Application of neural networks in rapid estimation of the impact parameter of high-energy collisions from data obtained from microchannel plates detector

Authors: Galaktionov K.A., Roudnev V.A., Valiev F.F., Feofilov G.A.
Speaker: Galaktionov K.A.

Saint Petersburg State University

The use of new methods for processing data of a physical experiment. Application of machine learning methods on the NICA complex
https://indico.cern.ch/event/1306558/
28.08.2023 - 29.08.2023

Microchannel plate detectors[1]

Some features of these detectors:

- Variability in size
- Registration of charged particle hits, one per detector cell
- Time of flight resolution $\approx$ 50 - 200 ps

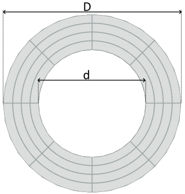


Fig. 1 Scheme of the configuration

Configuration № 1
(Big rings) $3.5 < \eta < 5.8$

Configuration № 2
(Small rings) $4.4 < \eta < 5.8$

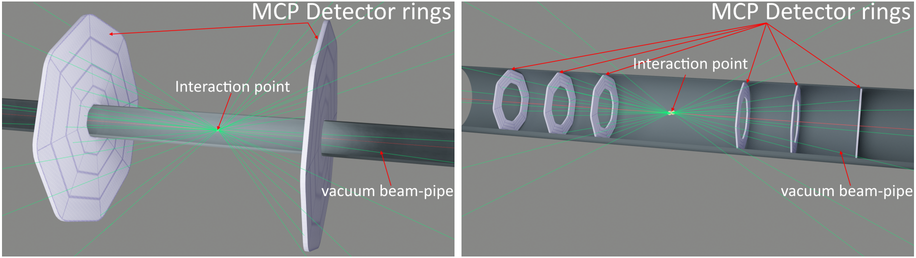


Fig. 2 Scheme of modeled detector configurations (not to scale). (left) - outside the beam-pipe in thin-wall vacuum chambers, one pair of big rings ($d = 5$ cm, $D = 50$ cm), (right) - inside vacuum beam-pipe, three pairs of small rings ($d = 3$ cm, $D = 5$ cm).

[1] A.A. Baldin et al. “Fast beam–beam collisions monitor for experiments at NICA”. In: (). DOI: https://doi.org/10.1016/j.nima.2019.04.108.

Datasets

We used two datasets:

1. QGSM dataset: 200 000 modeled Au+Au collisions $\sqrt{s_{NN}} = 11$ GeV.
2. EPOS dataset: 360 000 modeled Au+Au collisions $\sqrt{s_{NN}} = 11.5$ GeV.

Both dataset has bdb weighted distribution of impact parameter (i.e., the number of events with an impact parameter b being proportional to b)

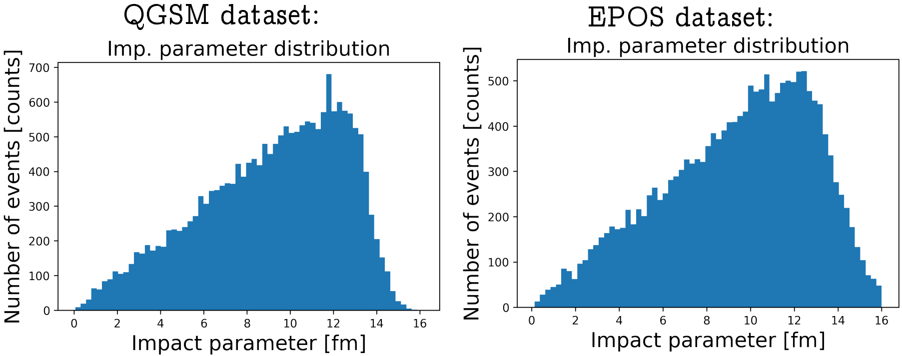


Fig. 3 Impact parameter distribution. (left) - QGSM dataset, (right) - EPOS dataset.

Event features

Statistical approach

Using statistical event features such as number of detected hits (N_{ch}) and mean polar angle of hits (Θ_{ch}).

Time-of-flight approach

Using information about every particle hit, including time of flight of particle and detector cell.

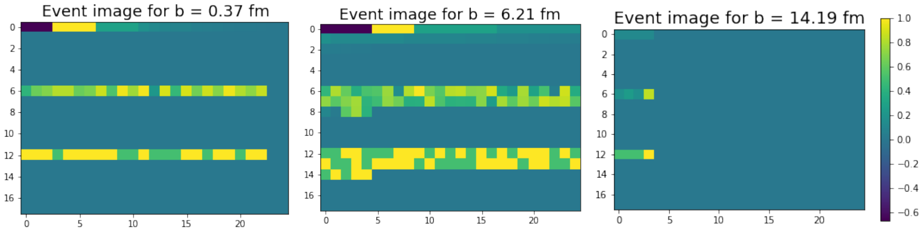


Fig. 4 Examples of event images (event features) for time-of-flight approach.

Artificial neural networks (ANNs)

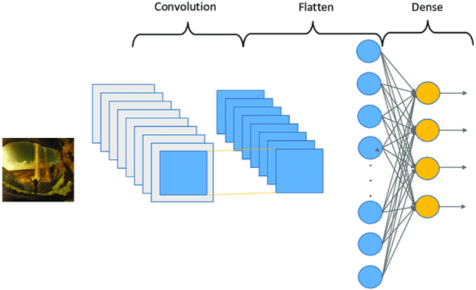


Fig. 5 Dense layers (Eq. 1) and convolution layers (Eq. 2) schemes.

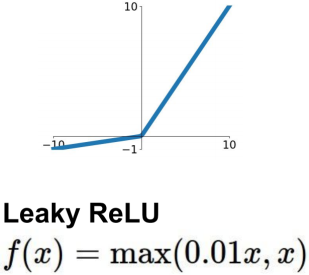


Fig. 6 Leaky RELU activation function

ANN - an example of supervised learning.
Formula describing a dense layer of a neural network.

$$y = \theta(x * A^T + b) \tag{1}$$

Formula describing a convolutional layer of a neural network.

$$out(N_i, C_{out_j}) = \theta(bias(C_{out_j}) + \sum_{k=0}^{C_{in}-1} weight(C_{out_j}, k) \star input(N_i, k)) \tag{2}$$

Where: y, out - outputs of layer; $x, input$ - inputs of layer; A^T - transpose of a matrix of weights; $weight$ - convolution kernel; b, bias - biases of layer, $\theta(x)$ - activation function.

Artificial neural networks (ANNs)

We have used two types of loss functions. Mean squared error (MSE):

$$\sqrt{MSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{b}_i - b_i)^2} \tag{3}$$

N – the size of the training set, b_i – the impact parameter of the i-th event, $\hat{b}_i$ – the estimation of the impact parameter of the i-th event, and the summation goes over the entire set.

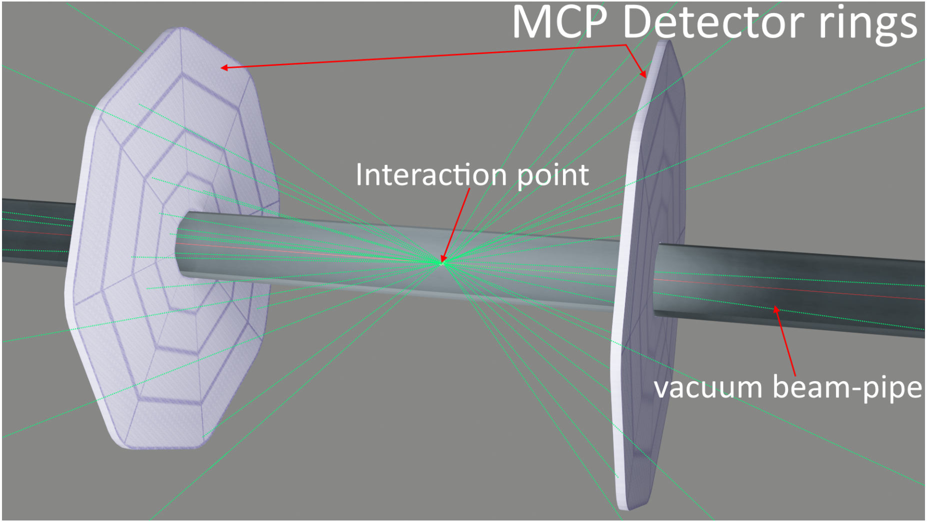
And binary cross-entropy (for classification problem):

$$CE = \frac{1}{N}\sum_{i=1}^{N}[-y_i \log(p_i) + (1 - y_i)\log(1 - p_i)] \tag{4}$$

N – the size of the training set, y_i – the real probability that the impact parameter of the i-th event is below the threshold (as we know impact parameter value, this probability can be 1 or 0), p_i – the estimated probability that the impact parameter of the i-th event is below the threshold, and the summation goes over the entire set.

For training process "ADAM" optimizer was used.

Configuration № 1 - Big rings



Configuration № 1 (Big rings). Statistical approach. Event features.

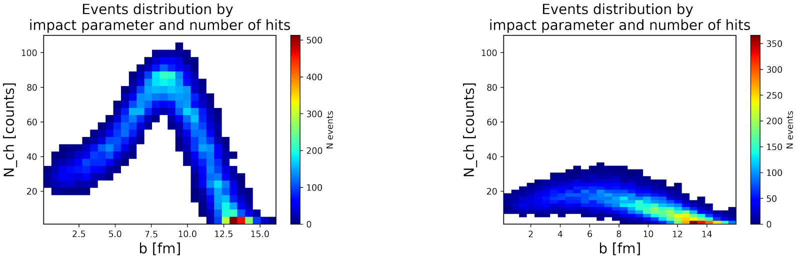


Fig. 7 Events distribution by number of registered particle hits. (left) - QGSM dataset, (right) - EPOS dataset.

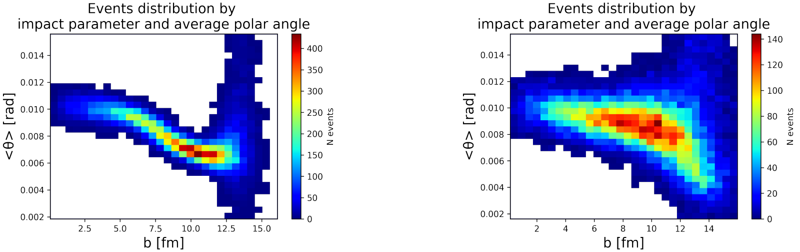


Fig. 8 Events distribution by mean polar angle of registered hits. (left) - QGSM dataset, (right) - EPOS dataset.

Configuration № 1 (Big rings). Statistical approach. Regression results.

The goal of the neural network was to estimate the value of the impact parameter of the event. Loss function - MSE on training set. Accuracy metrics: MSE, MAE.

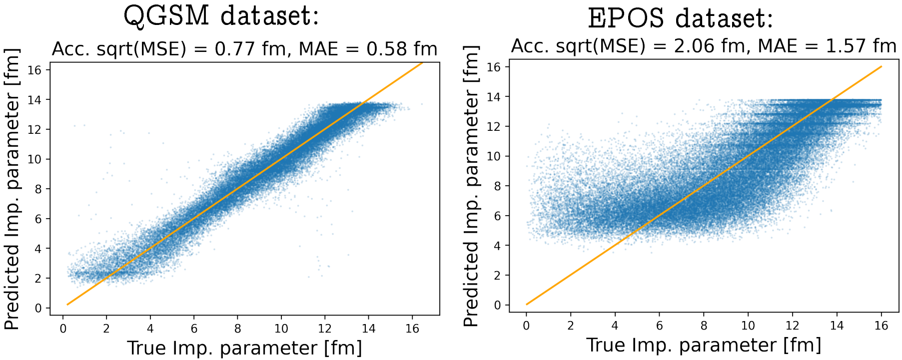


Fig. 9 Dependence of the evaluated impact parameter on the true value. Scatter plot, where each dot represents one event from test set. (left) - QGSM dataset, (right) - EPOS dataset.

Configuration № 1 (Big rings). Statistical approach. Classification results.

Class 1 - impact parameter below the threshold of 5 fm, Class 2 - above 5 fm. Loss function - cross entropy. Accuracy metrics: Accuracy - percentage of correctly identified events.

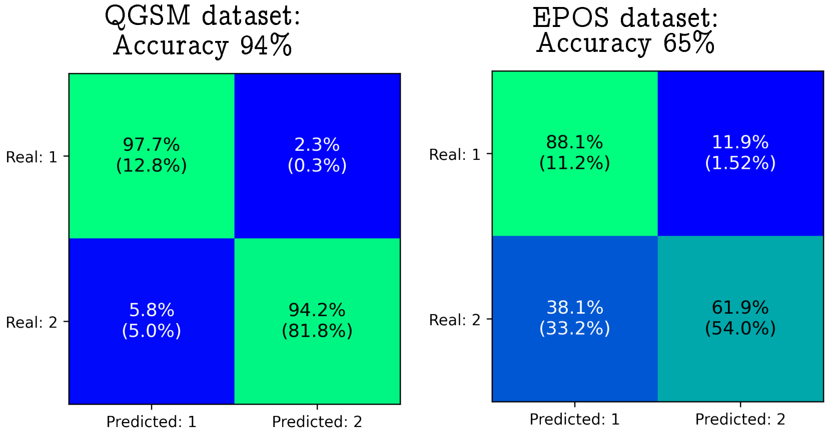
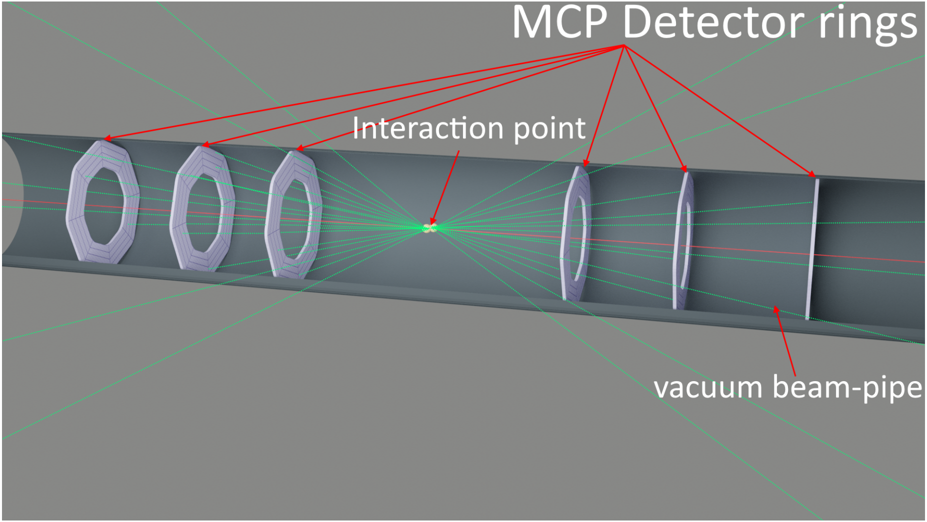


Fig. 10 Confusion matrices. Value in brackets - normalized to the number of events in test set, value outside of brackets - normalized to the number of events in real class.

Configuration № 2 - Small rings



Configuration № 2 (Small rings). Statistical approach. Event features.

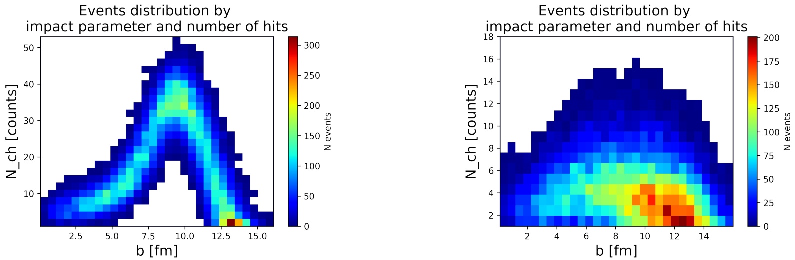


Fig. 11 Events distribution by number of registered particle hits. (left) - QGSM dataset, (right) - EPOS dataset.

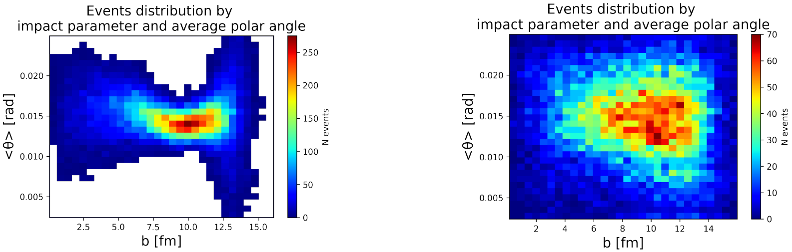


Fig. 12 Events distribution by mean polar angle of registered hits. (left) - QGSM dataset, (right) - EPOS dataset.

Value of particle type information.

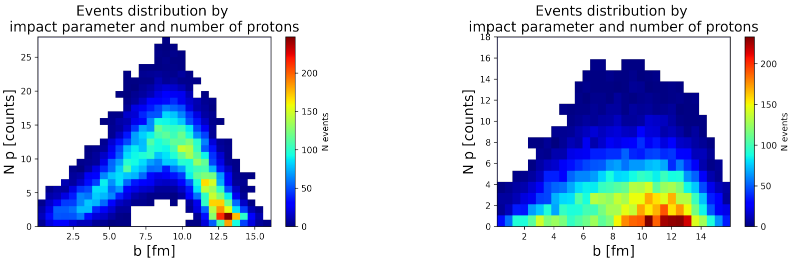


Fig. 13 Events distribution by number of registered protons. (left) - QGSM dataset, (right) - EPOS dataset.

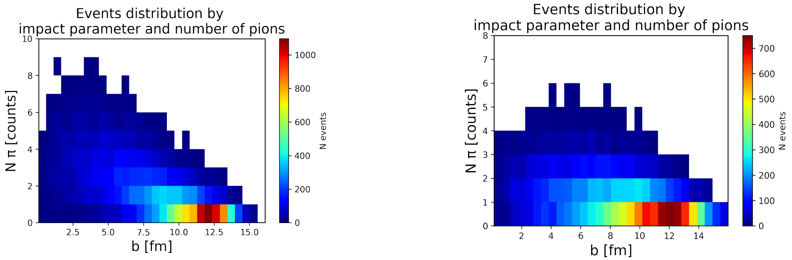


Fig. 14 Events distribution by number of registered pions. (left) - QGSM dataset, (right) - EPOS dataset.

Time of flight usage.

There is a difference between the most common time-of-flights of these two types of particles. Based on this difference, one can extract a hit feature:

$$\nu = \frac{1}{t - t_{0i}}$$

where t - time-of-flight of particle, t_{0i} - average time of flight of pions on i-th detector.

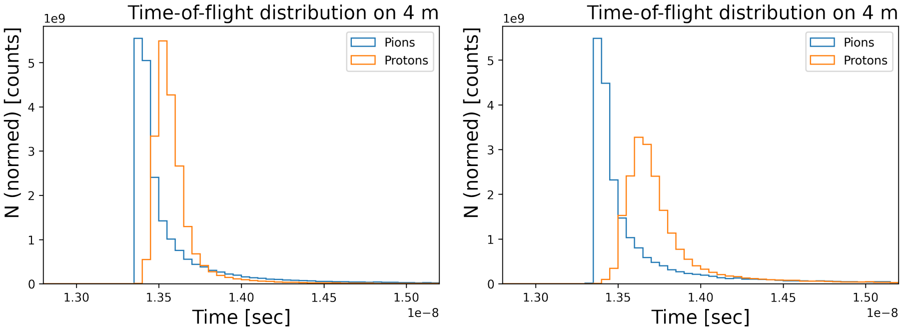


Fig. 15 Pions and protons time-of-flight distribution (at 4 m distance). (left) - QGSM dataset, (right) - EPOS dataset.

Configuration № 2 (Small rings). Time-of-flight approach. Regression results.

Loss function - MSE on training set. Accuracy metrics: MSE, MAE.

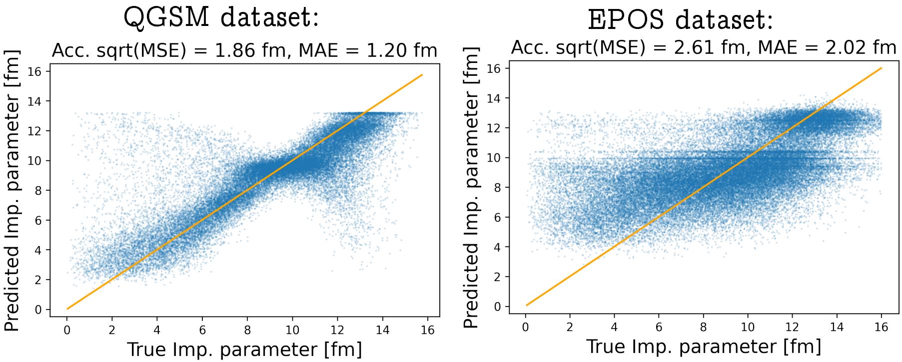


Fig. 16 Dependence of the evaluated impact parameter on the true value. Scatter plot, where each dot represents one event from test set. (left) - QGSM dataset, (right) - EPOS dataset.

Configuration № 2 (Small rings). Time-of-flight approach. Classification results.

Class 1 - impact parameter below the threshold of 5 fm, Class 2 - above 5 fm. Loss function - cross entropy. Accuracy metrics: Accuracy - percentage of correctly identified events.

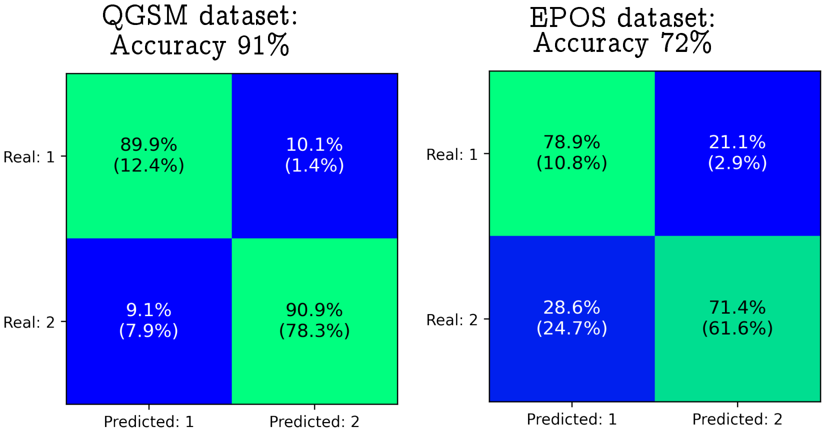
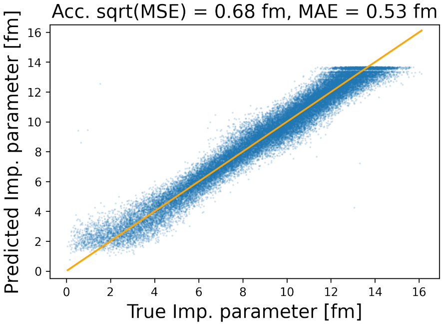


Fig. 17 Confusion matrices. Value in brackets - normalized to the number of events in test set, value outside of brackets - normalized to the number of events in real class.

Configuration № 1 (Big rings). Time-of-flight approach. Regression results.

Loss function - MSE on training set. Accuracy metrics: MSE, MAE.

QGSM dataset:



EPOS dataset:

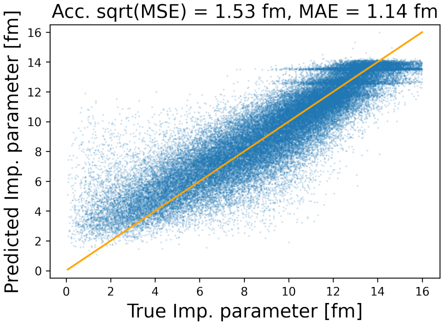
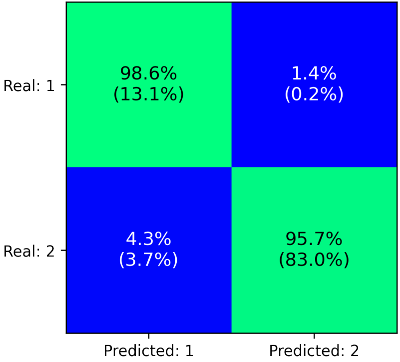


Fig. 18 Dependence of the evaluated impact parameter on the true value. Scatter plot, where each dot represents one event from test set. (left) - QGSM dataset, (right) - EPOS dataset.

Configuration № 1 (Big rings). Time-of-flight approach. Classification results.

Class 1 - impact parameter below the threshold of 5 fm, Class 2 - above 5 fm. Loss function - cross entropy. Accuracy metrics: Accuracy - percentage of correctly identified events.

QGSM dataset:
Accuracy 96%



EPOS dataset:
Accuracy 84%

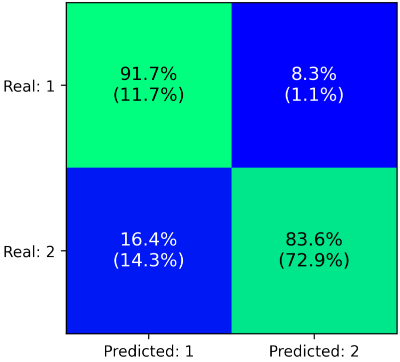


Fig. 19 Confusion matrices. Value in brackets - normalized to the number of events in test set, value outside of brackets - normalized to the number of events in real class.

Cross-dataset validation. Configuration № 1 (Big rings). Time-of-flight approach. Regression results.

The idea of cross-dataset validation is to train neural network on one dataset and test it's performance on the other dataset. Loss function - MSE on training set. Accuracy metrics: MSE, MAE.

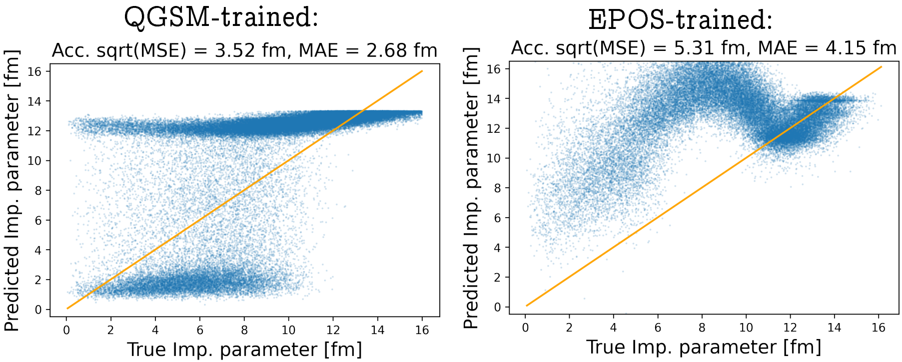
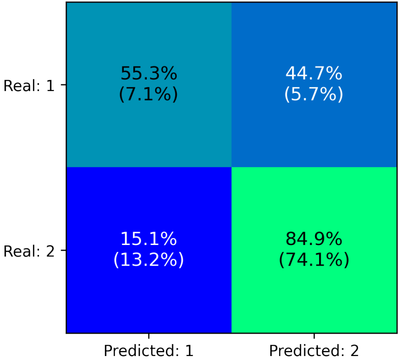


Fig. 20 Dependence of the evaluated impact parameter on the true value. Scatter plot, where each dot represents one event from test set. (left) - trained on QGSM, test on EPOS dataset, (right) - trained on EPOS, test on QGSM dataset.

Cross-dataset validation. Configuration № 1 (Big rings). Time-of-flight approach. Classification results.

Class 1 - impact parameter below the threshold of 5 fm, Class 2 - above 5 fm. Loss function - cross entropy. Accuracy metrics: Accuracy - percentage of correctly identified events.

QGSM-trained:
Accuracy 96%



EPOS-trained:
Accuracy 84%

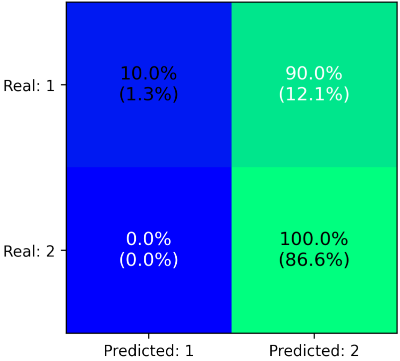


Fig. 21 Confusion matrices. (left) - trained on QGSM, test on EPOS dataset, (right) - trained on EPOS, test on QGSM dataset.

1 fm classification. Configuration № 1 (Big rings). Time-of-flight approach.

Class 1 - impact parameter below the threshold of 1 fm, Class 2 - above 1 fm. Loss function - cross entropy. Accuracy metrics: Accuracy - percentage of correctly identified events.

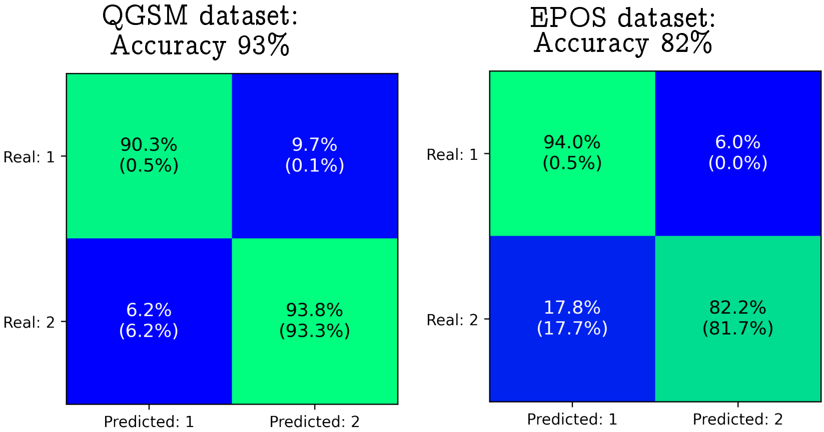


Fig. 22 Confusion matrices. (left) - trained on QGSM, test on EPOS dataset, (right) - trained on EPOS, test on QGSM dataset.

Overall comparison table

Event features	Threshold [fm]	QGSM		EPOS	
		TP [%]	FP [%]	TP [%]	FP [%]
Configuration №2 "Small rings"					
Time-of-flight	5	89.9 (12.4)	9.1 (7.9)	78.9 (10.8)	28.6 (24.7)
Time-of-flight	1	89.4 (0.5)	12.1 (12.1)	73.1 (0.3)	22.9 (22.7)
Configuration №1 "Big rings"					
Time-of-flight	5	98.6 (13.1)	4.3 (3.7)	91.7 (11.7)	16.4 (14.3)
Time-of-flight	1	90.3 (0.5)	6.2 (6.2)	94.0 (0.5)	17.8 (17.7)
Statistical	5	97.7 (12.8)	5.8 (5.0)	88.1 (11.2)	38.1 (33.2)
Statistical	1	98.9 (0.5)	8.8 (8.8)	77.2 (0.4)	21.1 (21.0)

TP - percentage of true positive predictions, FP - percentage of false positive predictions. Value in brackets - normalized to the number of events in test set, value outside of brackets - normalized to the number of events in real class.

Conclusions

+ The developed technology makes it possible to evaluate the impact parameter of single event and to highlight the events of head-on collisions
+ The use of the time-of-flight improves the quality of impact parameter estimation
+ An issue with the validity of the datasets has been faced. However, neural network approach turned out to be useful in any case.
+ With certain geometric characteristics and time-of-flight resolution, the problem is solved with both sets of data, which makes it possible to evaluate the requirements for detector equipment

Future research plans

- Validation of existing results on new synthetic datasets
- Building a universal algorithm and searching for event characteristics that are invariant with respect to the data generator
- Fine tuning models for future possible applications

Supported by Saint Petersburg State University, project ID: 94031112

Backup slides

Computational resources

Evaluation was performed by calculating the amount of floating point multiplications needed for the work of algorithm.

Big detectors geometry, statistical approach:

- 300 - 400 floating point multiplications
- Preprocessing: number of hits and mean angle
- 2 x 352 cells

Small detectors geometry, time of flight approach:

- 10000 - 80000 floating point multiplications
- Preprocessing: time-of-flight evaluation
- 6 x 32 cells

All values are approximate and require fine tuning.

Study on the impact of temporal resolution. Configuration № 1 (Big rings). QGSM dataset.

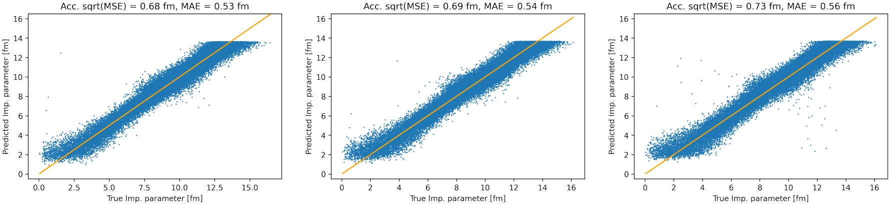


Fig. 23 Dependence of the evaluated impact parameter on the true value. (left) - 50 ps discrete values, (center) - 200 ps discrete values, (right) $t \sim \mathcal{N}(t_0, (200\ \text{ps})^2)$, where t_0 - accurate time-of-fligh.

Study on the impact of temporal resolution. Configuration № 1 (Big rings). EPOS dataset.

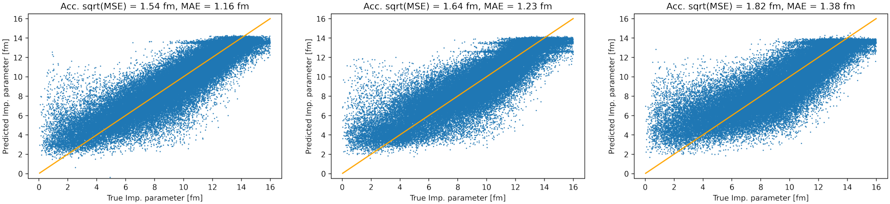


Fig. 24 Dependence of the evaluated impact parameter on the true value. (left) - 50 ps discrete values, (center) - 200 ps discrete values, (right) $t \sim \mathcal{N}(t_0, (200\ \mathrm{ps})^2)$, where t_0 - accurate time-of-fligh.