

Customization of the GENFIT2 fitting package in $\bar{\text{P}}\text{ANDA}$

Elisabetta **Prencipe**, on behalf of the $\bar{\text{P}}\text{ANDA}$ Collaboration^{1,a}

¹ *Forschungszentrum Jülich*

Abstract. $\bar{\text{P}}\text{ANDA}$ is a planned experiment at FAIR (Darmstadt, Germany) with a cooled antiproton beam in a momentum range [1.5; 15] GeV/c, allowing a wide physics program in nuclear and particle physics. It is the only experiment worldwide, which combines a solenoid field ($B=2\text{T}$) and a dipole field ($B=2\text{Tm}$) in an experiment with a fixed target topology, in that energy regime. The tracking system of $\bar{\text{P}}\text{ANDA}$ involves the presence of a high performance silicon vertex detector, a Straw-tubes central tracker, a GEM detector, a forward tracking system, and a luminosity monitor. The offline tracking algorithm is developed within the PandaRoot framework, which is a part of the FairRoot project. The algorithm here presented is based on a tool containing the Kalman Filter equations and a deterministic annealing filter (*GENFIT2*). *GENFIT2* offers to users also a Runge-Kutta track representation, and interfaces with Millepede II (useful for alignment) and RAVE (vertex finder). The Kalman-Filter-based algorithms have a wide range of applications; among those in particle physics they can perform extrapolations of track parameters and covariance matrices. The impact on physics simulations performed for the $\bar{\text{P}}\text{ANDA}$ experiment is showed for the first time, with the PandaRoot framework: improvement is shown for those channels where a good low momentum tracking is required ($p_T < 350$ MeV/c) of about a factor 2.

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 need to identify and classify processes based on detector signals. The task which reconstructs the particle trajectories, combining these signals, is called “tracking”. Collections of measured hits must be associated to a track candidate, which is then fitted. Tracks are then refined by adding missing hits and gathering the information from the different tracking subdetectors. More trajectories pointing to the same origin must be combined through vertex finding and fitting. With provided suitable collections of measurements and detector geometry, all these tasks can be performed with the new *GENFIT2* package.

GENFIT2 extends and improves the work previously performed by *GENFIT* [2], the external tracking tool available in the PandaRoot releases since 2009 up to now. PandaRoot [3] is based on FairRoot [4], and it is the official software framework for the Monte Carlo (MC) simulations, event reconstruction and analysis of the $\bar{\text{P}}\text{ANDA}$ experiment. Actually the old revision called *GENFIT* was developed as part of PandaRoot, since 2007; later on it became an independent open-source code.

^ae-mail: e.prencipe@fz-juelich.de

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. It holds a sequence of *TrackPoint* objects, which can contain *measurements*, *FitterInfo* objects, which hold all fitter specific information.

A *Track* object contains one or more track representations, representing the particle hypotheses that should be fitted. One of them must be selected as the cardinal track representation. This can be done by the user or by *GENFIT2*, which selects the track representation that best fits the measurements. The *Track* also contains a *FitStatus* object, which stores general information (number of iterations, convergence, etc.) and fit properties (χ^2 , NDof, *p*-value, track length, etc.). The track candidate (*TrackCand*) serves as a helper class, basically storing indices of raw detector hits in *TrackCandHit* objects, which can also be overloaded by the user to store additional information. *WireTrackCandHit* objects can store the information to deal with the left-right ambiguity (see Section 2.5).

GENFIT2 MeasurementFactory can build a *Track* from a *TrackCand*. This *Track* can then be processed by the various fitting algorithms. After fitting, *Track* objects contain a lot of data. Usually, not all of this information needs to be stored on disk. The user can decide which data to keep, 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 [9]. It makes use of 5 parameters, namely q/p , u' , v' , u , v , and their covariance matrix¹. As track following routines, to perform track propagation, the Runge-Kutta calculations are used.

An abstract interface class interacts with the detector geometry. Implementations using the ROOT class *TGeoManager* [10] and Geant4 class *G4Navigator* [11] are available. During fitting, material properties are used to calculate the following effects: mean energy loss and its straggling for charged particles according to the Bethe Bloch formula and Landau distribution, respectively; multiple scattering (using the Highland-Lynch-Dahl formula [12]), where the full noise matrix is calculated; soft Bremsstrahlung energy loss and related straggling for e^- and e^+ . The step sizes used for the Runge-Kutta extrapolation should be as large as possible to save unnecessary computation, while still being small enough to keep errors reasonably small. An adaptive step-size calculation is done in the *RKTrackRep*, taking field inhomogeneities and curvature of the particle into account. To calculate material effects correctly, extrapolation stops at material boundaries and steps are dimensioned in order not to exceed a maximum relative momentum loss in the material. *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.

The projection of the measurement covariance on the virtual detector plane gives the covariance in the plane.

Prior the attempt of using the *GENFIT2 RKTrackRep*, PandaRoot has made use of the *GeaneTrackRep*, based on GEANE [13], since the old *RKTrackRep* available in the former *GENFIT* tool was not very well performant. Lots of improvements have been done in the past years, and preliminary results are now available. GEANE is a track follower distributed within the *Geant3* package. It is written in FORTRAN, and a C++ interface has been developed in FairRoot.

¹*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 $(x, y, z, \mathbf{T}, q/p)$. The variables above 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 \mathbf{T} ; the local coordinates on the plane u and v ; the direction cosines relative to the plane reference frame u' and v' .

2.3 Fitter implementation

Four different track-fitting algorithms are currently implemented:

- two Kalman filters, one which linearizes the transport around the state predictions, and one which linearizes around a reference track; the latter is innovative in *GENFIT2*, compared to *GENFIT*;
- a deterministic annealing filter (*DAF*);
- a general broken lines (GBL) fitter.

The *GENFIT2* tool provides the possibility to store several *Measurements* of the same type in one *TrackPoint*, mainly for using the *DAF* to assign weights to them. Wire measurements also produce two *MeasurementOnPlane* objects, representing the passage of the particle on either side of the wire.

The *GENFIT2* tool provides several ways to handle multiple measurements:

- the weighted average of the individual *Measurements* is calculated. This is used by the *DAF*;
- the *Measurement* closest to the state prediction or reference state is selected;
- if the *TrackPoint* has a wire measurement, the side that is closest to the prediction or reference is selected; otherwise the average is selected.

For wire measurements, it turned out that selecting the side closest to the state prediction is the best option for the Kalman fitters. As convergence criteria, a minimum and maximum number of iterations can be set, which are 2 and 4 by default. After the minimum number of iterations, *GENFIT2* checks if the p -value has changed less than a certain amount since the previous iteration; the default is 10^{-3} . However, tracks with a p -value close to zero are often considered as converged with this criterion, even though the χ^2 , albeit big, is still changing significantly, indicating that the fit is still improving. This occurs often for tracks that are given bad start parameters. To cure this issue, a non-convergence criterion has been introduced: if the relative change in χ^2 from one iteration to the next is larger than 20%, the fit will continue. Again, this number is user-adjustable.

2.4 Kalman fitter

The Kalman fitter is a procedure to minimize χ^2 , and is fulfilled with 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 variables) and other inaccuracies, and produces estimates of unknown variables;
- iterates least square method in two steps: prediction, and update;
- is useful in several fields, with different versatile applications: it is used in high energy physics since many years, with success.

The Kalman fitter method implemented in *GENFIT2* is adapted from the formulas given in Ref. [14], with linearization around the state predictions. Optionally, a square-root formalism, adapted from Ref. [15] to the usecase relevant to track fitting, provides greater numerical stability at the expense of execution time.

2.4.1 Kalman fitter with reference track

State predictions start in general from a prefit. They may stay very far from actual trajectories. Consequently, linearizing around them is not optimal; material and magnetic field lookups are also not done at the proper places. It is therefore common to linearize around reference states [14], which are calculated by extrapolating the start parameters to all *TrackPoint* objects. At later iterations, the smoothed states from the previous iterations are used as linearization points. However, if the change would be very small, reference states are not updated, saving computing time. It is also possible to let the fitter sort the measurements along the reference track, which can improve fitting accuracy. In addition to the convergence criteria detailed above, the fit is regarded 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) [16] is a powerful tool for the rejection of outlying measurements. It is a Kalman filter that uses a weighting procedure between iterations based on the measurement residuals to determine the proper weights. The user can select which of *GENFIT2* two Kalman-fitter implementations should be used and specify the annealing scheme. The DAF is also perfectly suited to resolving the left-right ambiguities of wire measurements.

However, a problem can occur: the weights of the *MeasurementOnPlane* objects must be initialized. The basic solution is to initialize both left and right measurements with a weight of 0.5. Effectively, the wire positions are taken as measurements in the first iteration, and their covariance is twice the mean of the individual covariances. So, all the wire positions have the same covariance, no matter how far from the actual trajectory they are. This systematically false estimate of the covariances biases the fit. *GENFIT2* implements a novel technique to initialize the weights that improves the fitting efficiency: measurements with larger drift radii are assigned smaller weights, leading to larger covariances since the wire position is expected to be farther away from the trajectory. In contrast, measurements with smaller drift radii, which are closer to the trajectory, get larger weights.

2.6 Generalized broken lines fitter

The generalized broken lines (GBL) method of track fitting [17] was implemented especially for the purpose of alignment with Millipede II. It is equivalent to the Kalman fitter (with thin scatterers instead of continuous materials), but fits tracks in their entirety 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, has been implemented in *GENFIT2*. Rave [18] is a detector-independent toolkit for vertex reconstruction, originally developed for the CMS experiment [19]. *GFRave* takes full advantage of the *GENFIT2* material model and the sophisticated algorithms of Rave, allowing for precise and fast vertex reconstruction. *GFRave* is not interfaced with PandaRoot, for the time being.

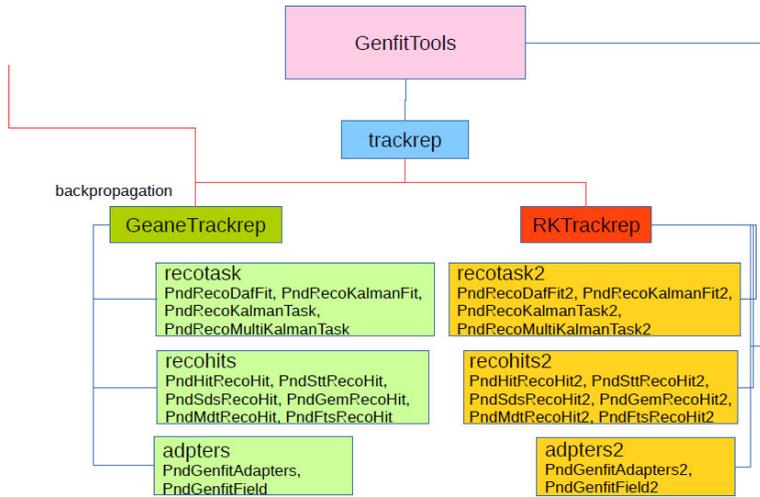


Figure 3. Structure of the *GenfitTools* package, in the PandaRoot trunk 28747.

3 Performance in PandaRoot

An interface is built between *GENFIT2* and PandaRoot: it is the so called package *GenfitTools*. The basic structure of *GenfitTools* is the same used to interface the old *GENFIT* package, but with a few modifications, due to the new classes introduced now in *GENFIT2* (see Fig. 3). The new *GenfitTools* structure was created to allow users to use both, *GENFIT* or *GENFIT2*, as fitting tool, and then compare results.

GenfitTools includes several objects:

- adapters(2);
- recohits(2);
- recotasks(2);
- trackrep.

The *adapters*, *recohits*, *recotasks* refer to the *GENFIT* interface to the Pandaroot classes; the *adapters2*, *recohits2*, *recotasks2* refers to the *GENFIT2* interface to the PandaRoot classes. The user should use *PndTrack* objects only, so whenever a *GENFIT2* update is provided, informations are properly stored in the PandaRoot objects and there is no need to change additional PandaRoot packages.

In order to evaluate the capabilities of the *GENFIT2* and the new *GenfitTools* package, first tests have been performed using the MC Box Generator.

Five particle hypotheses have been tested: electrons, muons, pions, kaons and protons. We have generated separately single particle events for each type, using different p_T values. We tested particles at a fixed θ 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 Fig. 4. This is obtained by converting TOSCA [20] calculations into PandaRoot format.

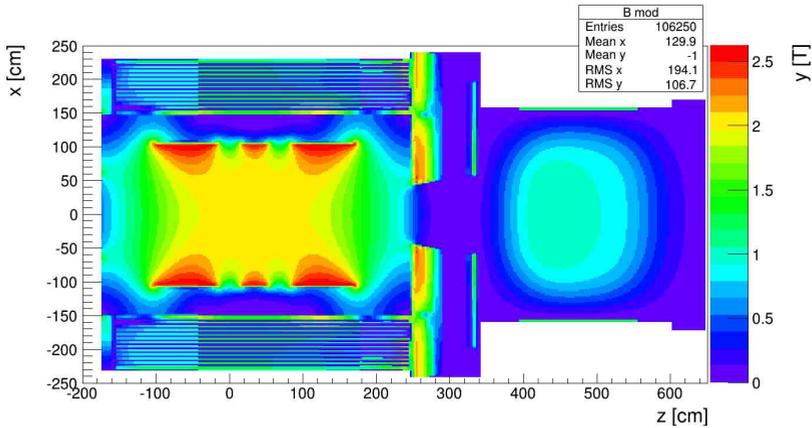


Figure 4. Magnetic field scheme in \bar{P} ANDA: in the central part of the detector it is constant and equal to 2T; in the dipole area it is 2 T·m.

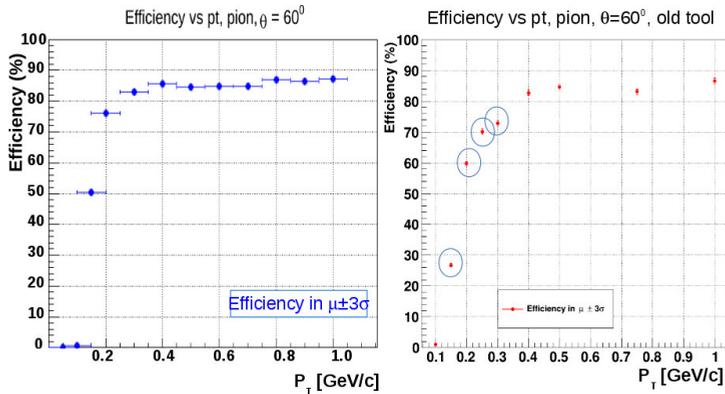


Figure 5. Efficiency versus the transverse momentum of charged pions, emitted at $\theta = 60^\circ$. The MC Box generator produces single particle events. The left plot shows the performance of the *GENFIT2* package in the PandaRoot trunk revision 28747, in comparison with *GENFIT* (right plot). A big improvement is observed for $p_T < 350$ MeV/c, when using *GENFIT2*.

Figure 5 shows the efficiency change of a single pion event reconstruction, simulated with the MC Box Generator, when changing p_T . A comparison between the *GENFIT2* (Fig. 5-left), and *GENFIT* (Fig. 5-right) is given, where a clear improvement for low momentum tracks is observed in *GENFIT2*. In this test, 25000 single particle events were generated, for each p_T value. Efficiency is evaluated as the ratio between the reconstructed events, selected in a 3σ window around the momentum peak, and the generated events. The $\Delta p/p$ resolution at 1 GeV/c is about 1.8%, and the ϕ angle resolution is of about 2.5 mrad.

A momentum cut larger than 50 MeV/c is set up: the reason is that at least 1 hit is supposed to be found in the STT, to let the Pattern Recognition tool work. 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 $\bar{\text{PANDA}}$ detector.

Figure 6 shows the gaussian fit of the pull distributions of tracking parameters, i.e. 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 xy plane and z axis, respectively. The width of the gaussian fit is close to the unit, as expected.

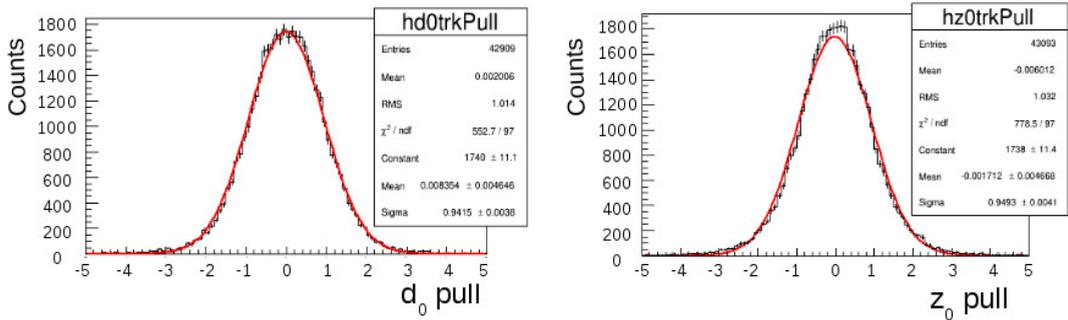


Figure 6. Generated values are subtracted to the reconstructed values of the variables d_0 and z_0 , then divided by their error. The plots obtained for $p_T = 1.0$ GeV/c show the pull of the distributions as indicated by the labels.

4 Visualization

The *GENFIT2* package features a sophisticated 3D event display to visualize fitted tracks. Detector geometry, measurements, detector planes, reference tracks, forward and backward fits (predictions and updates), smoothed tracks, and covariances of measurements and tracks can be drawn. Tracks can be refitted with different algorithms and settings, and fit results can be viewed instantly.

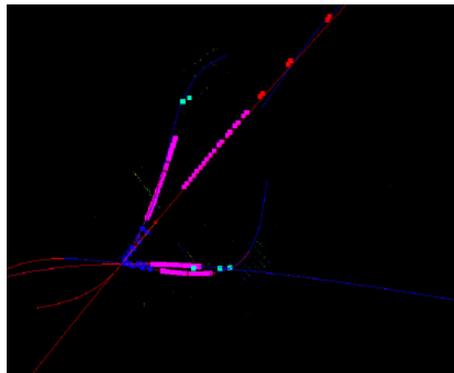


Figure 7. Example of an event where the $\bar{\text{PANDA}}$ detector geometry is shown, together with the track. The color code to read events is the following: blue for MVD, magenta for STT, red for GEM, cyan for SciTil.

In PandaRoot, the package called *PndEventDisplay* provides same features, derived from the ROOT EVE package [21]. So, the *GENFIT2* event display was not interfaced with PandaRoot. In Fig. 7 an example of events observed with *PndEventDisplay* are reported.

5 Conclusions

We have demonstrated that *GENFIT2* improves low momentum tracking, for pions. Similar studies have been performed using all other mass hypotheses: electron, muon, kaon and proton. The new track representation, available in *GENFIT2*, looks working better than the previous one used in PandaRoot, taken from *GEANE*. Examples with electron reconstruction were shown during the workshop in Ref. [22]. The future plan is to use *GENFIT2* as part of a PandaRoot release for MC simulations. Tests are ongoing on some benchmark channels of the $\bar{\text{P}}\text{ANDA}$ physics program, more sensitive to the reconstruction of low momentum tracks, i.e. channels involving the reconstruction of D_s mesons. Preliminary results are promising.

References

- [1] J. Rauch and T. Schlüter, arXiv:1410.3698 [physics.ins-det].
- [2] C. Höppner *et al*, NIM A **620**, 518-525 (2010).
- [3] D. Bertini *et al*, JOPCS **119** 032011 (2008).
- [4] S. Spataro, JOPCS **396** 022048 (2012).
- [5] W. Erni *et al*, arXiv:0903.3905 [hep-ex].
- [6] D. N. Brown *et al*, *The BaBar Track Fitting Algorithm*, Proceedings of CHEP 2000, Padova, Italy (2000).
- [7] T. Bilka *et al*, IEEE Trans. Nucl. Sci. 62(3), 1-7 (2015).
- [8] V. Blobel, NIM A **566**, 5-13x (2006).
- [9] GEANT - **CERN Program Library Long Write-up**, W5013 (1993).
- [10] R. Brun and F. Rademakers, AIHENP96 Workshop (Lausanne) **389**, 81-86 (1996).
- [11] GEANT4 Coll., GEANT 4 NIM A **506**, 250-303 (2003).
- [12] V.L.Highland NIM **129**, 479 (1975); G.R. Lynch and O.I. Dahl NIM B **58**, 6 (1991).
- [13] M. Innocente *et al*., **CERN Program Library**, W5013-E (1991).
- [14] R. Frühwirth *et al*, *Data Analysis Techniques for High-Energy Physics*. Cambridge University Press (2000).
- [15] B. D.O. Anderson and J. B. Moore, *Optimal Filtering*. Dover Publications (1979).
- [16] R. Frühwirth and A. Strandlie, *Track Fitting with Ambiguities and Noise: A Study of Elastic Tracking and Nonlinear Filters*. Computer Physics Communications, 197-214 (1999).
- [17] C. Kleinwort, NIM A **673**, 107-110 (2012).
- [18] W. Waltenberger. IEEE Trans. Nucl. Sci., **58**, 434-444 (2011).
- [19] CMS Collaboration, JINST, 0803:S08004 (2008).
- [20] *The TOSCA Reference Manual*, Vector Fields Limited, 24 Bankside, Kidlington, Oxford, OX5 1JE, England.
- [21] “EVE - the Event Visualization Environment of ROOT”, <https://root.cern.ch/eve>.
- [22] E. Prencipe, “Customization of GENFIT2 fitting package for $\bar{\text{P}}\text{ANDA}$ ”, <https://indico.hephy.oeaw.ac.at/event/86/session/3/contribution/24/material/slides/0.pdf>. “Connecting the dots” workshop, Vienna, 22-24 Feb. 2016.