Refining Fast Simulation Using Machine Learning

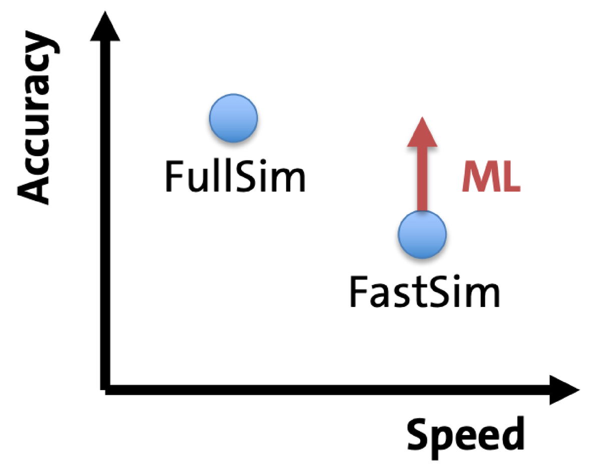
Acelya Deniz Güngördü [1] Dorukhan Boncukcu [1]
Kevin Pedro [2] Samuel Louis Bein [3] Moritz Jonas Wolf [4]
Mirac Ekim Vural [1] Altan Cakir [1]

[1]Istanbul Technical University [2]Fermilab [3]UCLouvain

[4]University of Hamburg

Introduction

- CMS physics analysis rely on large quantities of simulated data
- LHC Phase 2: higher luminosity, complex new detectors, more data
- CMS experiment uses two simulation chains:
  1. **FullSim**: Based on Geant4, high accuracy but slower
  2. **FastSim**: Approximate techniques, faster but less accurate
- FastSim is a rapid Monte Carlo application for detector simulation and event reconstruction, approximately 10 times faster than FullSim.
- FastSim's speed advantage comes with reduced accuracy in some observables.
- R&D presented: Refine FastSim outputs using ML



Data Sample and Method

Training sample: SUSY simplified model „T1tttt" simulated:

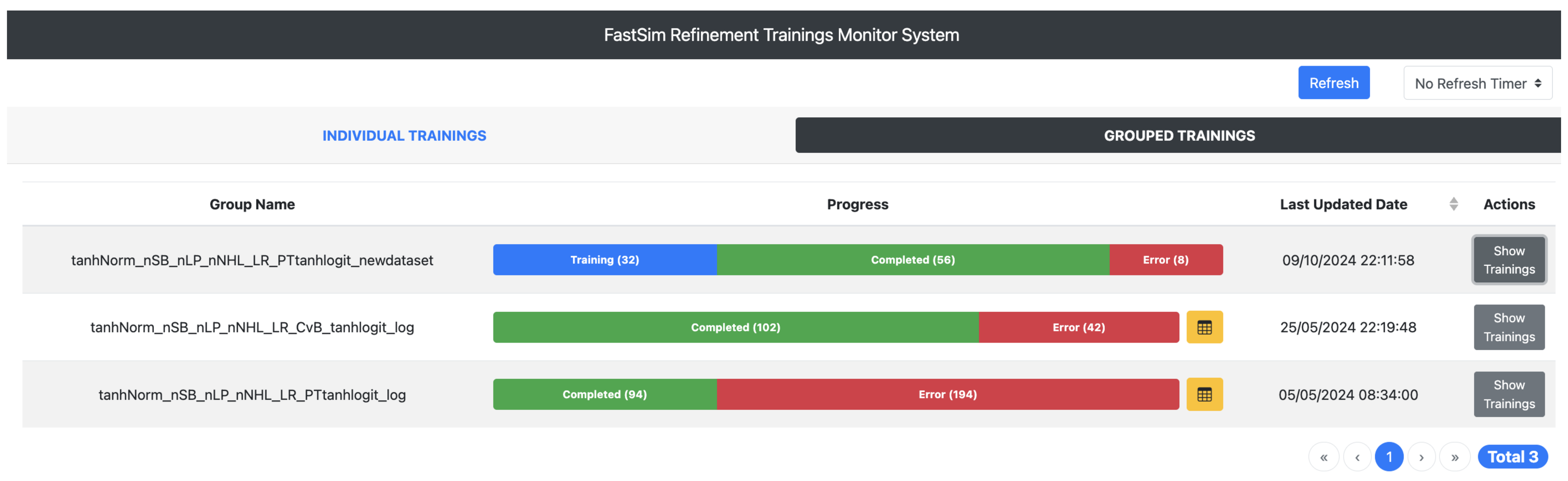
1. Gen $\rightarrow$ FastSim + PU
2. Same Gen $\rightarrow$ FullSim + PU

The aim is to establish a refined version of the FastSim data sample, which is more similar to the FullSim output, i.e., more accurate.

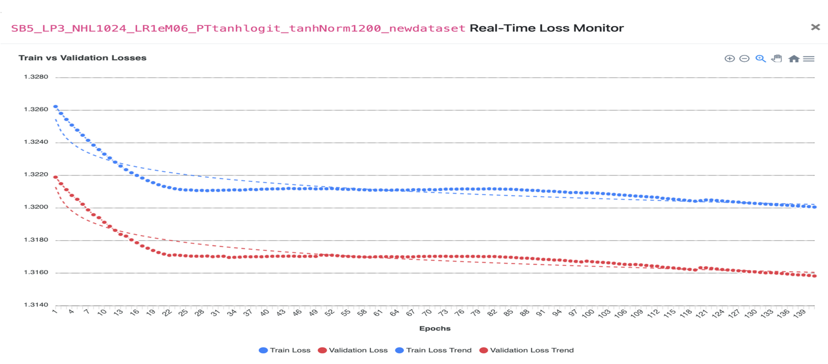
Matching jets using ΔR angular criterion

- **Network Inputs and Targets:**
  Input: FastSim variables $\mathbf{x}^{\text{Fast}}$ = 4 DeepJet discriminators and p_{T}, $\vec{x} = (p_{\text{T}}\ b\ \text{CvB}\ \text{CvL}\ \text{QvG})^{\intercal}$
  Parameters: $\mathbf{y} = p_{\text{T}}^{\text{GEN}}, \eta^{\text{GEN}}$, true hadron flavor (b, c, or light quark/gluon)
  Output: Refined variables $\mathbf{x}^{\text{Refined}}$ = 4 DeepJet discriminators and p_{T}
  Target: FullSim variables $\mathbf{x}^{\text{Full}}$ = 4 DeepJet discriminators and p_{T}

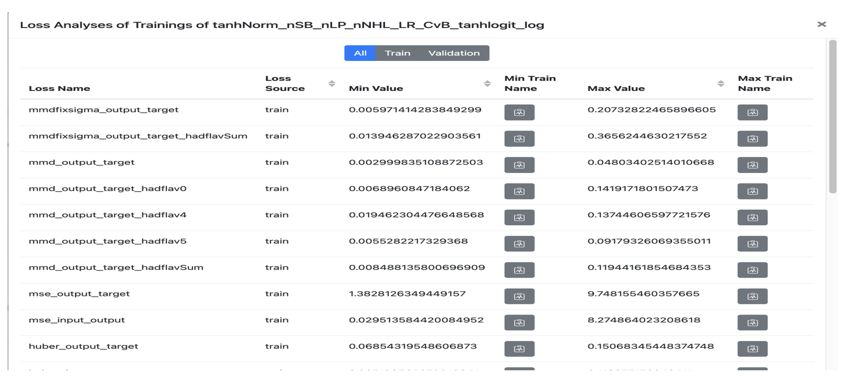
Training Framework



(a) Grid Search Mechanism



(b) Real-Time Loss Monitor



(c) Comparing of Losses in a Grid Search

Figure 1. Training Framework

- A grid search system was integrated into the monitoring system.
- All models in the grid are displayed in the grouped trainings tab as a grouped training.
- Summary table lists best models and all loss values

Conclusion

Refinement of FastSim leads to significantly improved agreement with FullSim.

Training monitoring system implemented to track progress across various training sessions.

The refinement of different variables, such as electrons, muons, and jet p_{T},ongoing.

Examination of the different outcomes produced by various input variable transformations.

References

[1] S. Bein, P. Connor, K. Pedro, P. Schleper, and M. Wolf. Refining fast simulation using machine learning. In *EPJ Web of Conferences*, volume 295, page 09032. EDP Sciences, 2024.

This manuscript has been authored by Fermi Research Alliance, LLC under Contract No. DE-AC02-07CH11359 with the U.S. Department of Energy, Office of Science, Office of High Energy Physics.

Network Architecture

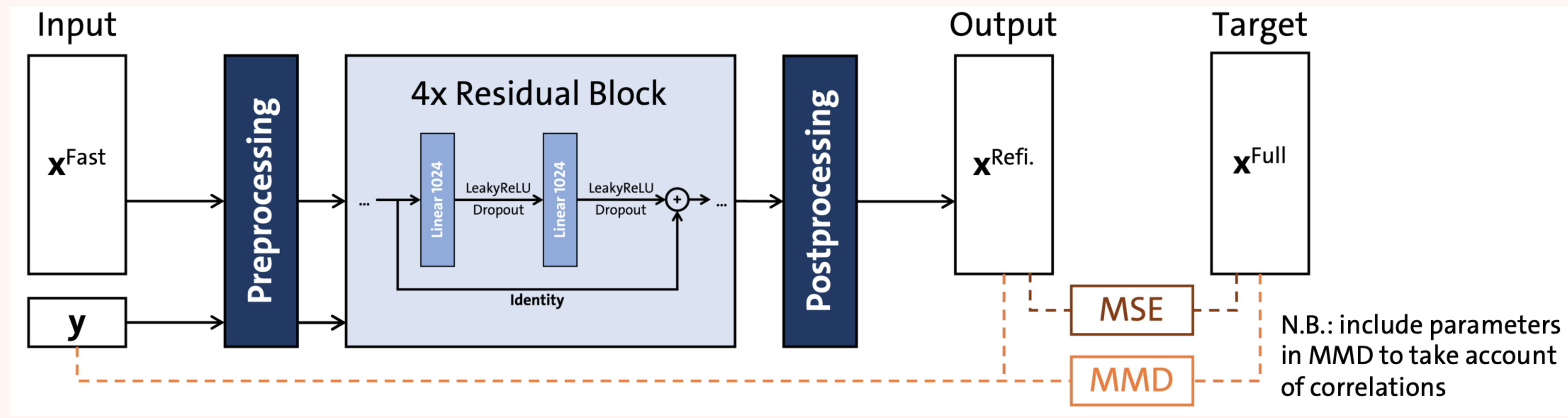


Figure 2. Network Architecture of Refinement

Primary loss: Maximum Mean Discrepancy (MMD)

- Comparing ensembles of jets
- To cope with independent stochasticity in both simulation chains

Given two samples from $P(X)$ and $Q(Y)$:

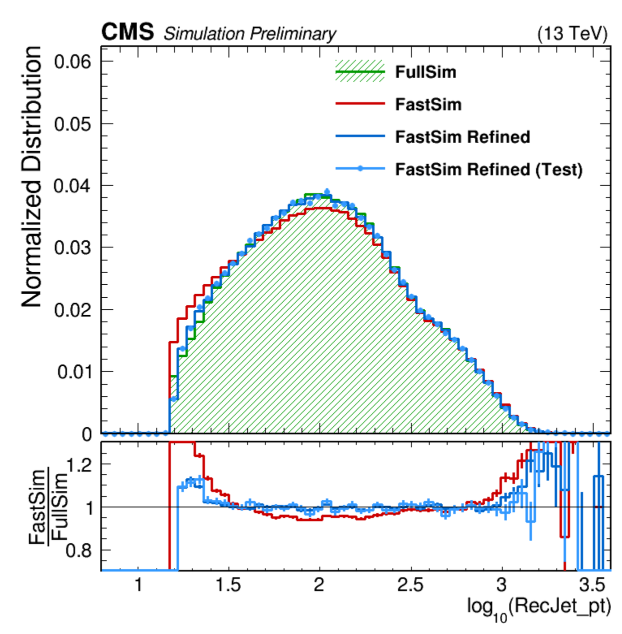
$$\widehat{\text{MMD}}(P,Q) = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}k(x_i,x_j) + \frac{1}{m^2}\sum_{i=1}^{m}\sum_{j=1}^{m}k(y_i,y_j) - \frac{2}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m}k(x_i,y_j)$$

where $n = m =$ batch size $= 2048$ and k : Gaussian kernel (adaptive σ)

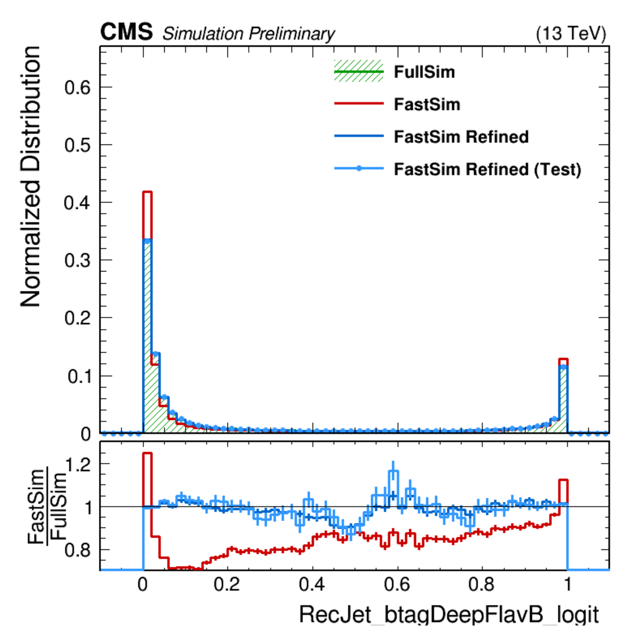
Combine loss terms via MDMM algorithm:

- Modified differential method of multipliers that allows us to account for multiple loss terms such as MSE and loss terms that enforce boundary conditions or unitarity

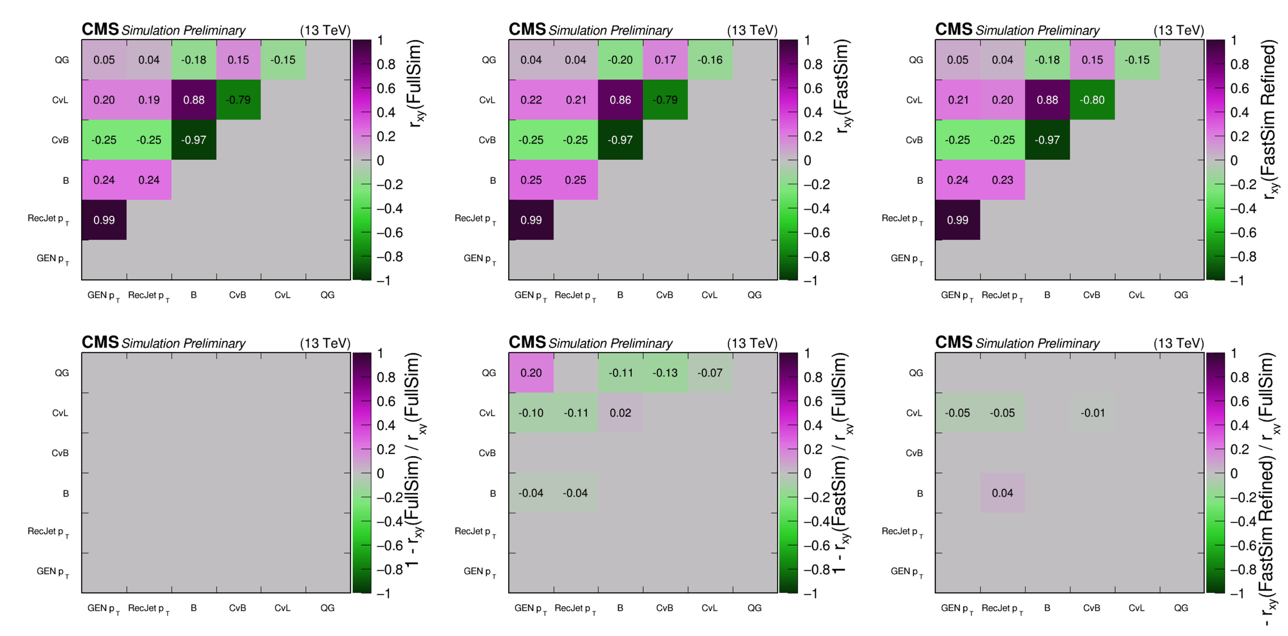
Results



(a) Refinement of jet p_{T}

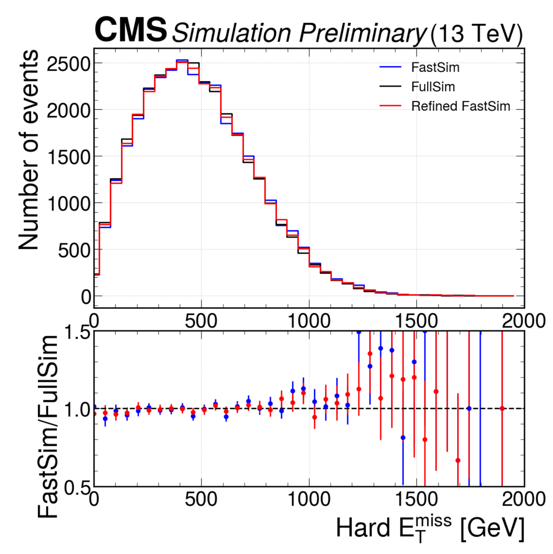


(b) Refinement of b tagged Deep FlavB

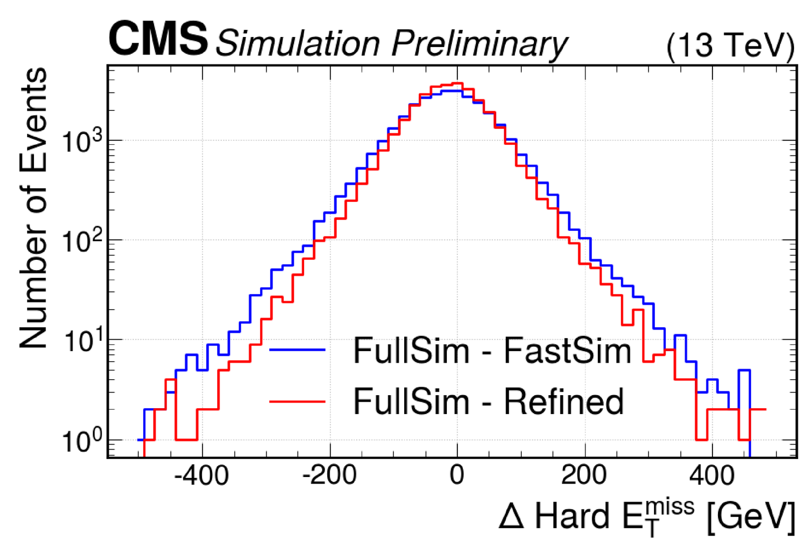


(c) Correlation matrix for FullSim, FastSim, FastSim Refined

Propagate corrected jets to missing transverse energy $E_{\text{T}}^{\text{miss}}$



(d) histograms comparing Fast and Refined to Full



(e) 1D discrepancy Fast-Full and Refined-Full)