

Towards Fast Displaced Vertex Finding

KIM ALBERTSSON^{1,2} FEDERICO MELONI³

¹*CERN*, ²*Luleå University of Technology*, ³*DESY*

ABSTRACT

Many Standard Model extensions predict metastable massive particles that can be detected by looking for displaced decay vertices in the inner detector volume.

Current approaches to search for these events in high-energy particle collisions rely on the presence of additional energetic signatures to make an online selection during data-taking, as the reconstruction of displaced vertices is computationally intensive. Enabling trigger-level reconstruction of displaced vertices could significantly enhance the reach of such searches.

This work is a first step approximating the location of the primary vertex in an idealised detector geometry using a 4-layer dense neural networks for regression of the vertex location yielding a precision of $O(1 \text{ mm})$ [$O(20 \text{ mm})$] RMS in a low [high] track multiplicity environment.

PRESENTED AT

Connecting the Dots and Workshop on Intelligent Trackers (CTD/WIT 2019)
Instituto de Física Corpuscular (IFIC), Valencia, Spain
April 2-5, 2019

1 Introduction

Despite the experimental and theoretical success of the Standard Model of particle physics, gaps still exist. Many beyond Standard Model (BSM) theories addressing these gaps have been proposed, some predicting new particles that can appear at current colliders. However, the particles remain elusive spurring development of specialised detection techniques.

One area of exploration is the search for long-lived particles (LLPs) [1]. These particles have an expected life-time long enough such that their decay occurs with a significant displacement from the location where they are produced. Of particular interest for this work are those particles that decay inside the detector volume, possibly giving rise to *displaced vertices*.

Traditional reconstruction techniques rely on explicit track reconstruction before fitting vertices and are optimised for primary vertex reconstruction and short-lived particles reducing their efficiency for displaced vertices. Such techniques can be adapted to finding displaced decays [2][3], but the resulting algorithm is unsuitable for online application due to high computational costs.

This work explores alternative techniques based on deep learning to reconstruct vertex locations directly from raw detector hits for application at the online trigger level.

To reduce computational overhead, the problem is divided into three steps: (i) Estimate the position of primary interaction; (ii) Define a region of interest based on trigger-level objects (e.g. high-energy muons); (iii) Find displaced vertices in constrained search space.

This work investigates the first step, primary vertex regression from detector hit data. The investigation is limited to feed-forward linear networks because of their small computational cost and thus suitability for integration in an online software trigger. Feed-forward networks are the simplest neural network approach, a method that can utilise hierarchical representation, enabling it to potentially discover indirect correlations in the data.

2 Data sets and generation

A custom event simulation was developed to study the problem using MadGraph5[4] and Pythia[5] for event generation and includes both primary and soft interactions (pile-up). Delphes[6] was used for particle propagation, with one custom module for simulation in a homogeneous magnetic field and one to implement the detector geometry. For this study the exact intersections between particle tracks and active detector surfaces was used. This implies that exact track parameters can be recovered from the data.

The simulation stores simulated particle track parameters and the three coordinates of each detector hit, r, φ, z . The detector geometry consists of eight barrel layers and in total 13 endcap layers, shown in Figure 1, approximating the pixel and SCT subsystems of the ATLAS inner detector. The detector surfaces are modelled as ideal shapes.

For this investigation, three different data sets were generated containing 200'000 events each:

- $e^+e^- \rightarrow Z \rightarrow \mu^+\mu^-$ — Electron-positron collisions produce clean events with small track multiplicities, used to study efficiency and biases.
- $pp \rightarrow Z \rightarrow \mu^+\mu^-$, and $pp \rightarrow t\bar{t}$ — Two proton-proton processes are studied. These collisions are generally busier with larger track multiplicities. Used to study robustness under pile-up. In the generated data, the mean number of tracks per event was 190 (250) for the Z boson (top pair) process.

3 Experimental setup

To establish a baseline model the data were preprocessed using the following two steps:

- For each event, the first N_t truth-level tracks are selected. Tracks are sorted according to their initial direction ϕ_{track} . If fewer than N_t tracks are available in the event the vector is zero-padded at this

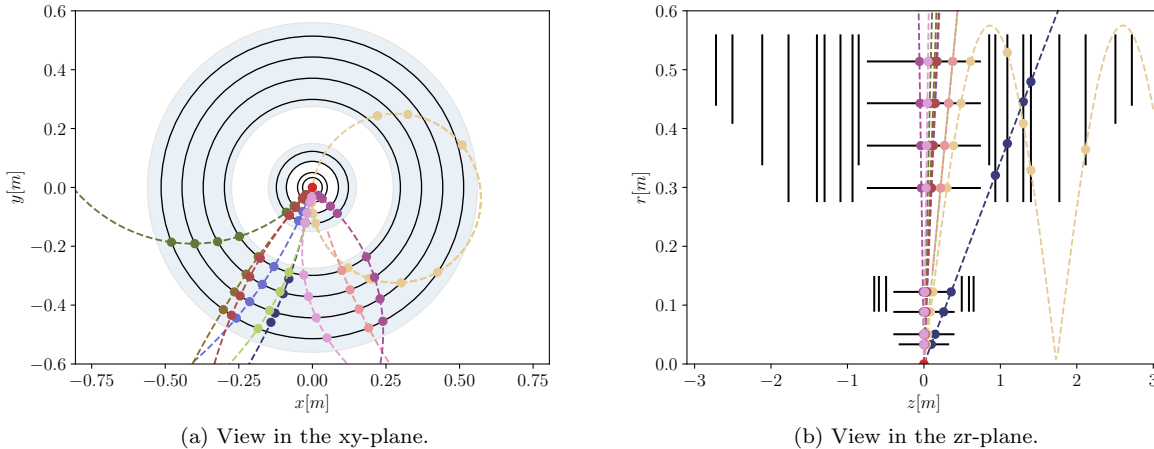


Figure 1: Example proton-proton event viewed with simulated detector. Dots represent detector hits, dotted lines represent the true track trajectory. Thick black lines correspond to detector surfaces. The light blue shaded area corresponds to endcap regions. The primary vertex is shown as a red dot.

stage. N_t was chosen as 2 for the electron data, and 200 for the proton processes, close to the mean number of tracks for a typical event.

- For each selected track, a fixed number of hits, N_h , were selected and sorted *in order of track propagation*. In experiments, N_h was selected to be large enough to cover a typical track.

The number of features per input hit, N_f , is 3.

The neural network hyper-parameter optimisation was performed using the $e^+e^- \rightarrow Z \rightarrow \mu^+\mu^-$ data. A grid search was used to perform the optimisation with a total of 1296 evaluated points using a regular logarithmic spacing for between 8 and 1024 hidden units per layer. The process and pile-up dependency was investigated using all three data sets. For both setups, the data were split into training data and validation data using half the data for each split. Early stopping was used for regularisation. Pile-up, when included, followed a Poisson distribution with $\mu = 1$.

The configurations are summarised in Table 1 where Π gives the total network input size and is calculated as $\Pi = N_t N_h N_f$.

	Input size				Layer size				Output
	N_t	N_h	N_f	Π	1	2	3	4	size
(i) Hyper-param.	2	14	3	84	32-1024	32-1024	8-256	8-256	1
(ii) Process dep.	200	10	3	6000	512	512	256	256	1

Table 1: Network configuration for the two experiment setups.

4 Results

The results for the hyper-parameter optimisation on the $e^+e^- \rightarrow Z \rightarrow \mu^+\mu^-$ data is shown in Figure 2. The best performing point yielded an RMS of 0.92 mm in z , the average was 1.2 mm and the worst performing model had an RMS of 3.1 mm.

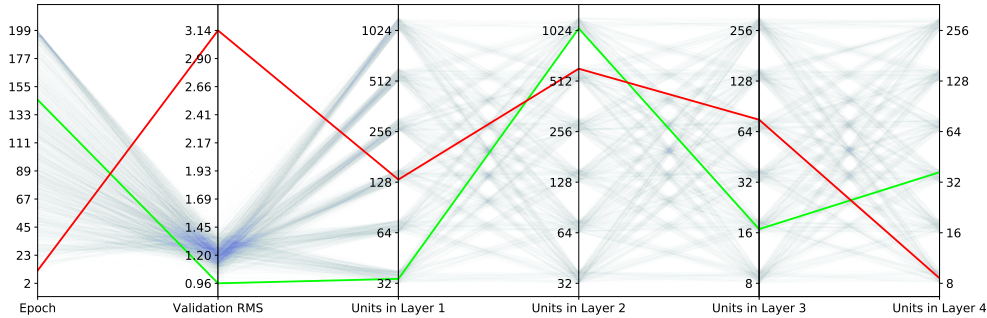


Figure 2: Parallel axis plot of different configurations of hyper-parameters trained on the $e^+e^- \rightarrow Z \rightarrow \mu^+\mu^-$ data set. The first axis shows the number of training epochs. The second axis shows the performance of the configuration as the average reconstruction error across all studied events. The last four axes show the number units in each of the layers. The green line represents the best performing model and the red line represents the worst performing model. Blue lines are less transparent with better performance.

	$e^+e^- \rightarrow Z \rightarrow \mu^+\mu^-$		$pp \rightarrow Z \rightarrow \mu^+\mu^-$		$pp \rightarrow t\bar{t}$	
	bias	RMS	bias	RMS	bias	RMS
No pile-up	-0.10 mm	0.98 mm	0.92 mm	20 mm	1.5 mm	16 mm
Pile-up ($\mu = 1$)	—	—	4.0 mm	28 mm	3.1 mm	22 mm

Table 2: Average of the squared error as bias, and the root mean square measure of the error as RMS when regressing the primary vertex z location from raw detector hits.

Results for the different physics processes considered are shown in Table 2. The reconstruction performs best at low track multiplicities. For larger multiplicities the performance is degraded, with a larger degradation for the Z boson sample as compared to the top pair one.

The time taken for evaluating a single event on a CPU was on the order of 0.10 ms, sufficiently fast for the inclusion in a software trigger, which often operate at time scales of several hundred ms.

The data preprocessing is highly idealised. The network is fed fixed length tracks with hits in track order, furthermore the tracks are sorted. Additionally, the hits use the exact detector surface crossing, meaning track parameters are exactly recoverable. This point coupled with the fact that different assignments of hyper parameters perform similarly suggests a lack of model capacity.

5 Conclusion

An initial study of the regression of the z -coordinate of the primary vertex from detector hits was investigated using a 4-layer feed-forward linear network. The results show that using this setup a precision of $O(1 \text{ mm})$ RMS can be reached in an idealised low track multiplicity setting. The performance is degraded to $O(20 \text{ mm})$ RMS for processes with a track multiplicity on the order of 200.

The results highlight a limited modelling capacity in the currently considered network architecture and future work will focus on finding a model better suited for the problem setup.

References

- [1] Lee, L. et al., "Collider Searches for Long-Lived Particles Beyond the Standard Model", Prog. Part. Nucl. Phys. vol. 106 p. 210-255 (2019).

- [2] ATLAS Collaboration, "Performance of the reconstruction of large impact parameter tracks in the ATLAS inner detector", ATL-PHYS-PUB-2017-014 (2017), <https://cds.cern.ch/record/2275635>.
- [3] CMS Collaboration, "Description and performance of track and primary-vertex reconstruction with the CMS tracker", JINST vol. 9 (2014), <https://cds.cern.ch/record/2275635>.
- [4] Alwall J. et al., "The automated computation of tree-level and next-to-leading order differential cross sections, and their matching to parton shower simulations," JHEP 1407 (2014) 079.
- [5] Sjöstrand, T. et al., "An Introduction to PYTHIA 8.2," Comput.Phys.Commun. 191 (2015) 159-177.
- [6] DELPHES 3, de Favereau, J., Delaere, C. et al., "DELPHES 3: a modular framework for fast simulation of a generic collider experiment," J. High Energ. Phys. (2014) 2014: 57.