The PandaRoot framework for simulation, reconstruction and analysis

The PandaRoot framework for simulation, reconstruction and analysis

Stefano Spataro for the PANDA collaboration

Dipartimento di Fisica Generale, Universitá di Torino and INFN, Italy

E-mail: stefano.spataro@to.infn.it

Abstract.   The PANDA experiment at the future facility FAIR will study anti-proton proton and anti-proton nucleus collisions in a beam momentum range from 2 GeV/c up to 15 GeV/c. The *PandaRoot* framework is part of the FairRoot project, a common software framework for the future FAIR experiments, and is currently used to simulate detector performances and to evaluate different detector concepts. It is based on the packages ROOT and Virtual MonteCarlo with Geant3 and Geant4. Different reconstruction algorithms for tracking and particle identification are under development and optimization, in order to achieve the performance requirements of the experiment. In the central tracker a first track fit is performed using a conformal map transformation based on a helix assumption, then the track is used as input for a Kalman Filter (package *genfit*), using GEANE as track follower. The track is then correlated to the pid detectors (e.g. Cerenkov detectors, EM Calorimeter or Muon Chambers) to evaluate a global particle identification probability, using a Bayesian approach or multivariate methods. Further implemented packages in PandaRoot are: the analysis tools framework *Rho*, the kinematic fitter package for vertex and mass constraint fits, and a fast simulation code based upon parametrized detector responses. PandaRoot was also tested on an Alien-based GRID infrastructure. The contribution will report about the status of PandaRoot and show some example results for analysis of physics benchmark channels.

1. Introduction

PANDA[1] is a fixed target experiment which will be built at the FAIR facility (Darmstadt, Germany), and it will study anti-proton collisions against protons/nuclei in a momentum range of up to 15 GeV/c. The physics program foresees an intensive study of charmonium and open charm states, searching also for new exotic states (such as glueballs, hybrids or molecules) at these energies, as well as study of electromagnetic form factors, Drell-Yan processes, generalized parton distributions and single/double hypernuclei spectroscopy. The high event rate in the detectors (around 10^7 Hz) causes a high probability of overlapping events during the data acquistion, and it does not allow a standard 1st level trigger selection and event building. This is a challenge not only for the frontend electronics and data acquisition system, but also for the software which has to deal with such data stream based on time stamps and not on an event basis.

The *PandaRoot* framework is the offline software of the PANDA experiment, based on ROOT[2] and the Virtual MonteCarlo[3] packages, and it will be discussed in this paper. First an introduction about the general structure of the framework will be presented, then the current software implementations will be described, in terms of geometry, tracking, particle identification and analysis.
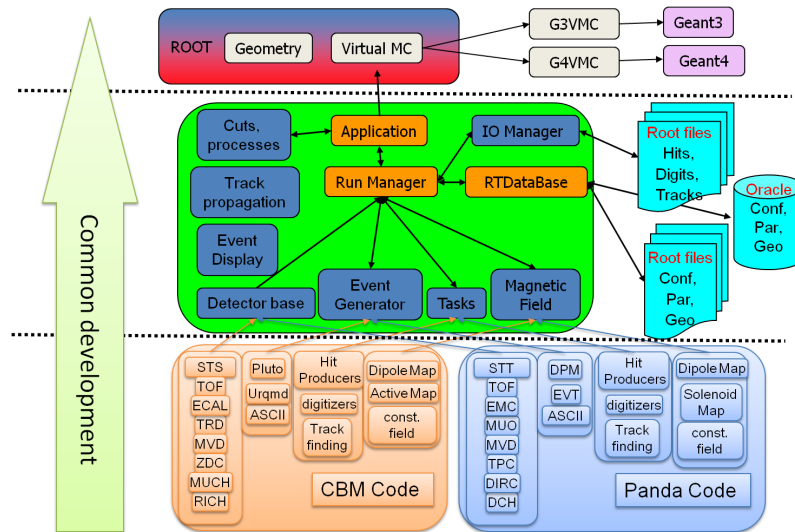
Figure 1. Structure of the framework. At the top level stay ROOT and the Virtual Montecarlo. On a step below, the FairRoot framework manages the general infrastructure with simulation and tasks. Detector specifics and reconstruction stay at the bottom level.

2. Structure of the framework

PandaRoot is the implementation of the PANDA detector simulation and reconstruction code inside the *FairRoot* framework (see Figure 1).

The *FairRoot*[4] software framework has been developed in order to have a common computing structure for the future FAIR experiments (i.e. PANDA, CBM and R3B). It is based on the ROOT package and it has a dynamic data structure based on trees and branches. *FairRoot* is developed by the GSI-IT department and it handles the basic features, such as the interfaces with simulation, tasks, parameter database and with the I/O. Detector simulation is handled by the Virtual MonteCarlo, which allows to use as transport models Geant3 or Geant4 with the same geometry definition and detector code, thus to compare the results for a better validation with data.

Detector specifics and reconstruction code are developed within *PandaRoot*. Several event generators (EvtGen, DPM, UrQMD, Pythia, Fluka) are used to create physics and background events, which are sent to the transport model by Virtual MonteCarlo. Detector geometries can be described using ASCII files or by ROOT objects, while the magnetic field map comes from TOSCA calculations. After, the detector responses are simulated by digitizers. Hits from tracking detectors are combined in order to form charged tracks, which are then correlated to PID detectors to form charged candidates; clusters from the electromagnetic calorimeter which are not correlated with tracks form the neutral candidates. Different particle identification algorithms assign a particle probability to each candidate, and this information is finally sent to the analysis, where the user can select his own particle identification criterium. Figure 2 shows a scheme of the data flow.

The software is maintained under different C++ compilers and several Linux distribution and Mac OSX, so that the user can install it in his own laptop or institute computer without any restrictions. Moreover, PandaRoot runs on an ALIEN2 based GRID[5].
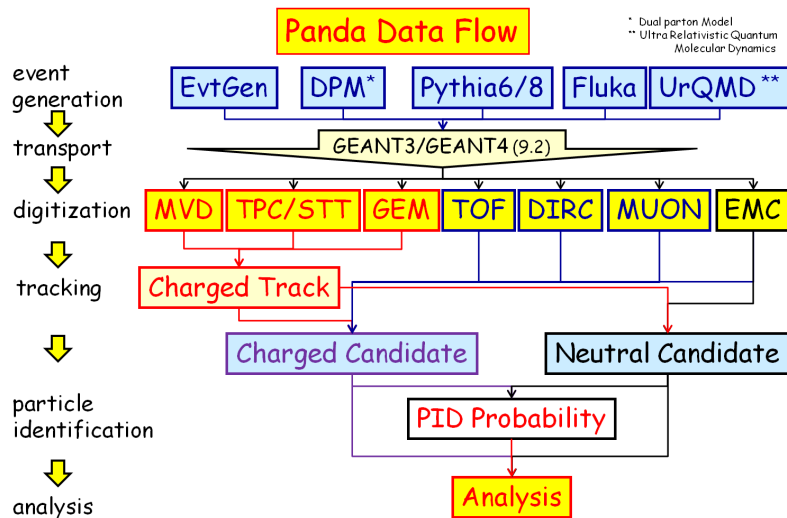
Figure 2. Scheme of the PandaRoot data flow.

3. Detector Geometry

Panda is a fixed target spectrometer, divided into two regions. Around the target region there is the so-called *Target Spectrometer*, which is inside a 2 T solenoid. A Micro Vertex Detector (MVD), several GEM detector planes, a DIRC Cherenov Detector and an Electromagnetic Calorimeter are placed inside. At present two different detector options are under evaluation for the central tracker, i.e. a Straw-Tube Tracker (STT) or a Time Projection Chamber (TPC); moreover, a Time-Of-Flight detector is under study. Outside the magnet coils, within the return iron yoke several planes of muon tracker (MDT) are placed. The region below $10°$ is covered by the *Forward Spectrometer*, where the tracking is performed by six planes of the Forward Tracking System (FTS), based on straw tubes, placed before, inside and after a dipole magnetic field; moreover TOF detectors, a Shashlik EM calorimeter and muon chambers are foreseen.
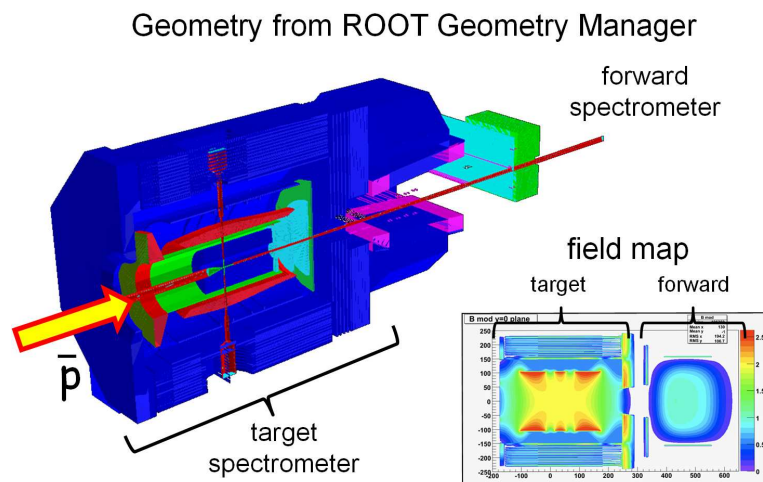

Figure 3. Geometry from the ROOT geometry manager and magnetic field map.

Geometry can be described using an ASCII format or using ROOT objects. An application has been developed converting directly CAD drawings (STEP format) from engineers into ROOT geometry format, and it is currently used for complicated structures such as the Micro Vertex Detector and the iron yoke. Magnetic field maps are converted directly from TOSCA calculations.

Furthermore, the PandaRoot event display is based on the Event Visualization Environment (EVE) in ROOT; Figure 3 shows the geometry layout from the geometry manager, together with the field map.

4. Tracking

A realistic reconstruction code is developed merging the informations from different tracking detectors. While the Forward Tracker is still under development, the target region tracking is almost complete and two different detector concepts for the central tracker, i.e. STT or TPC, are currently under investigation. The basic assumption for pattern recognition in this region is that the field is constant and the charged tracks follow helix trajectories. With the TPC option, hits from MVD, TPC and GEMs are matched together by means of conformal mapping[6], which transforms circles in the XY plane into straight lines for an easier pattern recognition. The same code works also for the STT detector, but at presend an improved algorythm based on a track follower is going to be finalized.

After the helix fitting, for a high resolution momentum reconstruction it is necessary to consider the magnetic field unhomogeneities (in particular in the GEM region), energy loss and the different data structure and error calculation of different detectors. This is achieved by the Kalman Filter GENFIT[7], developed within the PANDA collaboration, and using GEANE[8] as track follower. Figure 4 and Figure 5 show the achieved values of pattern recognition efficiency and momentum resolution.
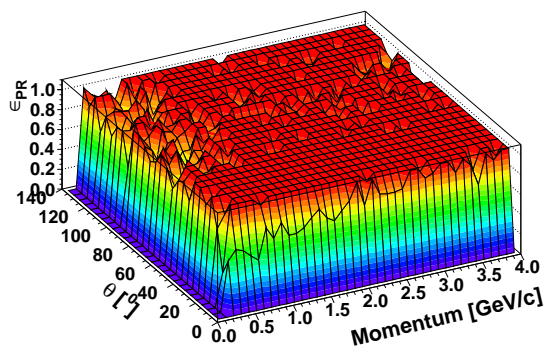

Figure 4. Pattern recognition efficiency for the MVD+TPC+GEM option, as a function of θ and momentum.
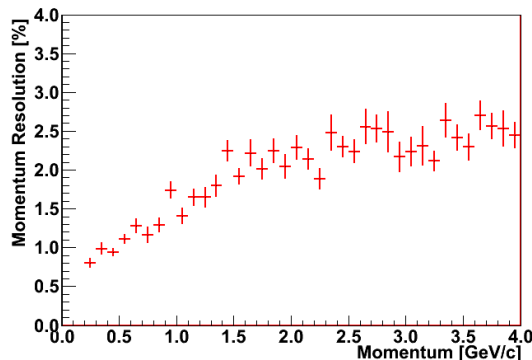
Figure 5. Momentum resolution for the MVD+TPC+GEM option as a function of momentum, after Kalman Filter.

5. Particle Identification

The Panda Spectrometer includes many different concepts of PID detectors, which have to be merged together, and most of the analysis need very high identification performances. For these reasons two approaches are followed to develop PID methods. Standard algorithms based on Bayesian approach are implemented for a fast identification, when only few paramers are used and signals can be easily separated. This is the case of the MVD and STT detectors, where

dE/dx is used for particle identification, and of the DIRC detector by Cherenkov angle, where at present only a fast simulation of the Cherenkov angle smeared by photon statistics is present (see Figure 6).

For a high level identification, in cases where very small signals have to be extracted from large data sets as a function of many parameters, a second approach based on multivariate analysis technique is under implementation. In particular, the TMVA toolkit[9] is under evaluation; Figure 7 shows the ROC curve (background rejection vs signal efficiency) for electron identification using the EMC cluster shower properties, comparing the performances for the different included algorithms using default parameters. Multivariate analyses are also evaluated to separate EMC clusters induced by photons from the π^0 ones, when the two outgoing photons have small opening angle and cannot be separated easily, and foreseen to identify muons from the pions in the Muon Tracker.

The design of the pid software is modular, storing the pid probability values coming from different algorithms in separated objects which can be merged to form a global probability according to the analysis specificity; this allows also the possibility to check different algorythm combinations to achieve the envisaged performances.
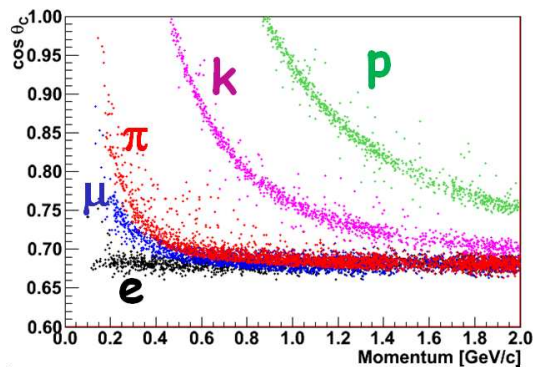

Figure 6. DIRC Cherenkov angle as a function of particle momentum - Separation between different particles (prob > 20%).
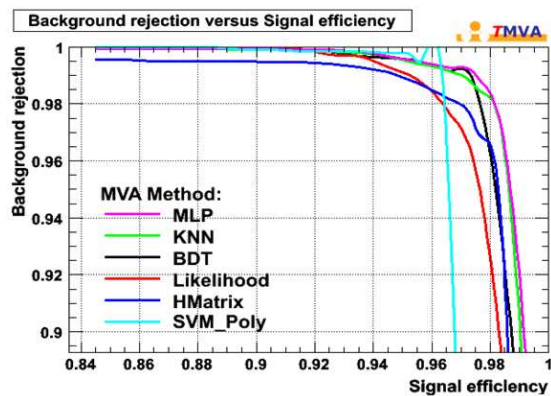

Figure 7. Electron identification: background rejection vs signal efficiency for different TMVA algorithms using EMC.

6. Analysis

For the high-level analysis the Rho[10] package has been included inside the PandaRoot framework; the user has the possibility to select *candidates* according to his pid selections and kinematic constraints, and combine them for the physics analysis. At present different kinematic and vertex fitters have been implemented and are under test.

In Figure 8 an example of the reconstruction of the channel $\bar{p}p \rightarrow \psi(3770) \rightarrow \bar{D}_0 D_0 \rightarrow K^+\pi^- K^-\pi^+$ is shown; the invariant mass distribution of the D candidates ($K^\pm\pi^\mp$) before the kinematic fitting is presented on the left, while the plot on the right shows the same distribution after the kinematic fitter, imposing constraints on the total invariant mass and on the vertex position. Recent studies show good results also for the vertex fitting of long living particles, i.e. Λ.

A *fast simulation*[11] package is also implemented, where the generated particles are filtered by a parametrization of acceptance, efficiency and resolution skipping the geant simulation and reconstruction; this allows to have a reliable simulation for different design options estimating the efficiency of various physics benchmark channels without requiring a time consuming full simulation and reconstruction.
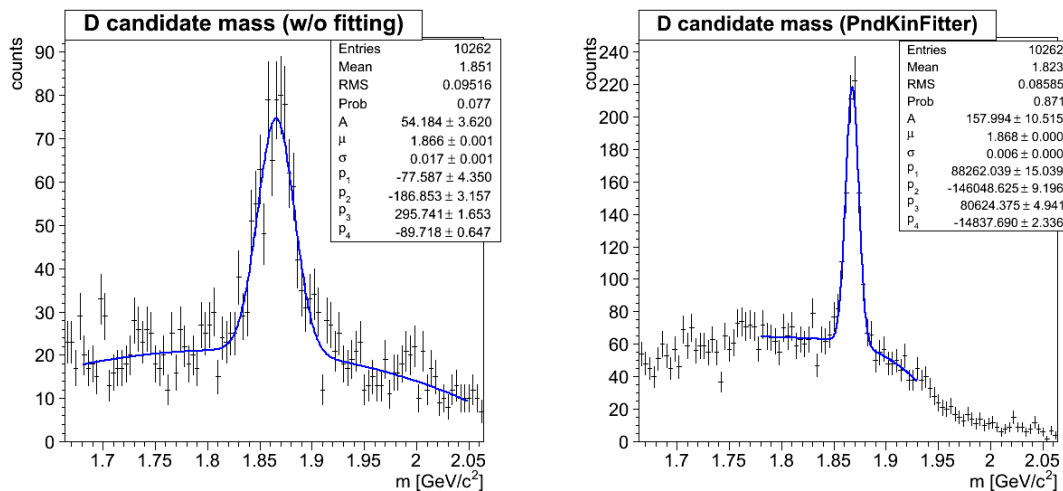
Figure 8. Analysis: reconstruction of the channel $\bar{p}p \to \psi(3770) \to \bar{D}_0 D_0 \to K^+\pi^- K^-\pi^+$: invariant mass distribution of the D candidates ($K^\pm\pi^\mp$) before (left) and after (right) the kinematic fitting.

7. Conclusions and outlook

PandaRoot is the official framework for the PANDA full simulation, reconstruction and analysis. The current state-of-art of the software was presented in this contribution, in terms of simulation, tracking, particle identification and analysis. The software is almost complete and it is currently being used for physics analysis, but still there are ongoing development activities to improve the global tracking and the particle identification.

The use of graphics processor units (GPUs) for event reconstruction is also under study; recent tests have shown that the implementation of parts of the tracking code (i.e. track fitting) with CUDA increases the computing performances of orders of magnitude compared to the CPUs[12]. One of the most challenging aspects of the PANDA computing is the high event rate ($\leq 2 \times 10^7$ evt/s) which will be handled without any hardware trigger, but streaming all raw data to the DAQ. In this case the *event* is not defined by the DAQ (event builder), but all the signals are stored with time stamps which need to be deconvoluted by the software. At present the software structure is going to be redesigned from an event basis to a time ordered simulation, and the reconstruction algorithms are needed to handle the additional time information and do the event building. This is a quite important and challenging project which has just started, and it will be finalized in the next months.

References

[1] The PANDA Collaboration 2009 *Physics Performance Report for PANDA: Strong Interaction Studies with Antiprotons* (arXiv:0903.3905v1)
[2] Brun R and Rademakers F 1997 *Nucl. Intr. Meth.* A **389** 81-86
[3] Hrinacova I *et al.* 2003 *Proc. of Computing in High Energy and Nuclear Physics (La Jolla)* THJT006
[4] http://cbmroot.gsi.de
[5] Protopopescu D *et al.* 2008 *GSI Scientific Report 2008* 241
[6] Yepes P *Nucl. Inst. Meth.* A **380** 582-585
[7] Höppner C *et al.* 2010 *Nucl. Intr. Meth.* A **620** 518-525
[8] Fontana A *et al.* 2008 *J. Phys.: Conf. Ser.* **119** 032018
[9] Hoecker A *et al.* 2007 *PoS A CAT* 040 [physics/0703039]
[10] http://savannah.fzk.de/websites/hep/rho/
[11] Götzen K 2008 *J. Phys.: Conf. Ser.* **119** 032020
[12] Al-Turany M *et al.* 2010 *J. Phys.: Conf. Ser.* **219** 042001