Software for MUonE

Marcin Kucharczyk

IFJ PAN Kraków

Workshop on Muon Precision Physics
Liverpool, 7-9 November 2022

Outline

FairRoot based software (FairMUonE)

- generation
- simulation
- digitization
- reconstruction

READY FOR FULL SIMULATION/RECONSTRUCTION (test runs & final detector)

Alignment

- global alignment
- final detector *z* alignment based on particle energy loss

Machine learning techniques for MUonE online

- fast pattern recognition for trigger
- real time event reconstruction

Computing resources

FairMUonE - generation

FairMUonE software package based on FairRoot framework

Generation

- LO signal generator
- NLO MESMER generator already implemented

  → interface (Fortran/C++)

  → input: beam parameters → output: μ-e scattering vertex

  → event-by-event as for LO
- Accurate beam profile implemented

  → simple profiles with fixed energy also available
- Job configuration allows to choose what to be generated

  → minBias or μ-e signal
- Possibility to store separate container of signal particles

  → container with gereration parameters (e.g. weights) will be added

FairMUonE - simulation

Simulation

- New version of Geant4 v10.7.1 implemented into FairRoot

  $\rightarrow$ containing updated settings relevant for MUonE

  $\rightarrow$ recommended as reproducing correctly the angular distributions of e^+e^- pair production from muon interactions in the material

- Full test-run detector geometry implemented

  $\rightarrow$ separate *Si* sensors and calorimeter crystals

  $\rightarrow$ module tilt

  $\rightarrow$ works with any number of stations, modules, calorimeter crystals etc.

  $\rightarrow$ common geometry *(.yaml files)* for simulation/digitization/reconstruction

- Bethe-Heitler 5D Model of γ-ray conversions to e^+e^-

  $\rightarrow$ more realistic description of pair production in Geant4

  $\rightarrow$ being implemented to Geant4

FairMUonE - digitization

Digitization for tracking stations implemented

- same detector model as in simulation
- full tracker digitization
- realistic electronic noise and channel cross-talk
- stubs finally produced

Produce MC in Raw Data format

→ DAQ would distribute C/C++ API (headers and a library) that will decode/encode data into simple bitfield structure

→ coding/encoding planned to be implemented it into the simulation interfacing digitizer to the encoding API

Calo digitization

- ’basic’ version of full calorimeter digitization implemented
- more realistic model ongoing

FairMUonE - reconstruction

Track reconstruction

- effective pattern recognition using all the hits *(x, y, stereo)*
- efficient close track reconstruction + clone killing
- track linking to MC
- Linear track fitting with iterative procedure and outlier removal
- Kalman filter

  $\rightarrow$ output is usually not a straight line

  $\rightarrow$ forward and backward iterations $\rightarrow$ best estimated first state

Vertex reconstruction

- Kinematic fit forcing tracks to go through the same point, *z* position fixed to the middle of the target; find (x_v, y_v) at z_{target} which minimize sum of χ^2_{vtx}

  $\rightarrow$ calculate new track parameters *(slopes)* & angles of outgoing ’electron’ and ’muon’

  $\rightarrow$ for mu-e elastic scattering signal
- Adaptive vertex fitter

  $\rightarrow$ adaptive weighted least quare method

  $\rightarrow$ Tukey biweight method used to assign weight to a track according to its χ^2(IP)

  $\rightarrow$ useful for background studies & longitudinal alignment

Job configuration

Ability to configure output ntuple contents using job configuration,

e.g. *gen+sim+digi+reco* *digi+reco* *gen+sim+reco*

Finally: single ntuple

- generation tree
  - → with additional container of signal particles
  - → container with gereration parameters to be added
- simulation tree
  - → MC tracks
  - → Geant "hits" in the tracker
  - → Geant "hits" in the calorimeter
- digitization tree
  - → strip digits
  - → stubs
- reconstruction tree
  - → reconstructed 3D tracks *(linear fit & Kalman filter saved)*
  - → reconstructed vertices *(least square fit & adaptive vertex fitter saved)*
  - → best χ^2 vertex

Reconstruction of test-beam 2018 data

Published in JINST 16 (2021) P06005

- Setup located downstream COMPASS
- Aim of the measurement campaign

  → muon-electron elastic scattering with high statistics
- Using muons from pions decays *(hadron beam)*

  → estimated beam momentum p_{beam} = (187±7) GeV
- Measure correlation between the scattering angles

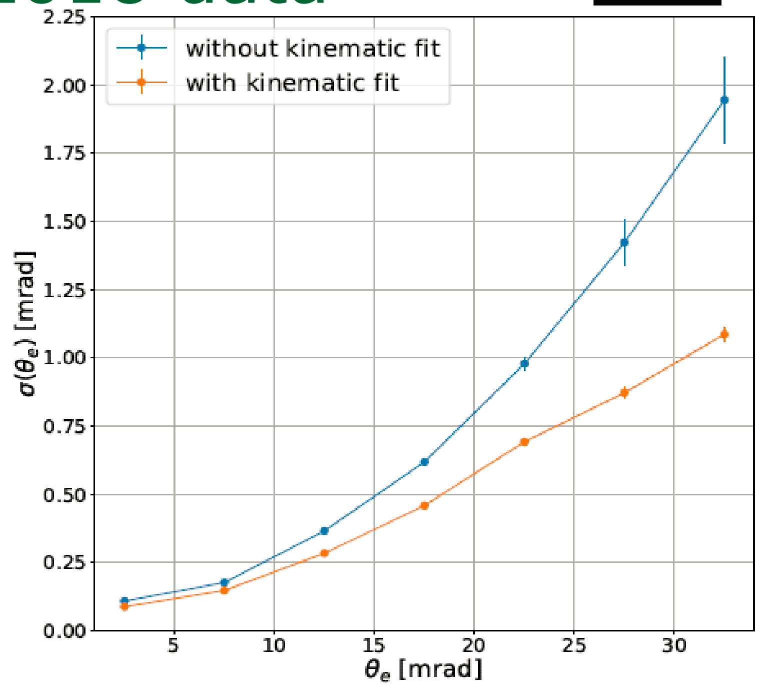
  → muon angle vs the electron angle
- Electron energy vs electron angle correlation and PID
- Detector

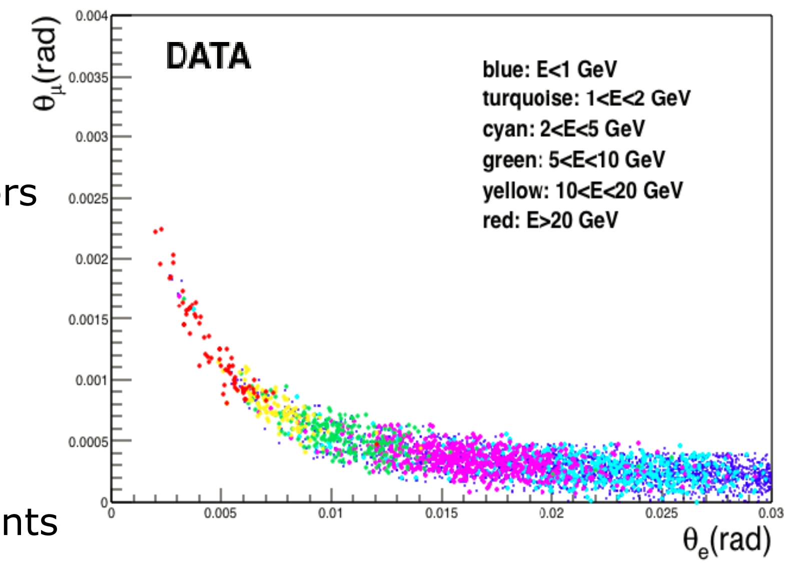
  → tracking system:

  16 stations equipped with AGILE silicon strip sensors

  400 micron thick, single sided, about 40 micron intrinsic hit resolution

  → electromagnetic calorimeter: 3x3 cell matrix, BGO-PMT crystals, ~8×8 cm^2





Conclusion: able to select a clean sample of elastic events

Tracking efficiency studies

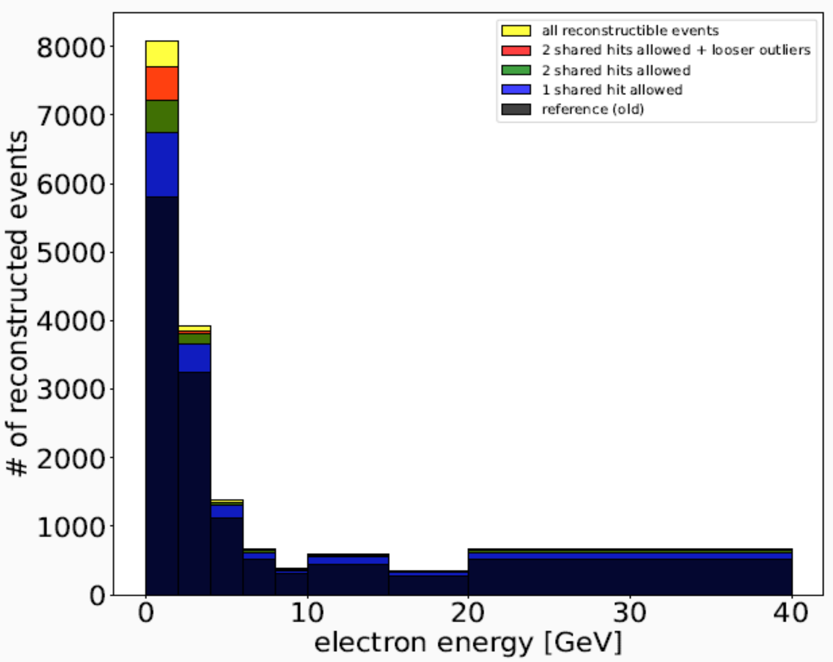
Deal with problems related to **close tracks** / multiple scattering for electrons

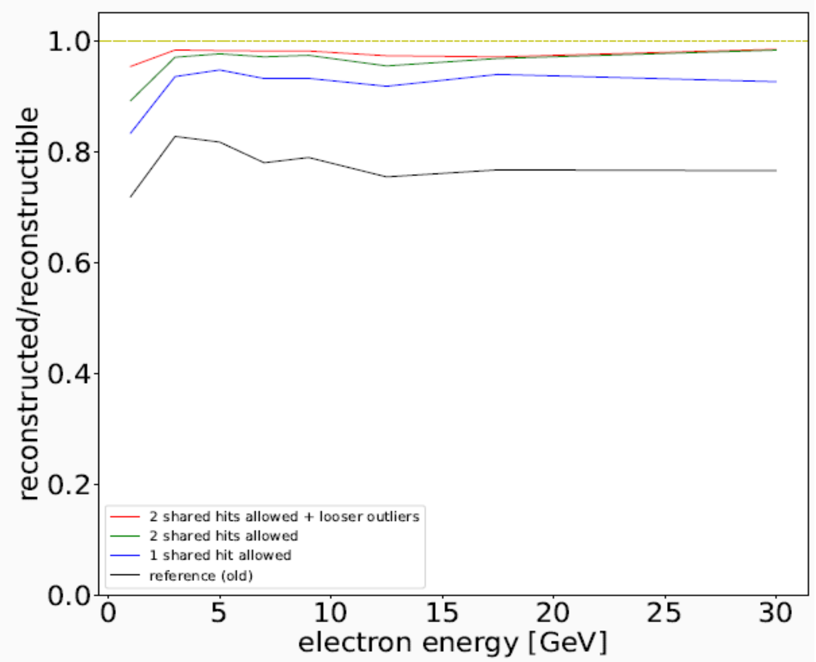
Possible solutions:

- loose the cuts in clone removal algorithm, i.e. allowing hits to be shared between tracks
- loosening threshold for outlier hits

This gives a possibility to achieve almost 100% tracking efficiency for loose cuts (depending on electron energy)

Possibility to optimally choose the working point according to the signal efficiency vs background rejection





Global alignment

- Global alignment *(align x,y,z positions in a single step)*

  → alignment parameters α determined by minimizing the global χ^2

$$\chi^2_{\mathrm{global}} = \sum_i \chi^2_i, \qquad \frac{\mathrm{d}\chi^2}{\mathrm{d}\boldsymbol{\alpha}} = 0, \qquad \boldsymbol{\rho} \equiv \boldsymbol{\rho}(\boldsymbol{\pi}(\boldsymbol{\alpha}), \boldsymbol{\alpha})$$

  → global χ^2 is simply the sum of the χ^2 values for all tracks

  → but now the residuals ρ depend on the alignment parameters α as well as track fit parameters π

In general: minimize global χ^2 over all the parameters *(e.g. track and vtx parameters, bend etc.)*

  → first determine the total derivatives of χ^2 with respect to the alignment parameters

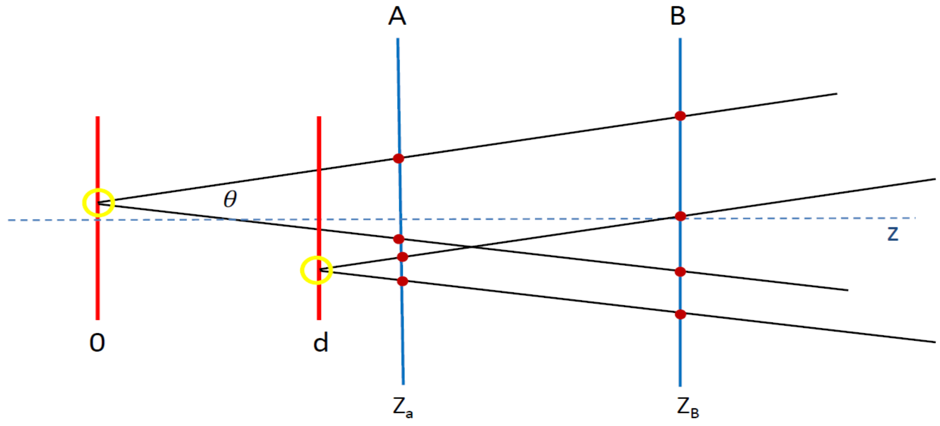
  → solving the system of equations with first and second derivatives of the global χ^2 with respect to α gives the covariance matrix of the residuals of the track fit

MUonE is perfectly suited for the Global χ^2 approach. The problem is manifestly linear, and the convergence should be reached in a single go.

Longitudinal alignment

- Longitudinal software alignment
  → $\Delta z \sim 10\mu m$ ultimate precision requires the software alignment
  → local survey *(e.g. laser technique)* could give a precision $\Delta z \sim$ 50-100μm
  → solution: very thin target *(e.g. 10 μm)* & reconstruct the vertex from this target

- How to get the absolute scale (first station)?
  → precisely positioned two thin foil targets



→ use adaptive vertex fitter
→ use particle gun with pions to enhance the multiplicity of tracks from vertex
→ use global alignment: to align in *(x,y)* and to find the *z* positions/resolutions

Use scattered muon energy loss in *Si* sensors / target to provide absolute *z* to consecutive stations

Real-time event reconstruction in MUonE

Deep Machine Learning (DNN) techniques for the final detector trigger

- Two-level trigger for the final detector

1. Fast FPGA cards based flexible trigger system employing tracks / track elements (stubs) and allowing to read-out full detector at the rate of 40 MHz

   → use deep machine learning based pattern recognition for track reconstruction

   - improve reconstruction precision and speed up processing

   → events accepted on the FPGA-based stage will be then passed to the second level, based on the **full online event reconstruction** *(if rate still too high)*

2. Offline quality tracking in the real time reconstruction to select signal at highest possible efficiency ONLINE

   → full track and vertex reconstruction in the real time

   → parallel processing architectures

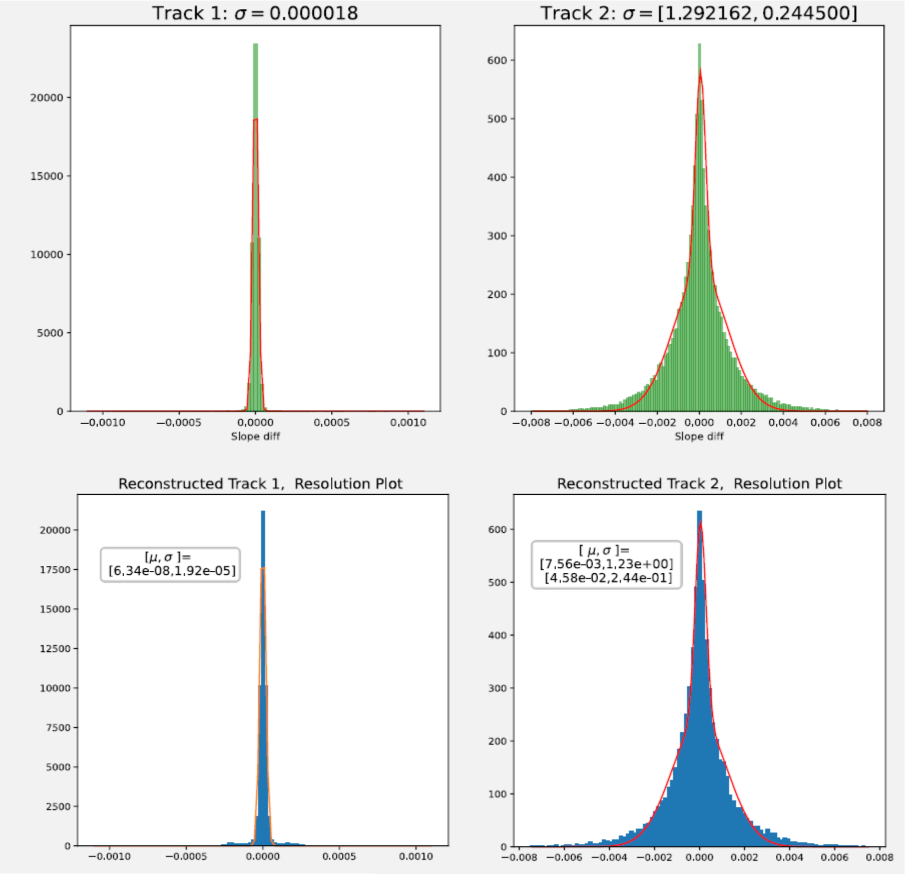
   → two candidates are: a) new multicore (CPU) XF86 processors with AVX256
   b) servers with GPUs

   → Deep Machine Learning used for pattern recognition and track fitting

Deep Machine Learning for MUonE online

Computer Science 20(4) (2019) 477-493
DCAI 2021 Lecture Notes, vol. 2, p 202-205

- Idea: SPEED UP PATTERN RECOGNITION STEP
  - → output is 3D regardless of 2D (x-z, y-z) inputs
- Input: all hits concatenated, no distinction between X, Y and stereo layers
- Ground truth: MC track slope parameters
- Model
  - → PyTorch
  - → 6 linear layers
  - → up to 2000 neurons per layer
- Loss function
  - → MSELoss from PyTorch – uses difference between predicted slope parameters and ground truth

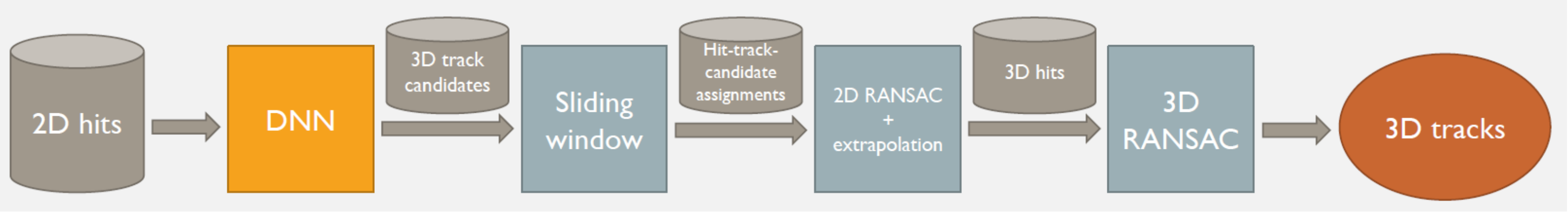


- Track 1: muon, Track 2: electron
- Upper: DNN based algorithm
- Lower: „conventional" reconstruction

3D tracking with DNN

‚Direct' algorithm used

- Reconstructed tracks used to assign hits to particles - pattern recognition
- Linear fits using RANSAC (Random sample consensus) method to fit the tracks



VERY PROMISING PRELIMINARY RESULTS

Direct method proved to be successful

- Hit assignment: avg. 11.75 in 12-hit event (98%)
- **Events reconstructed: 94%**

Plans

- Validation of the results
- Optimizing training process
- Optimizing reconstruction + stereo layers included in the fits
- Use Graph Neural Network (GNN)

Storage and CPU for Test Run

STORAGE: 1 PB storage on eos for raw data and simulation

CPU: Use local production with Krakow Cloud and new Bologna workstation

Bologna workstation

→ AMD Ryzen 9 5950X (16 cores, 32 threads) processor

→ RAM 4 x 32 = 128 GB

→ SSD 1 TB NVMe

→ Storage 4 x 12 TB HDD (7200rpm) in RAID10 configuration

Krakow Cloud Cluster

→ 100-200 cores

→ 1-2 TB storage
(transfer data run by run to eos)

Summary

- Software (FairMUonE) ready for data reconstruction and the full detector simulation for both test run and final detector configuration
- Track reconstruction: linear fit & Kalman filter
- Vertrex reconstruction: kinematic fit (signal) & Adaptive Vertex Fitter
- Global alignment with constraint on the absolute *z* distance using two-thin target system
- Machine learning techniques for MUonE trigger / online event reconstruction are being developed
- Storage and CPU resources for Test Runs allocated