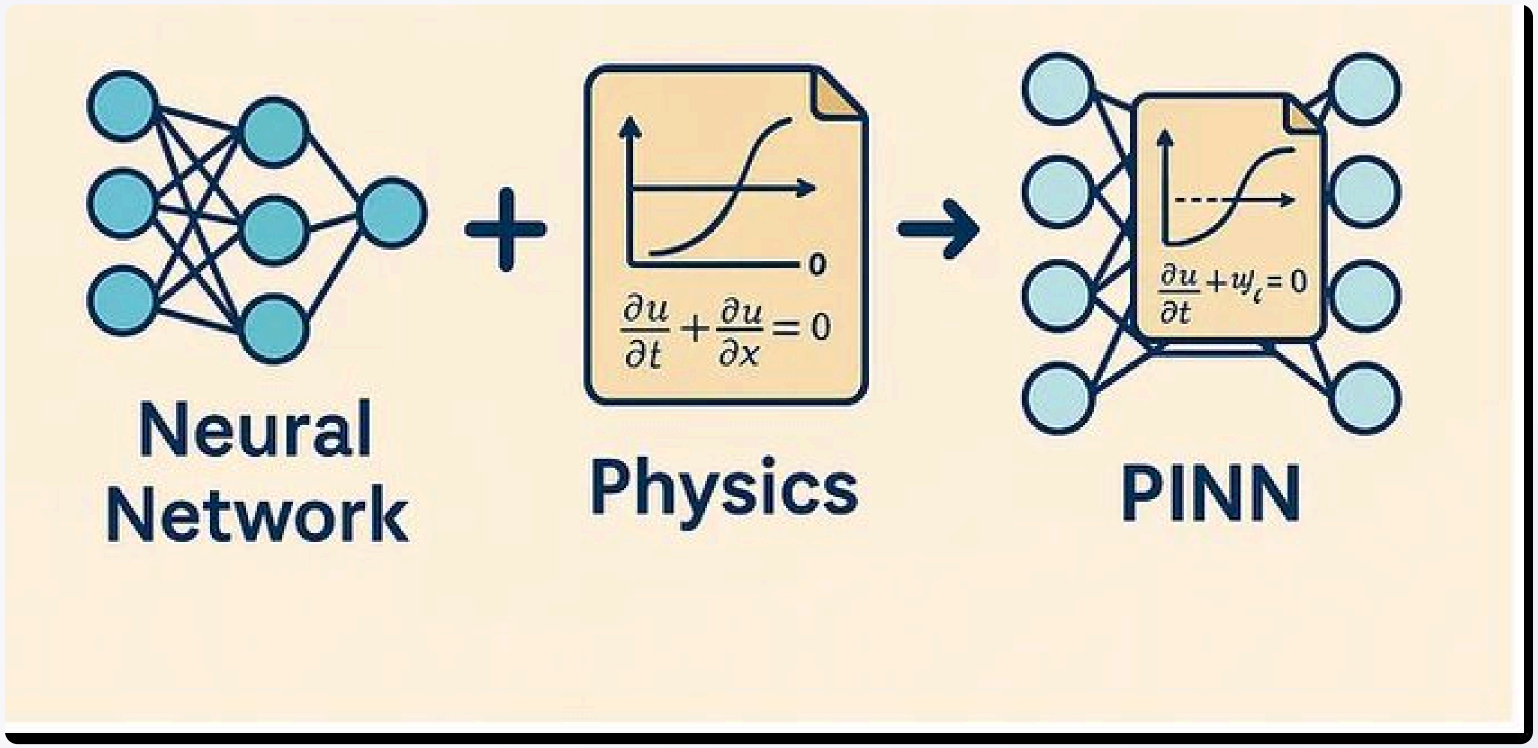
- Combination of object condensation loss and GravNet layer of GNN

- Object condensation loss $L_p = s_c(L_\beta + L_V)$, L_V ***- potential,*** L_β ***- condensation score,*** L_p ***- payload***

- GravNet: graph build in learning representation space

PINN : data-driven solution / discovery of nonlinear partial differential equation

(Journal of Computational Physics 378, 686 (2019))



- *PDE solution as ordinary NN*
- *PDE solution data - only internal data: initial / boundary and collocation*
- *PDE discovery data - external (noisy) + internal data to estimate PDE parameters*

Loss function $L = L_{PDE} + L_{BC}$,

L_{PDE} *- structure imposed by equation,* L_{BC} *- boundary conditions*

PINN : data-driven solution / discovery of nonlinear partial differential equation

Physics-informed neural networks (PINNs) / Neural Operators.

- Equation of State at High Baryon Densities from a Thermodynamically Informed Neural Network (2026)
- Inferring identified hadron production in pp collisions with physics-informed machine learning at the LHC (2026)
- Lattice fermion formulation via Physics-Informed Neural Networks: Ginsparg-Wilson relation and Overlap fermions [DOI] (2026)
- Physics-Informed Neural Networks for Solving Two-Flavor Neutrino Oscillations in Vacuum and Matter Environments for Atmospheric and Reactor Neutrinos (2026)
- Four-dimensional QCD equation of state from a quasi-parton model with physics-informed neural networks (2026)
- Probing Proton Structure via Physics-Guided Neural Networks in Holographic QCD (2026)
- Determining the NJL Coupling and AMM in Magnetized QCD Matter via Machine Learning (2026)
- Physics-Informed Global Extraction of the Universal Small-x Dipole Amplitude (2026)
- Solving stiff dark matter equations via Jacobian Normalization with Physics-Informed Neural Networks (2026)
- Wilson loops with neural networks [DOI] (2026)
- Parton Fragmentation Functions Extracted with a Physics-Informed Neural Network (2026)
- Heavy Quarkonium Spectrum and Decay Constants from a Neural-Network-Based Holographic Model [DOI] (2026)
- Universality of Gluon Saturation from Physics-Informed Neural Networks [DOI] (2026)
- Physics-informed neural networks for angular-momentum conservation in computational relativistic spin hydrodynamics (2026)
- AdS/Deep-Learning made easy II: neural network-based approaches to holography and inverse problems (2025)
- Physics-informed neural network (PINN) modeling of charged particle multiplicity using the two-component framework in heavy-ion collisions: A comparison with data-driven neural networks [DOI] (2025)
- Physics-informed continuous normalizing flows to learn the electric field within a time-projection chamber [DOI] (2025)
- Spectral functions in Minkowski quantum electrodynamics from neural reconstruction: Benchmarking against dispersive Dyson–Schwinger integral equations (2025)
- QINNs: Quantum-Informed Neural Networks (2025)
- Deep Neural Network extraction of Unpolarized Transverse Momentum Distributions [DOI] (2025)
- Aspects of holographic entanglement using physics-informed-neural-networks (2025)
- Physics Informed Neural Networks for design optimisation of diamond particle detectors for charged particle fast-tracking at high luminosity hadron colliders (2025)
- Physics-informed neural network solves minimal surfaces in curved spacetime [DOI] (2025)
- Fast prediction of the hydrodynamic QGP evolution in ultra-relativistic heavy-ion collisions using Fourier Neural Operators [DOI] (2025)
- Physics-Informed Neural Network Approach to Quark-Antiquark Color Flux Tube [DOI] (2025)

PINN for solving two-flavor neutrino oscillations

(arXiv:2604.22862)

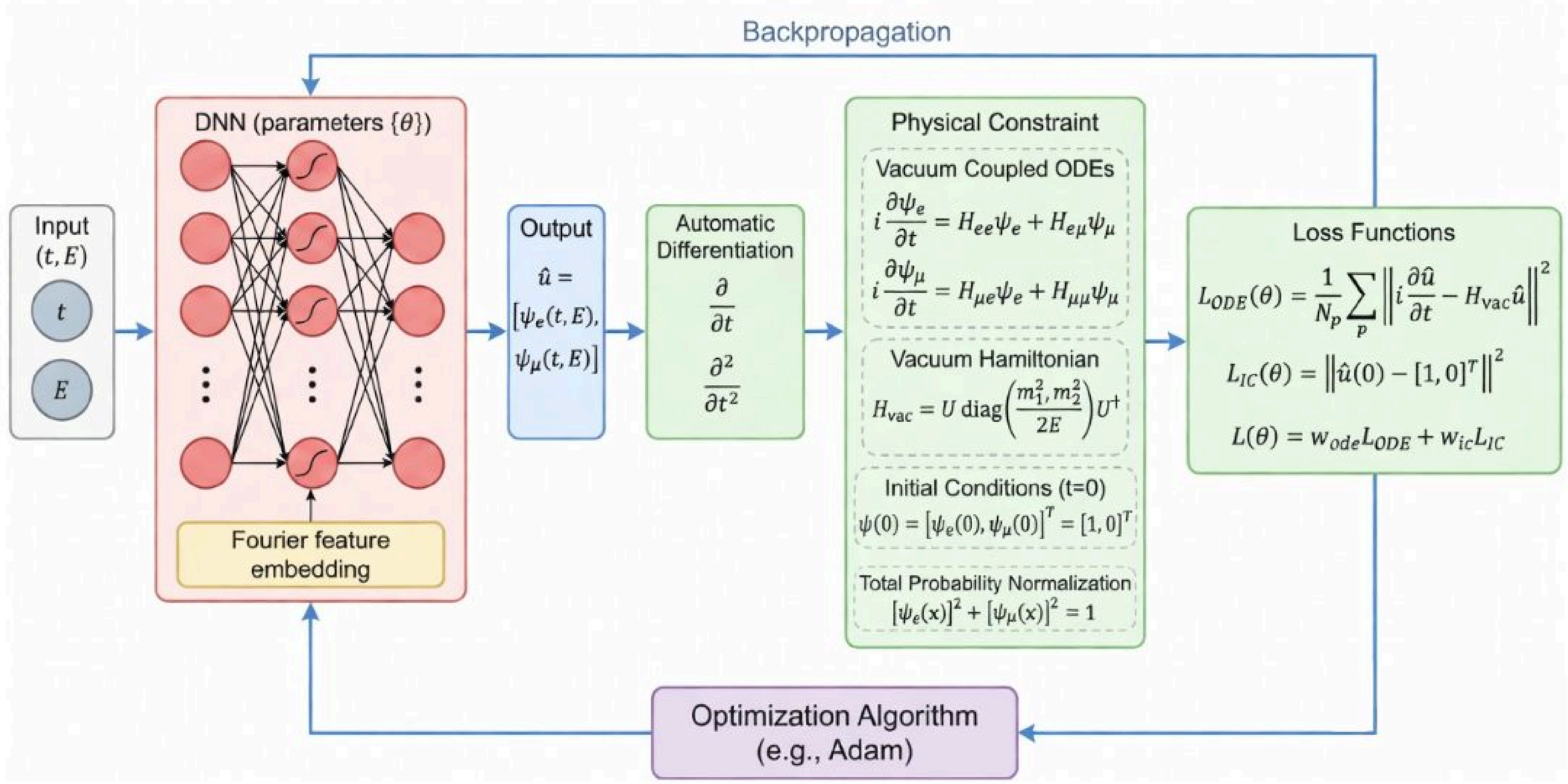


FIG. 1. PINNs architecture for neutrino propagation in vacuum.

PINN predictions are compared to the standard analytical solution

PINN for solving two-flavor neutrino oscillations

(arXiv:2604.22862)

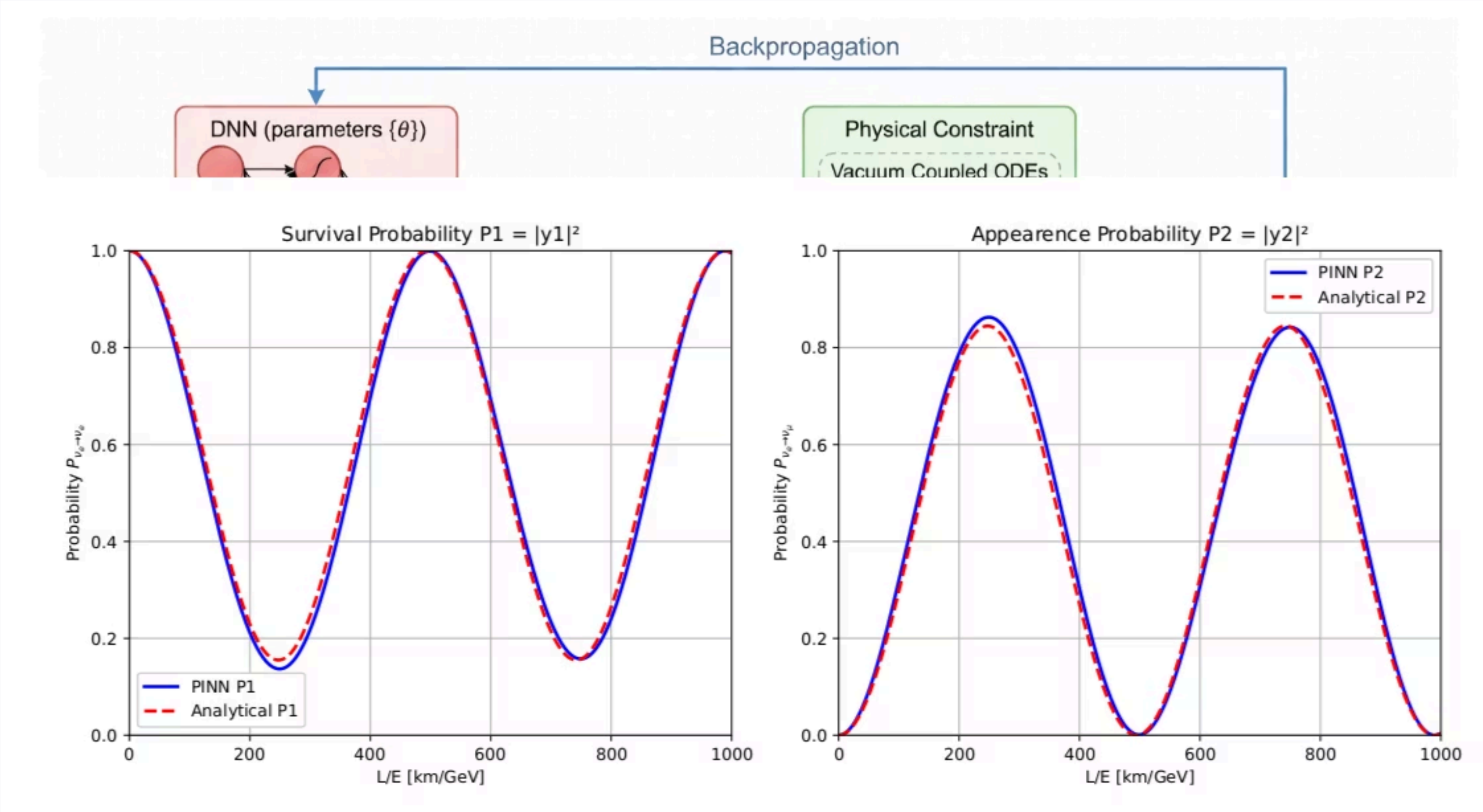


FIG. 3. Survival and Appearance Probability of Reactor neutrino in vacuum case

PINN predictions are compared to the standard analytical solution

newPINN: couple with conventional numerical solvers

(arXiv:2601:17207)

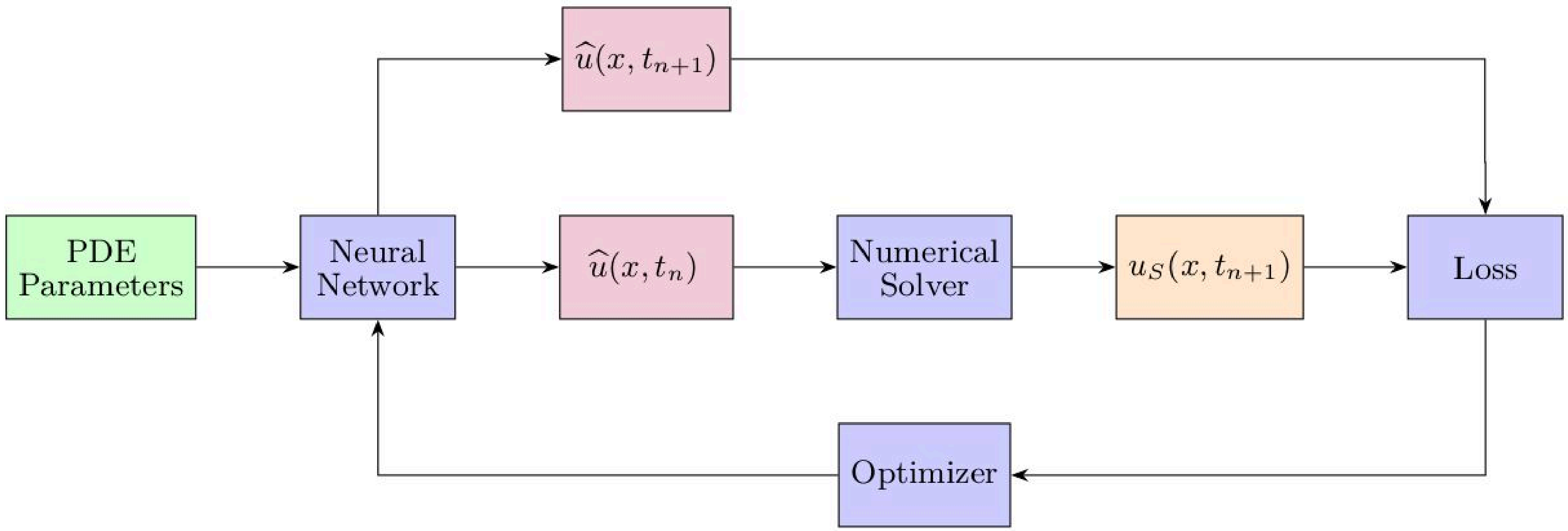
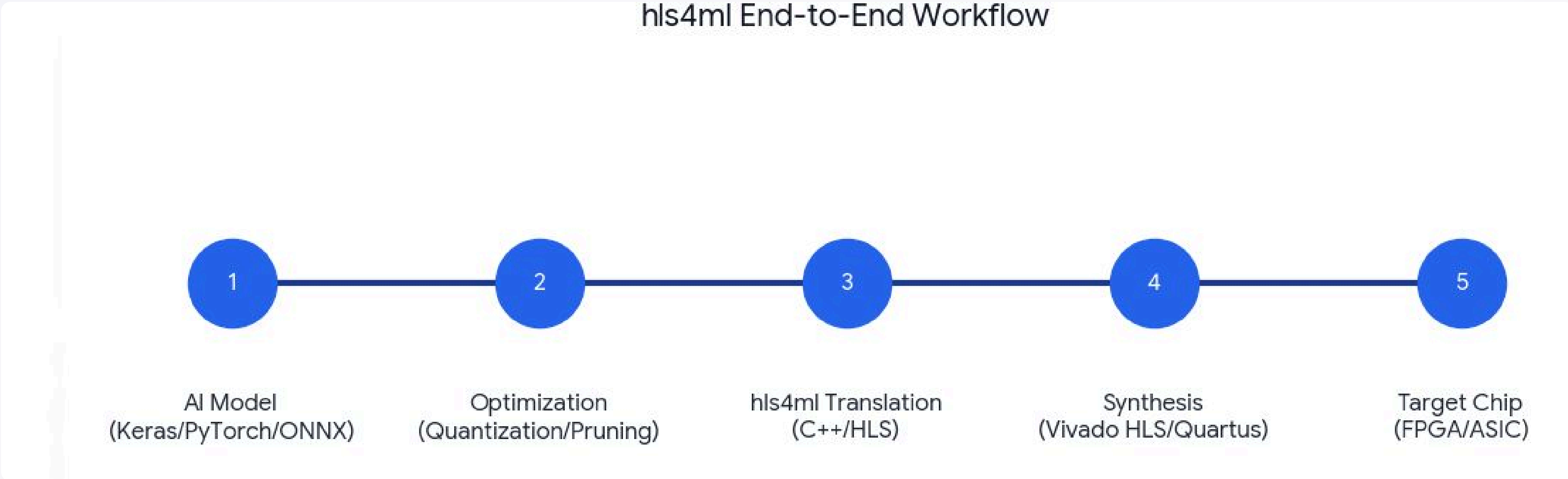


Figure 1: Transient NewPINNs: At each training iteration, the neural network produces a candidate solution at time t_n which is advanced to t_{n+1} by the numerical solver. A solver-consistency loss compares this solver-evolved state with the network's prediction at the next time step, enforcing temporal consistency under solver evolution. The solver is treated as a black-box operator and does not participate in gradient computation.

Instead of forcing the network to minimize a PDE residual derived via automatic differentiation, NewPINNs couples the network with a traditional numerical solver in a "pull-push" closed-loop solver-consistency training procedure

hls4ml: fast deep learning for real-time HEP application

(arXiv:2512:01463)



hls4ml is a paradigm to convert DL framework into full designs for FPGAs / ASICs to accelerate ML inference from 1 msec (as it is on CPU/GPU) up to 100 nsec (as it is on FPGA)

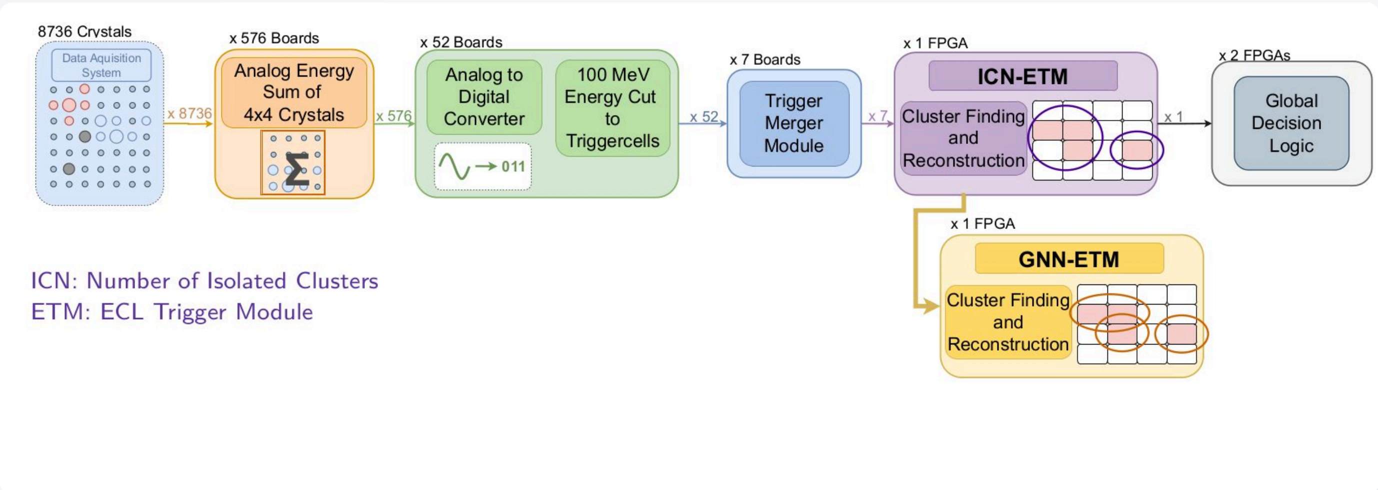
- Optimization - compress the model to keep chip space

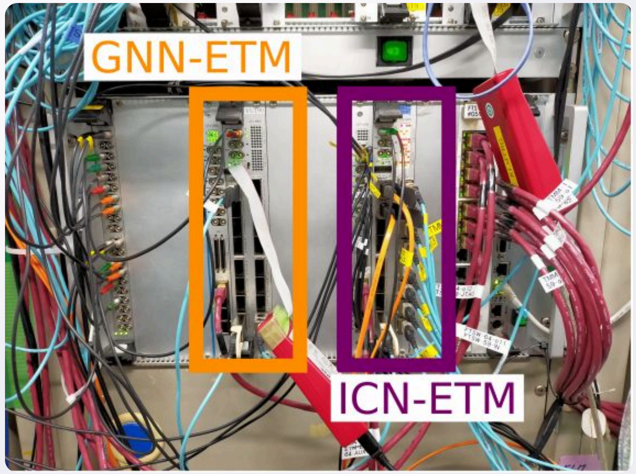
- Translation (high-level synthesis) - convert into C++ code

- Synthesis - translate C++ to byte firmware

Real-time GNN on FPGAs for the BELLE II calorimeter trigger

(JINST 21 P06039 (2026))

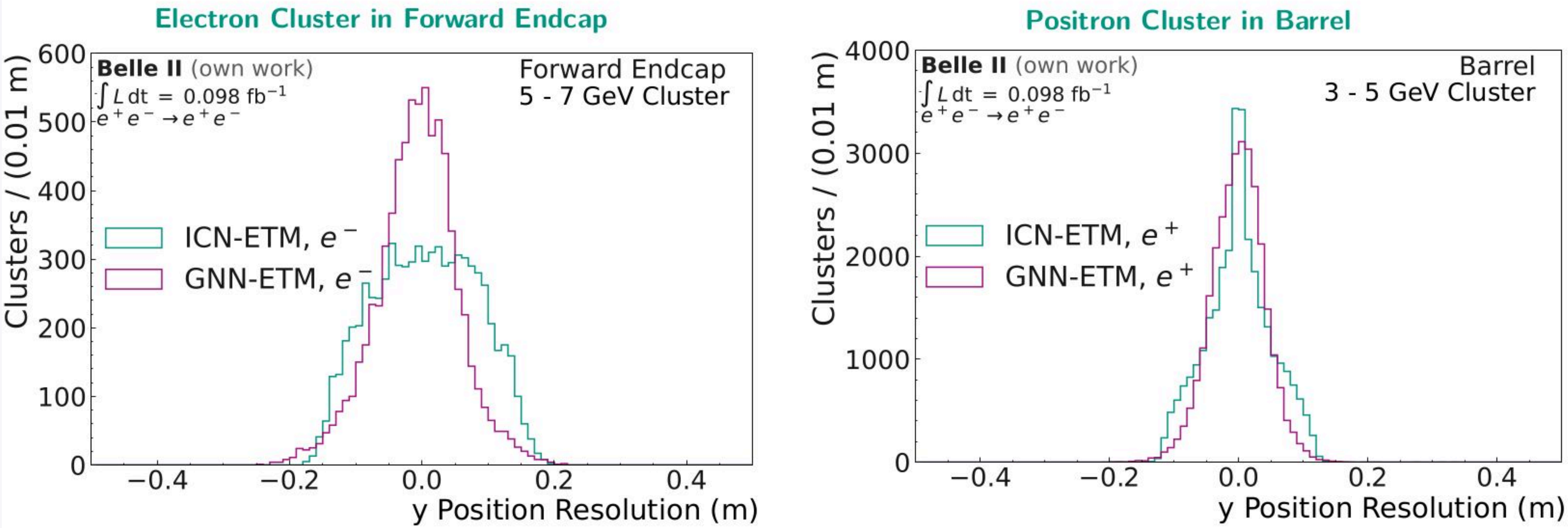




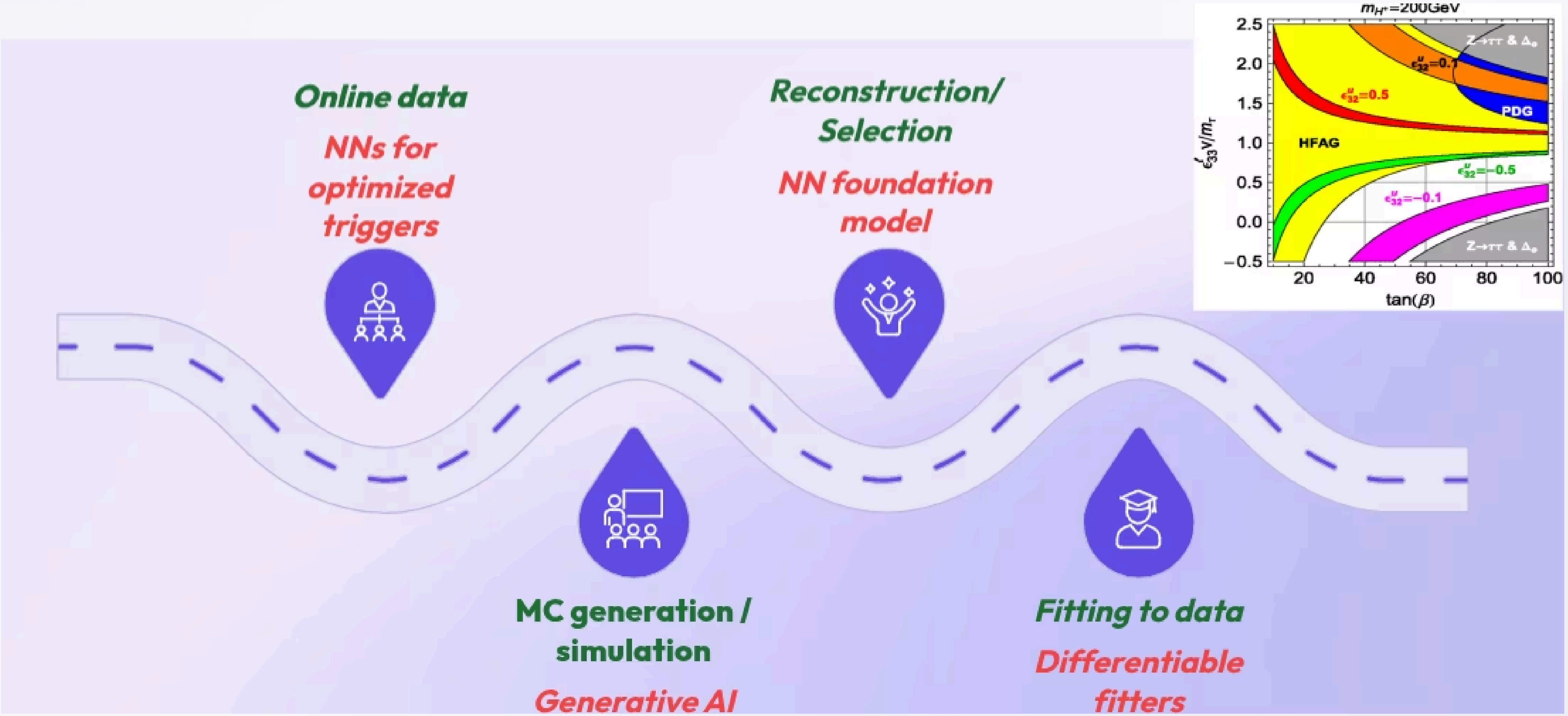
- GNN w/ similar architecture as for the Belle II tracking reconstruction is implemented in AMD Virtex UltraScale XCVU190 FPGA (2.3 M logic cells, 2k DSP)
- The design achieves the target throughput of 8 MHz and latency of less than 3.1 μsec

Real-time GNN on FPGAs for the BELLE II calorimeter trigger

(JINST 21 P06039 (2026))

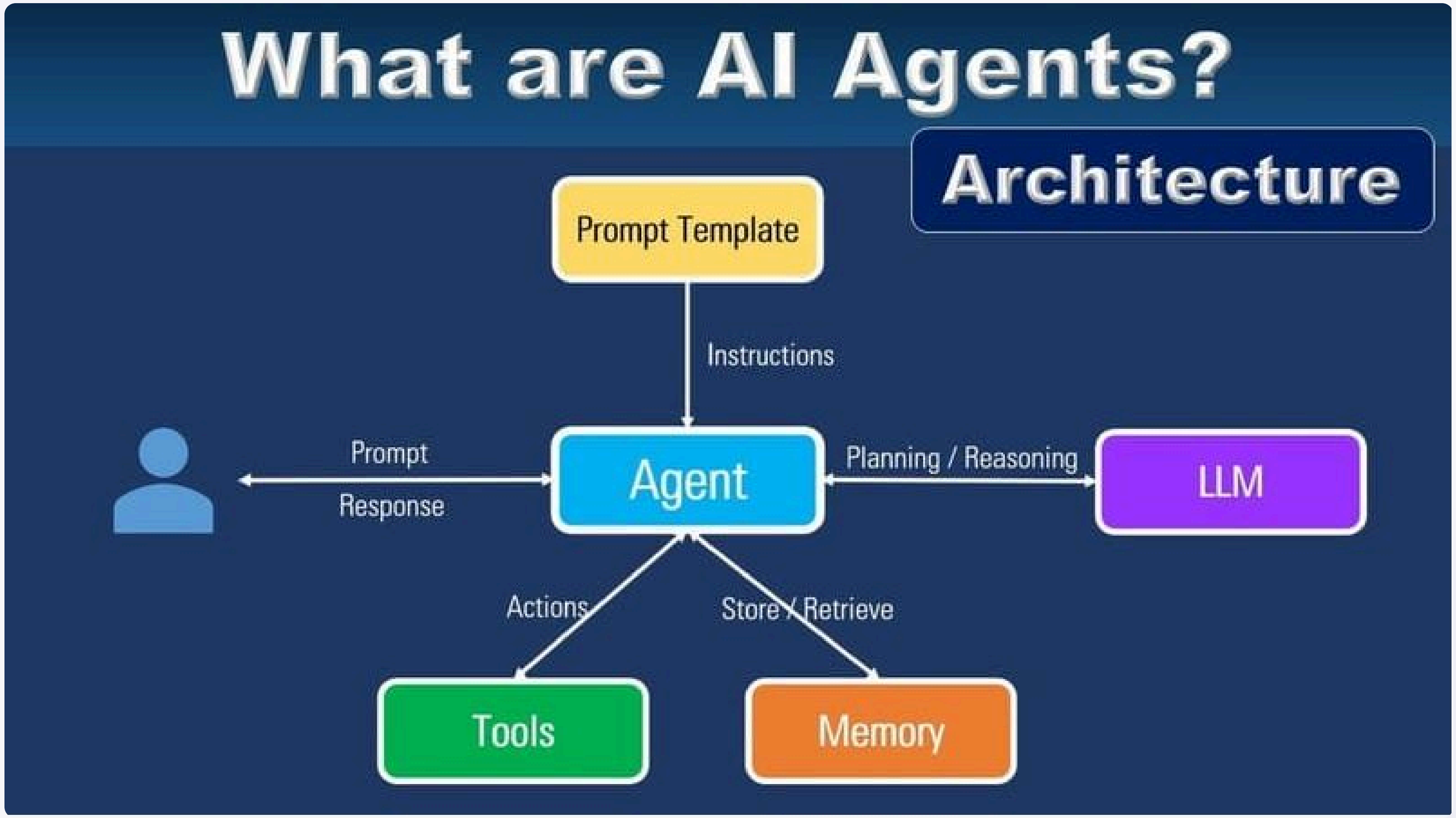


HEP analysis flows w/ ML appeared in all of these



AI agents as global AI

AI Agents are systems that can interact with the environment and make decisions in the real world without any human guidance or intervention.



When receiving a command (goal) from a user (Prompt), AI Agents initiate the goal analysis process, transferring the prompt to the core AI model (typically a LLM) and beginning to plan actions. The Agent will break down complex goals into specific tasks and subtasks, with clear priorities and dependencies. During implementation AI agents collect information from various sources, analyze information and calculate necessary actions. Throughout this process, the agent's Memory continuously stores information.

AI agents as global AI

Building AI Agents on LLMs Is a Recipe for Disaster

Yann Lecun

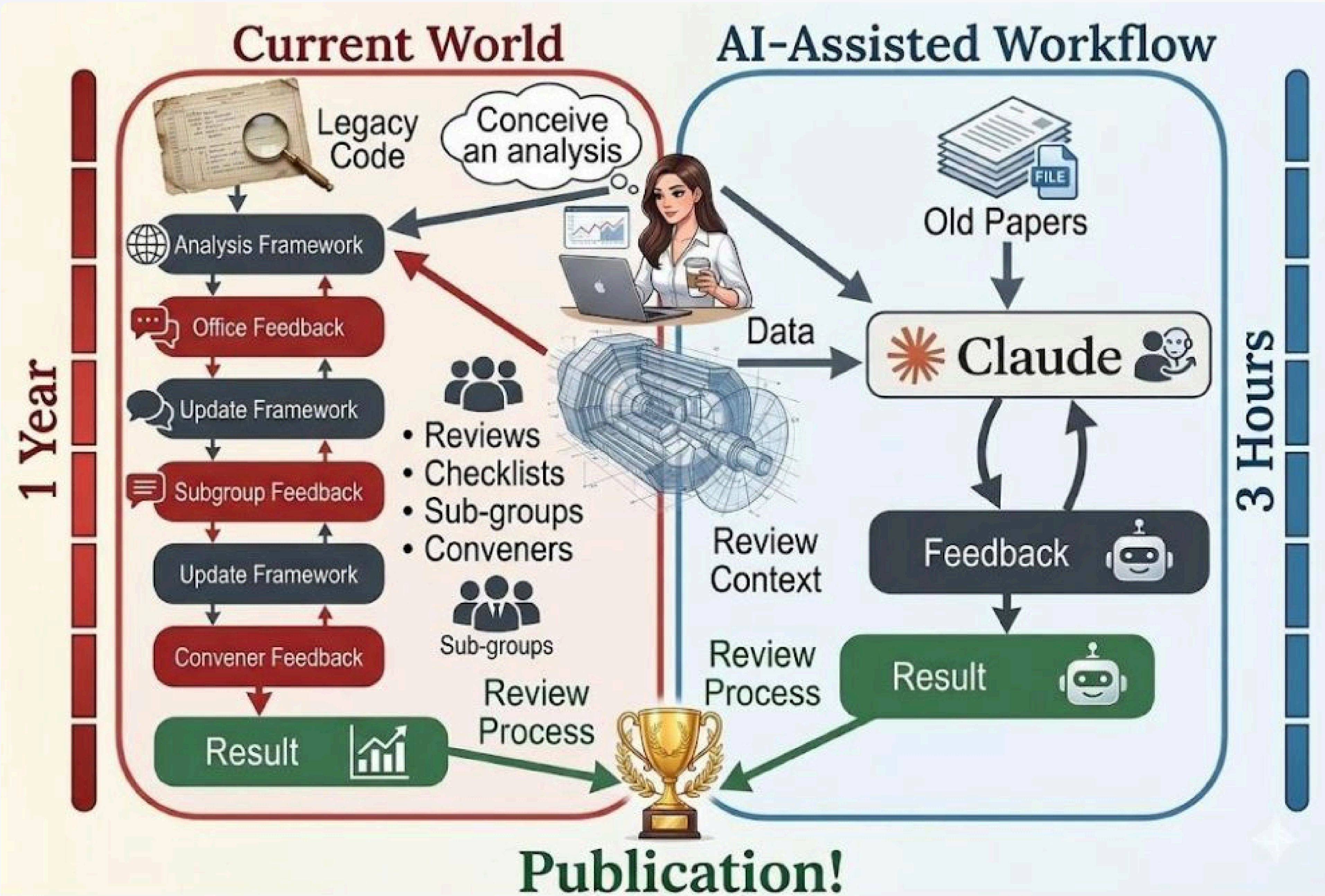
AI foundation father

Text is not merely words.
It is an imprint of our world.

Google
OpenAI
Anthropic

AI agents as global AI to automate HEP pipeline ?

(https://jfc-mit.github.io/)



LLM-based AI agents are now able to autonomously execute substantial portions of a high energy physics (HEP) analysis pipeline with minimal expert-curated input. Given access to a HEP dataset, an execution framework, and a corpus of prior experimental literature, we find that Claude Code succeeds in automating all stages of a typical analysis: event selection, background estimation, uncertainty quantification, statistical inference, and paper drafting.

AI agents as global AI to automate HEP pipeline ?

Automating High Energy Physics Data Analysis with LLM-Powered Agents (arXiv:2512.07785)

Agents of discovery (arXiv:2509.08535)

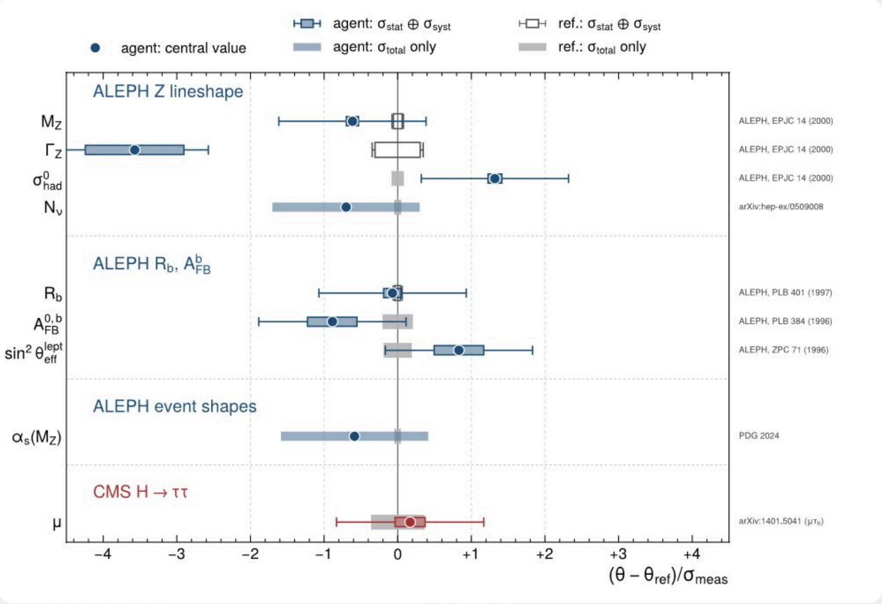
Agentic AI – Physicist Collaboration in Experimental Particle Physics: A Proof-of-Concept Measurement with LEP Open Data (arXiv:2603.05735)

SciFi: A Safe, Lightweight, User-Friendly, and Fully Autonomous Agentic AI Workflow for Scientific Applications (arXiv:2604.13180)

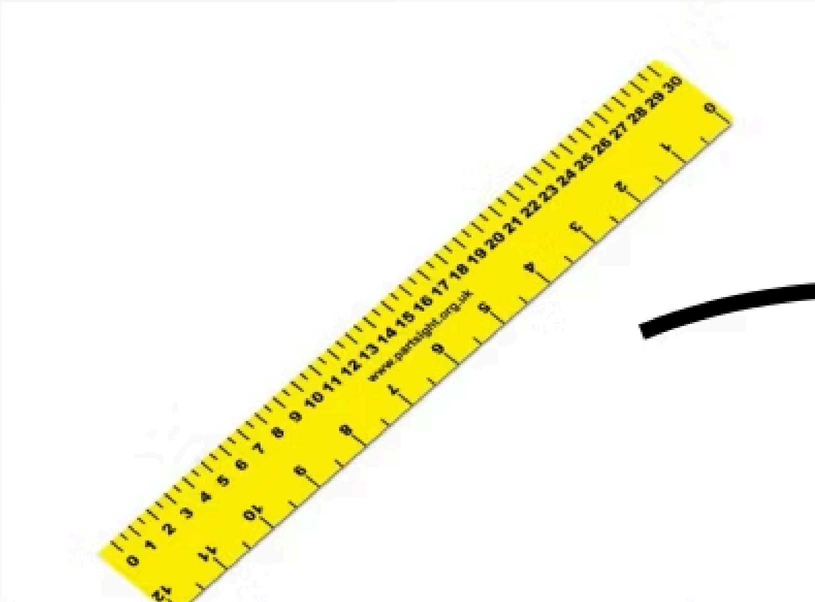
Collider-Bench: Benchmarking AI Agents with Particle Physics Analysis Reproduction (arXiv:2605.13950)

AI Agents Can Already Autonomously Perform Experimental High Energy Physics (arXiv:2603.20179)

Rather than replacing physicists, these tools promise to offload the repetitive technical burden of analysis code development, freeing researchers to focus on physics insight, truly novel method development, and rigorous validation.



Summary



AI is definitely useful to transfer routine work from the physicist to the AI.

But we must sense the limits of this tool's applicability so that artificial intelligence does not displace natural intelligence