

# Customization of the GENFIT2 Fitting Package in $\overline{\text{P}}\text{ANDA}$

Elisabetta **Prencipe**<sup>1</sup> on behalf of the  $\overline{\text{P}}\text{ANDA}$  Collaboration<sup>a</sup>

<sup>1</sup>*Forschungszentrum Jülich, Germany*

**Abstract.** The availability of a cooled antiproton beam in the energy range from 2.0 to 5.5 GeV will allow the  $\overline{\text{P}}\text{ANDA}$  experiment, planned for operation in FAIR (Darmstadt, Germany), to perform a wide nuclear and particle physics program.  $\overline{\text{P}}\text{ANDA}$  is the only experiment worldwide in that energy regime that combines a solenoidal magnetic field ( $B = 2\text{ T}$ ) and a dipole field (maximum bending power equal to  $2\text{ Tm}$ ) in a fixed-target experiment. The offline tracking algorithm is developed within the PandaRoot framework, which is part of the FairRoot project. Here the GENFIT2 package is presented. The tool contains an implementation of the Kalman filter and the deterministic annealing filter for charged particle track reconstruction, together with a Runge-Kutta track representation. It also interfaces with Millipede II and RAVE, for alignment and vertex reconstruction, respectively. The performance of this track-fitting package for the  $\overline{\text{P}}\text{ANDA}$  experiment is shown for the first time, within the PandaRoot framework. For those channels where a good low momentum tracking is required, i.e.  $p_{\text{T}} < 350\text{ MeV}/c$ , an improvement by a factor of about two is shown.

## 1 Introduction

The GENFIT2 package [1] is a standalone tool that provides an extensible modular open-source framework, performing track fitting and other related tasks. It is available via *GitHub*.

Tracking is the core of physics analyses. All particle physics experiments contain different types of tracking detectors, to collect position information and extract from that signal the direction and the momentum of the particles. The tracking detectors must therefore be able to measure with good precision where the particle passed. When tracking is performed inside a magnetic field, it is possible to estimate the momentum of the particle by the trajectory curvature. For local track fitters, such as the Kalman filter, the additional knowledge of the particle mass allows to fully reconstruct the track, and infer the most accurate value of its momentum. Collections of measured detector signals (hits) must be associated to a track candidate, which is then fitted. After the fitting procedure, tracks are then refined by adding missing hits, and gathering the information from the different tracking subdetectors. Trajectories pointing to the same origin must be combined through vertex finding and fitting. All these tasks can be performed by the new GENFIT2 package, given a suitable collection of measurements and the description of the detector geometry.

GENFIT2 extends and improves the work previously performed by GENFIT [2], the external tracking tool available in PandaRoot releases (as part of PandaRoot [3] since 2007, and as independent

---

<sup>a</sup>e-mail: e.prencipe@fz-juelich.de

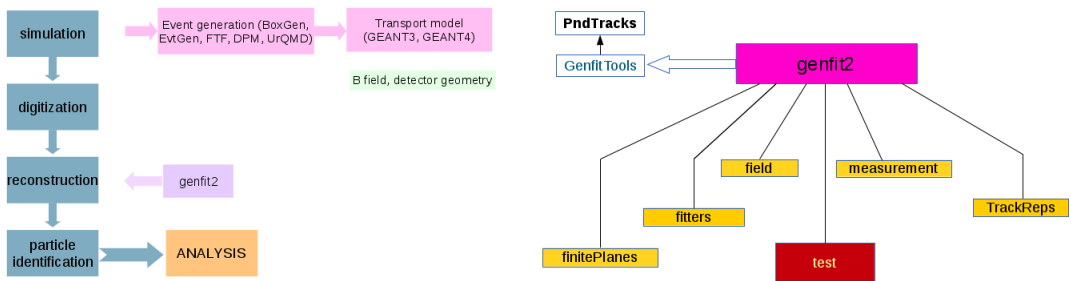
open source code since 2009). PandaRoot is based on FairRoot [4], and it is the official software framework for the Monte-Carlo (MC) simulations, event reconstruction and analysis procedures of the PANDA experiment. Several other experiments right now make use of GENFIT and/or GENFIT2; among these are Belle II, FOPI, SHIP, and BGO-OD.

In the following we will not describe the PANDA detector in detail, since it is extensively done elsewhere [5]. Here we mention only the main tracking detector components, as this report intends only to briefly document the implementation of the GENFIT2 classes inside PandaRoot, and its main features and performances therein. PANDA is a fixed-target experiment, with almost full coverage, including different kind of gaseous and solid state detectors. Its tracking system consists of a high performance silicon vertex detector (MVD), a straw-tube central tracker (STT), a gas-electron multiplier (GEM) tracker, and a forward tracking system (FTS).

The present documentation is based mostly on Refs. [1], [2] and [6], and shows additional plots based on the PandaRoot performance only. These have been obtained using the PandaRoot trunk-revision 28747, where GENFIT and GENFIT2 are both available.

The paper is organized as follows: a general introduction is given; in Section 2 the documentation of the GENFIT2 classes is provided, with a general overview of the structure of this modular code, and what was used then inside PandaRoot; Section 3 is dedicated to the GENFIT2 customization inside PandaRoot; Section 4 shows the visualization of a track in the PANDA event display; the final Section 5 summarizes the goals and achievements of this preliminary work.

## 2 Structure of the code



**Figure 1.** General structure of PandaRoot (left); customization of GENFIT2 in PandaRoot (right). RAVE and the GENFIT2 event display are not yet interfaced with PandaRoot. The PandaRoot package *GenfitTools* works as an interface between the GENFIT (GENFIT2) classes and PandaRoot: Track objects (GENFIT or GENFIT2 tracks) are transformed into PndTrack objects (PANDA tracks).

The GENFIT2 package is a C++ object-oriented modular tool. In Figure 1 the PandaRoot general structure and the GENFIT2 tool classes interfaced to it are shown. Event generation and analysis in PandaRoot is performed in four main steps:

1. Simulation. Events can be produced using different MC generators, that rely on different models to generate the physics process that the user wants to investigate. Several MC event generators are available in PandaRoot: *EvtGen*, Dual-Parton model (*DPM*), Fritiof model (*FTF*),

the Ultrarelativistic Quantum Molecular Dynamics model (*UrQMD*), a single particle generator (*BoxGen*), etc. The output of this process consists of two ROOT files: one containing the simulation parameters, and one the main information about MC tracks and points generated in the  $\overline{\text{PANDA}}$  subdetectors: track number, event number,  $x$ - $y$ - $z$  coordinates,  $x$ - $y$ - $z$  momentum coordinates, time, path length, etc. The detector simulation is performed either by GEANT3 or by GEANT4, according to the choice by the user executing the simulation macro.

2. Digitization. The digitization process runs after the simulation. It simulates the detector response to the passage of particles. At this stage a ROOT file is produced that saves the main information about hits in the  $\overline{\text{PANDA}}$  subdetectors.
3. Reconstruction. This is the step where GENFIT (or GENFIT2) is invoked, and where tracks are fitted using the GENFIT (or GENFIT2) methods. The output ROOT file stores all information related to the track combination from different subdetectors: time stamp, time stamp error, space and momentum coordinates of the first and last hit,  $\chi^2$  of the track fit, number of degrees-of-freedom NDof, fit quality flags, etc.
4. Particle Identification (PID). This last step runs after the reconstruction process. The PID technique uses a likelihood method, combining variables that help in separating tracks with different mass hypotheses. All variables involved in the PID process can be checked in the output file:  $dE/dx$ , Cherenkov angle, electromagnetic calorimeter EMC lateral moment, Zernike moments, etc.

After these four steps, the user can run the analysis macro, with different settings, depending on the specific simulated decay mode. The selection of events is done in this last step.

## 2.1 Track fitting in GENFIT2

In GENFIT2, track fitting is based on the following three components:

- measurements;
- track representations;
- fitting algorithms.

The measured coordinates and their covariance matrix are contained in objects of class `Measurements`, as described extensively in Ref. [1]. The GENFIT2 package includes predefined measurement classes for various detector types. For planar detectors, the detector plane is given by the detector geometry, whereas for `WireMeasurements` and `SpacepointMeasurements` virtual detector planes are built.

A track representation consists of a five-dimensional state vector that parameterizes the track and the corresponding covariance matrix. It also provides the possibility to extrapolate the parameters and covariance matrix from one point of the detector to another. The GENFIT2 tool implements a track representation based on a Runge-Kutta extrapolator (`RKTrackRep`, see Section 2.2). Fitting algorithms use the measurements and track representations to calculate fit results.

All information related to a track is kept in the `Track` object, which contains both the measurement and track representation objects. The `Track` holds a sequence of `TrackPoints`, which contain the `Measurements`.

GENFIT2 can work with five different track representations, one for each mass hypothesis: electron, muon, pion, kaon, and proton. The one that best fits to the measurement must be selected as the cardinal track representation. The selection can be done either by the user or by GENFIT2 itself. The

Track class also stores general information (number of iterations used by the fit, etc.) and fit properties ( $\chi^2$ , NDof, etc.).

To summarize, the GENFIT2 class called MeasurementFactory can build a Track out of a list of hits, an object of TrackCand. This Track can then be processed by the various fitting algorithms, described in the next Sections. The user can eventually decide which Track data to keep inside the object, depending on the needs.

## 2.2 Runge-Kutta track representation

The Runge-Kutta track representation (RKTrackRep) is based on a Runge-Kutta extrapolator from GEANT3 [7]. It uses five parameters, namely  $q/p, u', v', u, v$ , and their covariance matrix. GENFIT2 can actually use different coordinate systems: 6D-coordinates ( $x, y, z, p_x, p_y, p_z$ ), and the charge; local coordinates ( $q/p, u', v', u, v$ ); and global coordinates ( $x, y, z, T, q/p$ ). These variables are defined as the space coordinates  $x, y, z$ ; the momentum coordinates  $p_x, p_y, p_z$ ; the charge  $q$ ; the absolute value of the momentum  $p$ ; the direction unit vector  $T$ ; the local coordinates on the plane  $u$  and  $v$ ; the direction cosines relative to the plane reference frame  $u'$  and  $v'$ .

The Runge-Kutta calculations are used as track-following routines, to perform track propagation inside the detector geometry, by means of a sequence of steps. In each step, the physical processes that might occur in those particular experimental conditions are taken into account.

An abstract interface class interacts with the detector geometry, either via implementations using the ROOT class TGeoManager [8] or via the GEANT4 class G4Navigator [9]. During the fitting procedure, several effects are computed, that are related to material properties: the mean energy loss and its straggling for charged particles according to the Bethe Bloch formula and the Landau distribution, respectively; the multiple scattering (using the Highland-Lynch-Dahl formula [10, 11]), where the full noise matrix is calculated; the soft Bremsstrahlung energy loss and its straggling for electrons and positrons.

Step sizes for the Runge-Kutta propagation are evaluated inside the RKTrackRep, taking into account field inhomogeneities, track curvature and the above mentioned material effects. RKTrackRep provides different methods to find the point of closest approach (POCA) of the track to non-planar measurements. These are used to construct virtual detector planes, on which the measurement and its covariance matrix are projected.

## 2.3 Fitter implementation

GENFIT2 provides additional features compared to the old GENFIT package. Four different track-fitting algorithms are available:

- two Kalman filters, one which linearizes the transport around the state predictions, the other which linearizes around a reference track. This is an original feature of the GENFIT2 fitting tool;
- a deterministic annealing filter (DAF);
- a general broken lines fitter.

Every TrackPoint can store several Measurements, and the DAF can assign them weights. In GENFIT2, WireMeasurements and SpacepointMeasurements are implemented, depending on the detector and the user needs. The WireMeasurements class can also produce MeasurementOnPlane objects, representing the passage of the particle on both side of the wire. The GENFIT2 tool provides several ways to handle multiple measurements:

- the weighted average of the individual Measurements is calculated, and this is used by the DAF;
- the Measurements object closest to the state prediction or reference state is selected;
- in case the TrackPoint is a wire measurement, the side that is closest to the prediction or reference is selected; otherwise, the average is selected.

The WireMeasurements class has been tested in Belle II. It has been demonstrated that the best option to run the Kalman filter and obtain reasonable results is to select the side closest to the state prediction. Several iterations of the fit are run and in order to decide whether the fit has converged or not, the difference of the  $p$ -value between one iteration and the previous one is computed. When a predefined number of iterations is reached, the procedure stops. Also, the relative change in  $\chi^2$  between iterations is checked: if it is larger than 20%, then the fit continues. All parameters, such as the number of iterations and the threshold for the  $p$ -value, are user-adjustable.

## 2.4 Kalman filter implementation

The Kalman filter is a recursive version of a least-square fit. It iterates three steps: extrapolation, filter, and smoothing. It is based on a set of mathematical equations, that uses a series of measurements observed over time, containing “noise” (random effects) and other inaccuracies, and produces estimates of unobservable variables. It is a versatile tool with applications in many fields, and has been successfully used in high-energy physics for many years.

The Kalman filter implemented in GENFIT2 is adapted from the formulas given in Ref. [12], where the linearization is made around the state predictions. Additionally, a square-root formalism, adapted from Ref. [13], provides greater numerical stability at the cost of a larger execution time.

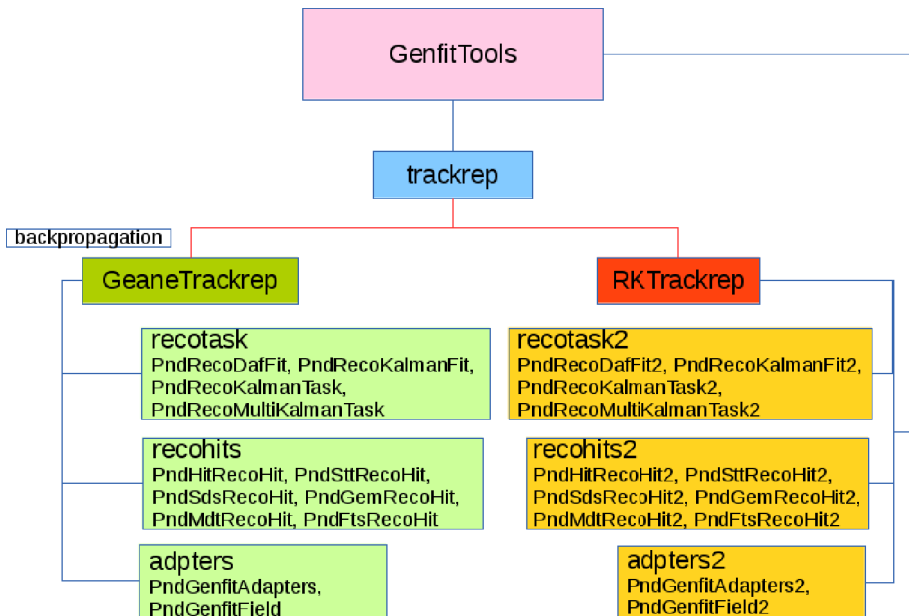
In general, state predictions start from a preliminary estimate of the track parameters. They might fall far from the real trajectories. As a consequence, using them as the linearization point is not an optimal solution; material and magnetic field lookups are also not done at the proper places. A general common solution is to linearize around reference states [12]: the start parameters to all TrackPoint objects are calculated from those. The smoothed states from the previous iterations are used as linearization points. However, for very small changes, reference states are not saving computing time. In addition to the convergence criteria detailed above, the fit is declared as converged if none of the reference states has been updated since the previous iteration.

## 2.5 Deterministic annealing filter

The deterministic annealing filter (DAF) [14] is a powerful tool used for the rejection of outlying measurements. It is a robust Kalman filter that makes use of a weighting procedure between iterations based on the measurement residuals to determine the proper weights. The user can select which of the two Kalman filter implementations should be used and can specify the annealing scheme.

The DAF is suited to resolve the left-right ambiguities of wire measurements. Both left and right measurements must be initialized. A standard solution is to start with the same weight, equal to 0.5. The wire positions are taken as measurements in the first iteration, and their covariance is twice the mean of the individual covariances. In this respect, all the wire positions have the same covariance value, independently on how far from the real trajectory they are. In this way the systematic indetermination affecting the covariance estimates can bias the fit result.

In GENFIT2, the fit is improved with a new method for the weight initialization: points with smaller drift radii get larger weights. In these cases it is safe to assume that the wire position is closer to the point where the particle actually passed; opposite applies to points with larger drift radii.



**Figure 2.** Structure of the *GenfitTools* package in the PandaRoot trunk-revision 28747.

## 2.6 Generalized broken lines fitter

The generalized broken lines method of track fitting [15] was implemented specifically for the purpose of alignment with Millipede II. It is equivalent to the Kalman filter (with thin scatterers instead of continuous materials), but fits tracks globally in one step, providing a natural interface to the Millipede II software. For alignment purposes, GENFIT2 also provides a set of interfaces for alignment parameters and derivatives which can be implemented by the detector classes.

## 2.7 Vertex reconstruction with *GFRave*

*GFRave*, an interface to the vertex-fitting framework RAVE [16], has been implemented in GENFIT2. RAVE is a detector-independent toolkit developed for the CMS experiment [17]. It deals with both finding (pattern recognition) and fitting (statistical estimation) of vertices. It takes a set of reconstructed particle tracks as an input, and produces reconstructed interaction vertices as an output. *GFRave* will allow precise and fast vertex reconstruction, although it is not yet interfaced with PandaRoot.

## 3 Performance in PandaRoot

An interface was built between GENFIT2 and PandaRoot, known as the package *GenfitTools*. The basic structure of *GenfitTools* is the same of the old GENFIT package, but with a few modifications, due to the new classes introduced now in GENFIT2 (see Figure 2). The new *GenfitTools* structure was created to allow the use of both GENFIT and GENFIT2 as fitting tool, and then to compare results.

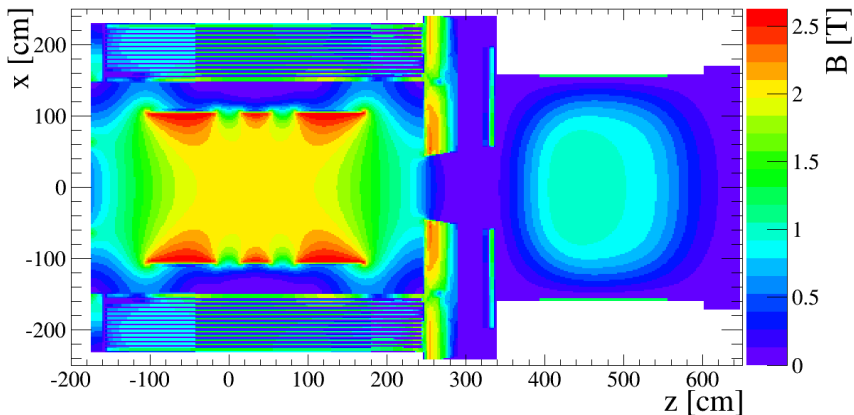
*GenfitTools* includes several classes: adapters and adapters2; recohits and recohits2; recotasks and recotasks2; and trackrep.

The classes `adapters`, `recohits`, and `recotasks` refer to the GENFIT interface to the PandaRoot classes, and the classes `adapters2`, `recohits2`, `recotasks2` refer to the GENFIT2 interface to the PandaRoot classes. The class `recohits` takes care of the hit reconstruction in different detectors; in the class `trackrep` the information about track representation is accessible; the class `recotasks` contains the functions needed to reconstruct tracks; the class `adapters` (`adapters2`) is the PandaRoot class where the transformation of a GENFIT (GENFIT2) track to the PndTrack is performed, and vice versa. The user should use objects of the PndTrack only, so whenever a GENFIT2 update is provided, information is properly stored in the PandaRoot objects and there is no need to change other PandaRoot packages.

In order to evaluate the capabilities of the GENFIT2 and the new *GenfitTools* package, first tests have been performed using the  $\bar{\text{P}}\text{ANDA}$  package *BoxGen*.

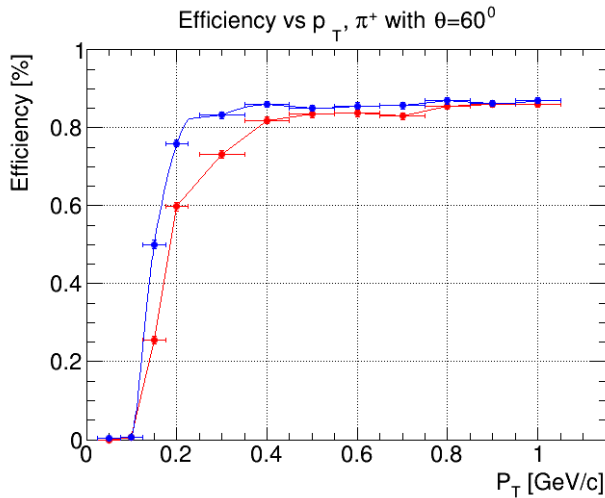
Five particle hypotheses have been tested: electrons, muons, pions, kaons, and protons. We have generated separately single particle events for each type, using different values of  $p_T$ . We tested particles at a fixed polar angle  $\theta = 60^\circ$ . We also checked the GENFIT2 performance in PandaRoot by changing the beam momentum, to evaluate the resolution and pull distributions of the tracking parameters.

The magnetic field map used in these tests was a realistic one, as shown in Figure 3. This is obtained by converting TOSCA [18] calculations into PandaRoot format.



**Figure 3.** Magnetic field scheme in  $\bar{\text{P}}\text{ANDA}$ . In the central part of the detector, the  $B$  field is constant and equal to 2 T, and in the dipole area the maximum bending power is 2 Tm.

Figure 4 shows the reconstruction efficiency of single pions, simulated with the MC *BoxGen* package, as a function of  $p_T$ . A comparison between GENFIT2 (blue points), and GENFIT (red points) is given: a clear improvement for low momentum tracks is observed with GENFIT2. In this test, 25000 single particle events were generated, for each  $p_T$  value. Efficiency is evaluated as the ratio between the number reconstructed events, selected in a  $3\sigma$  window around the momentum peak, and the number of generated events. The resolution  $\Delta p/p$  at 1 GeV/c is about 1.8%, and the  $\phi$  angle resolution is about 2.5 mrad. The efficiency shown in the plot takes into account the detector acceptance and a realistic simulation of the detector geometry, digitization and track reconstruction.



**Figure 4.** Reconstruction efficiency of charged pions emitted at  $\theta = 60^\circ$ , as a function of  $p_T$ . Single particle events were produced by the MC package *BoxGen*. The blue points show the performance of the GENFIT2 package in the PandaRoot trunk-revision 28747, compared to GENFIT (red points). A clear improvement is observed for  $p_T < 350$  MeV/c when using GENFIT2.

A transverse momentum cut larger than 50 MeV/c is applied, because the Pattern Recognition tool requires at least in the STT. The minimum radius of the STT volume is 16 cm, so the curvature radius of the charged particle must be larger than 8 cm. This requirement leads to a minimum  $p_T$  value of 48 MeV/c for every tracked particle in the PANDA detector.

Figure 5 shows the Gaussian fit to the distributions of the standardized residuals of the track parameters  $d_0$  and  $z_0$ , for  $p_T = 1.0$  GeV/c. These two variables are defined as the distance of the point of closest approach to the origin (0,0,0) in the  $x$ - $y$  plane and  $z$  axis, respectively. The standardized residual is defined as the difference of the generated values and the reconstructed value of the variable, divided by its error. The width of the Gaussian fit to the residuals is close to one, which is expected.

## 4 Visualization

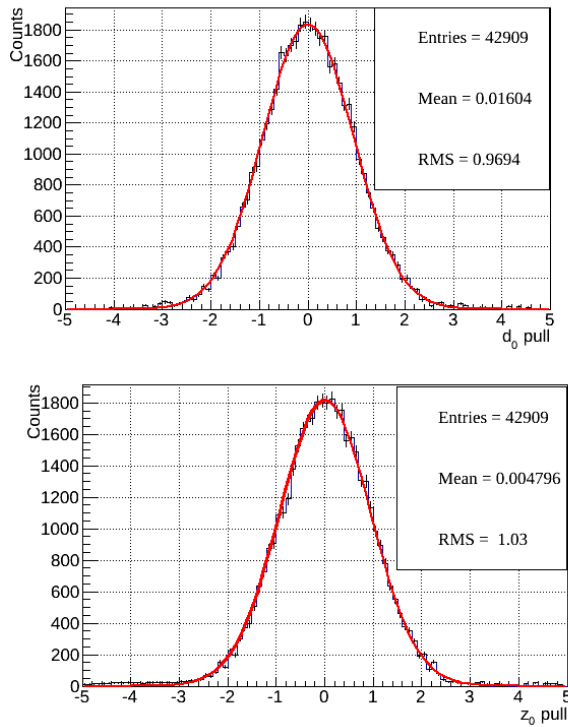
The GENFIT2 package also provides a 3D event display. Not only the fitted tracks can be shown (also "live" while performing the fit), but the geometry, all the hits (measured, extrapolated, and smoothed) and their reference planes can be drawn as well.

In PandaRoot, the package called *PndEventDisplay* provides similar features, derived from the ROOT EVE package [19]. Therefore, the GENFIT2 event display is not interfaced with PandaRoot. In Figure 6, an example of events observed with *PndEventDisplay* is shown.

## 5 Conclusions

We have demonstrated that GENFIT2 improves low momentum tracking for pions. Similar studies have been performed using other particle types: electron, muon, kaon, and proton. The new track representation available in GENFIT2 works better than the previous one used in PandaRoot. Examples





**Figure 5.** Distribution of the standardized residuals of  $d_0$  (top) and  $z_0$  (bottom), and the Gaussian fit to the distributions, for  $p_T = 1.0 \text{ GeV}/c$ .

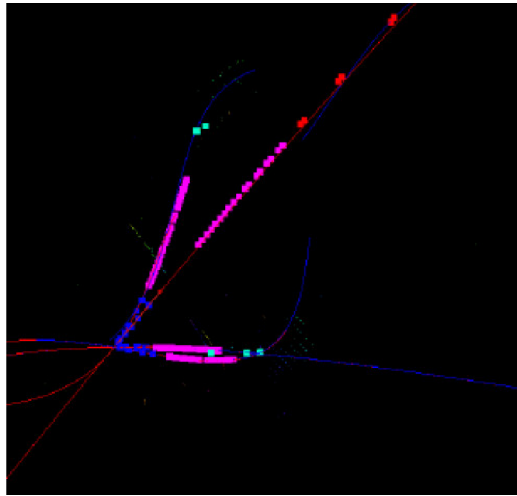
with electron reconstruction were shown during the workshop in Ref. [20]. The future plan is to use GENFIT2 as a part of a PandaRoot release for MC simulations. Tests are ongoing on some benchmark channels of the PANDA physics program, more sensitive to the reconstruction of low momentum tracks, for instance channels involving the reconstruction of  $D_s$  mesons. Preliminary results are promising.

## Acknowledgments

Special thanks go to Lia Lavezzi for the extremely valuable work on tracking tools performed inside the PandaRoot framework; and to Johannes Rauch and Tobias Schlüter, main authors of the GENFIT2 package [1], by which most of Section 2 of this report is inspired, for kind clarifications and fruitful discussions, and for their hard work in developing this track fitting tool.

## References

- [1] J. Rauch, T. Schlüter, J. Phys. Conf. Ser. **608**, 12042 (2015)
- [2] C. Höppner et al., Nucl. Inst. & Meth. A **620**, 518 (2010)
- [3] D. Bertini et al., J. Phys. Conf. Ser. **119**, 32011 (2008)



**Figure 6.** Example of an event in the  $\bar{P}$ ANDA detector. The different colors refer to hits and tracks in the different sub-detectors: blue for MVD, magenta for STT, red for GEM, cyan for SciTil (scintillator tile hodoscope).

- [4] S. Spataro, J. Phys. Conf. Ser. **396**, 22048 (2012)
- [5] W. Erni et al. (PANDA collaboration) (2009), [arXiv:0903.3905](https://arxiv.org/abs/0903.3905)
- [6] I. Gaponenko et al., Comp. Phys. Comm. **140**, 406 (2000)
- [7] R. Brun et al., *GEANT: detector description and simulation tool* (CERN, Geneva, 1994), <http://cds.cern.ch/record/1082634>
- [8] R. Brun, F. Rademakers, Nucl. Inst. & Meth. A **389**, 81 (1997)
- [9] S. Agostinelli et al. (GEANT4), Nucl. Inst. & Meth. A **506**, 250 (2003)
- [10] V.L. Highland, Nucl. Inst. & Meth. **129**, 497 (1975)
- [11] G.R. Lynch, O.I. Dahl, Nucl. Inst. & Meth. B **58**, 6 (1991)
- [12] R. Frühwirth et al., *Data Analysis Techniques for High-Energy Physics* (Cambridge University Press, Cambridge, UK, 2002)
- [13] B.D.O. Anderson, J.B. Moore, *Optimal Filtering* (Dover Publications, Mineola, NY, 1979)
- [14] R. Frühwirth, A. Strandlie, Comp. Phys. Comm. **120**, 197 (1999)
- [15] C. Kleinwort, Nucl. Inst. & Meth. A **673**, 107 (2012)
- [16] W. Waltenberger, IEEE Trans. Nucl. Sci. **58**, 434 (2011)
- [17] S. Chatrchyan et al. (CMS collaboration), JINST **3**, 361 (2008)
- [18] Vector Fields Limited, *The TOSCA Reference Manual*, Vol. VF068814 (24 Bankside, Kidlington, Oxford, OX5 1JE, England, 1988)
- [19] M. Tadel, J. Phys. Conf. Ser. **219**, 42055 (2010)
- [20] E. Prencipe, *Customization of GENFIT2 fitting package for PANDA*, Talk at “Connecting the Dots 2016” workshop (2016)