EuCAIFCon 2025: Accelerating TimeSPOT Detector Simulation with Deep Learning

Zhihua Liang (INFN Cagliari)
On behalf of the INFN Cagliari

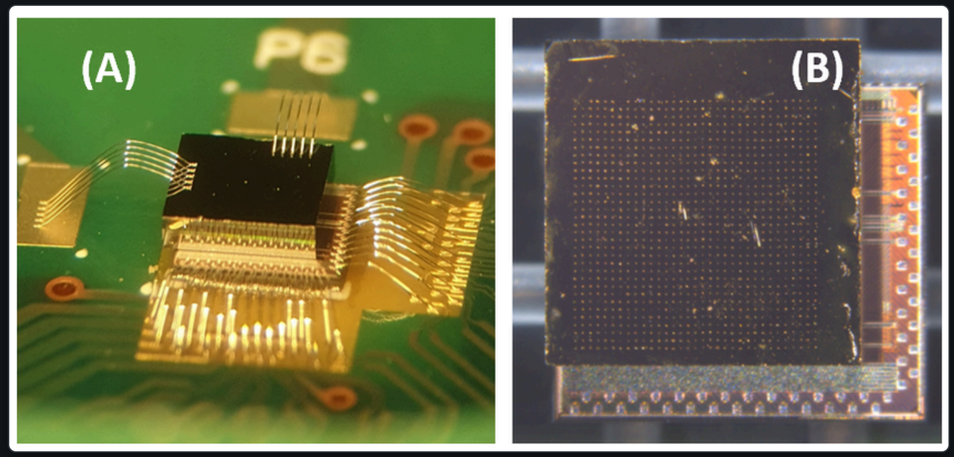
A Multi-Scale Challenge: From Microscopic Physics to Macroscopic Signals

The Physics Problem

- Next-generation particle detectors, like the **TimeSPOT 4D sensor**, must measure particle trajectories with extreme precision in both space and time (<50 picoseconds).
- Our “ground truth” for R&D is a detailed physics simulation that models charge transport within silicon (via engines like Geant4 & TCoDe).

The Computational Bottleneck

- This high-fidelity simulation is computationally intractable:
  - **1 particle hit ≈ 1 second of CPU time.**
- This prevents us from generating the billions of events needed for research.
- **The ML Task:** Learn the mapping from **microscopic interactions (UV)** — like a particle’s path and energy loss—to **emergent detector signals (IR)**, such as total charge and signal timing.



The TimeSPOT sensor: a complex physical system.

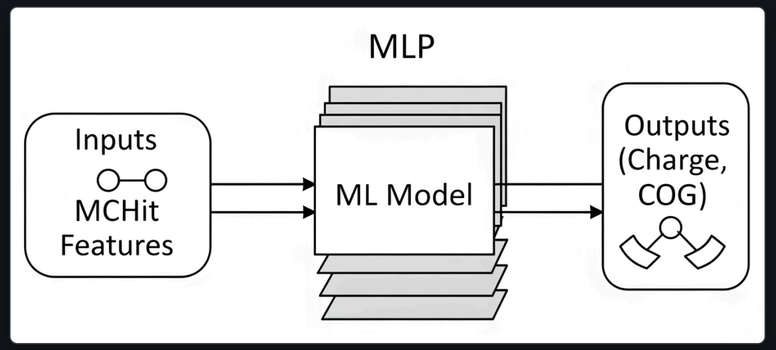
The simulation cost is a critical roadblock to scientific progress.

Our Approach: Compressing Physics with a Neural Surrogate

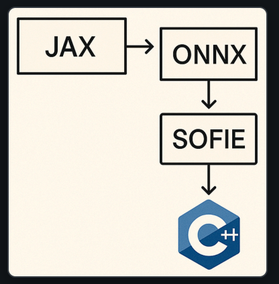
Learning an Effective Function

We trained a deep Multi-Layer Perceptron to act as a **surrogate** for the slow physics engine.

- **Inputs:** ~12 features describing particle kinematics & hit geometry.
- **Outputs:** The two key detector observables:
  1. Total Collected Charge
  2. Temporal Center of Gravity (Timing)
- **Architecture Rationale:** A symmetric “hourglass” MLP first projects inputs into a high-dimensional latent space to capture complex physical correlations, then compresses this representation to the target outputs.



Replace the slow physics engine with a fast, learned function.



A robust pipeline from research to production.

From Python Research to C++ Production

- Trained in **JAX**, exported to the standard **ONNX** format.
- **ROOT’s SOFIE** tool then auto-generates highly-optimized C++ code, creating a dependency-free engine for direct integration into our C++-based scientific software stack.

High-Fidelity Results & The Path Forward

1. Accuracy: The Model is Unbiased

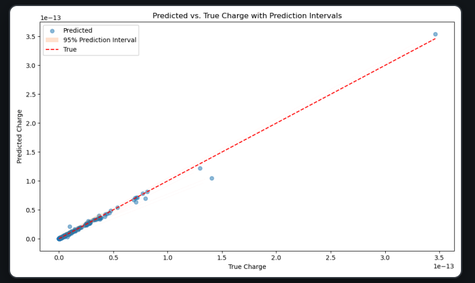
- The model faithfully reproduces the ground truth simulation.
- **Charge Prediction:** $R^2 > 0.99$
- **Timing Prediction:** $R^2 > 0.93$
- Residuals are centered at zero, indicating no systematic bias.

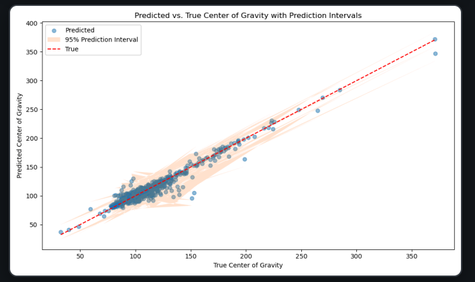
2. Speed: A Transformative Leap

- The surrogate is **~100,000x** faster than the full simulation.
- This turns a computational bottleneck into an on-demand tool, unlocking new research possibilities.

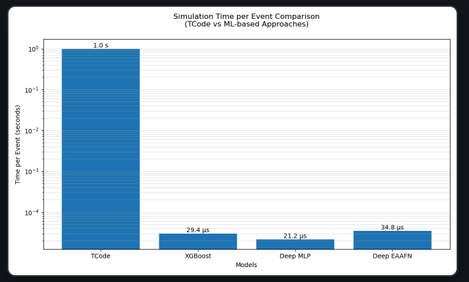
The Path Forward: Modeling Interactions

- **Next Step:** Model inter-pixel charge sharing. This requires moving from a simple MLP to a **Graph Neural Network (GNN)** to capture the system's topological structure.
- **Come see my Poster [Session A]** for architecture details, full validation plots, and a discussion on our software integration!





Unbiased predictions for both charge and timing.



A massive ~100,000x speedup.