

Customization of the general fitting tool GENFIT2 in PandaRoot

Elisabetta Prencipe¹, Stefano Spataro², and Tobias Stockmanns¹ on behalf of the $\bar{\text{P}}\text{ANDA}$ Collaboration

¹Forschungszentrum Jülich, Institute IKP1, Jülich 52425, Leo Brandt strasse, DE

² Università di Torino and INFN, via P. Giuria 1, Torino 10125, IT

E-mail: e.prencipe@fz-juelich.de

Abstract. $\bar{\text{P}}\text{ANDA}$ is a planned experiment at FAIR (Darmstadt) with a cooled antiproton beam in a 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 a spectrometer 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 GEM detector, a straw-tubes central tracker, 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 tool here presented is based on algorithms containing the Kalman Filter equations and a deterministic annealing filter. This general fitting tool (GENFIT2) offers to users also a Runge-Kutta track representation, and interfaces with Millepede II (useful for alignment) and RAVE (vertex finder). It is independent on the detector geometry and the magnetic field map, and written in C++ object-oriented modular code. Several fitting algorithms are available with GENFIT2, with user-adjustable parameters; therefore the tool is of friendly usage. A check on the fit convergence is done by GENFIT2 as well. 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 of GENFIT2 on physics simulations performed for the $\bar{\text{P}}\text{ANDA}$ experiment is shown, with the PandaRoot framework: significant improvement is reported for those channels where a good low momentum tracking is required ($p_{\text{T}} < 400$ MeV/ c).

1. Introduction

The GENFIT2 [1] package is a major upgrade of the generic track reconstruction toolkit GENFIT [2], developed at the Technical University of Munich. It provides an extensible modular open-source framework, performing track fitting and other related tasks. It is available via GitHub.

Tracking is an essential part of physics analyses. All modern 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 be able then to reconstruct with good precision where the particle passed. When tracking is performed inside a magnetic field, it is possible to reconstruct the momentum of a charged particle by determining 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 infers 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 or removing outliers, 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 [3]. 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 $\bar{\text{P}}\text{ANDA}$ experiment [5]. Several experiments right now make use of GENFIT and/or GENFIT2; among these, Belle II, FOPI, SHIP, and BGO-OD.

$\bar{\text{P}}\text{ANDA}$ [5] is a fixed target experiment, with almost full angular coverage, including different kind of gaseous and solid state detectors. Its tracking system involves the presence 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 document 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.

2. Structure of the code

The GENFIT2 package is a C++ object-oriented modular tool. The PandaRoot package `GenfitTools` works as an interface between the GENFIT (or GENFIT2) classes and PandaRoot, and make possible that `Track` objects (meaning GENFIT or GENFIT2 tracks) are transformed in `PndTrack` objects ($\bar{\text{P}}\text{ANDA}$ tracks), and vice versa.

An event in PandaRoot is processed in five main steps:

- **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. The detector simulation is performed by GEANT3 or GEANT4, which can be chosen by the user when running the simulation macro. 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 (`Box Generator`), etc. The output of this process consists of two ROOT files: one containing the simulation parameters, one the main information about MC tracks and points generated in the $\bar{\text{P}}\text{ANDA}$ subdetectors (track number, event number, x - y - z coordinates, x - y - z momentum coordinates, time, deposited energy).
- **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 digis in the $\bar{\text{P}}\text{ANDA}$ subdetectors (fired detector, fired channel, deposited energy, time stamp)
- **Reconstruction.** The reconstruction is subdivided into two parts: the local and the global reconstruction. In the local reconstruction the sub-detector specific information like fired channel numbers or deposited energy in TDC values is translated into global information, like a position in space or an energy in eV. The global reconstruction deals then with the generated hits and combines them into a more global information. This is the step where GENFIT (or GENFIT2) is invoked, and where tracks are fitted using the GENFIT (or GENFIT2) features. Here the information from all hits is combined together to identify the trajectories of charged particles (and calorimeter clusters from electromagnetic showers). 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, χ^2 of the track fit, number of degrees-of-freedom NDoF, fit quality flags).

- Particle Identification (PID). This 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.).
- Analysis. After the four steps mentioned above, the user can run the last step: the analysis. The physics analysis includes event selection, particle candidate combinatorics, particle fits and analysis of the resulting spectra.

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 called **Measurements**, as extensively described 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 **wire** and **spacepoint measurements**, virtual detector planes are built. A track representation consists of a five dimensional state vector and the corresponding covariance matrix, which parametrizes the track. It also provides the possibility to extrapolate the parameters and covariance matrix from one point of the apparatus to another. The GENFIT2 tool implements a track representation employing 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 5 different track representations, one for each mass hypothesis: electron, muon, pion, kaon, and proton. The one that best fits to the measurement is 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 (χ^2 , NDoF, etc.).

To summarize, the GENFIT2 class called **MeasurementFactory** can build a **Track** out of a list of hits (so called **TrackCand** object). 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, based on the GEANT3 code [9]. It uses five parameters, namely q/p , u' , v' , u , v , and their covariance matrix¹. 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,

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 T ; the local coordinates on the plane u and v ; the direction cosines relative to the plane reference frame u' and v' .

either via implementations using the ROOT class `TGeoManager` [10] or via the GEANT4 class `G4Navigator` [11]. 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 [12]-[13]), where the full noise matrix is calculated; the soft Bremsstrahlung energy loss and its straggling for e^- and e^+ . Step sizes for the Runge-Kutta propagation are evaluated inside the `RKTrackRep` class, 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 (GBL) fitter.

Every `TrackPoint` can store several `Measurements`, and the DAF can assign a weight to each of them. 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 sides 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 resonable 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 [14] 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 χ^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, which use a series of measurements observed over time, containing “noise” (random effects) and other inaccuracies, and produce 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. [15], where the linearization is made around the state predictions. Additionally, a square-root

formalism, adapted from Ref. [16], 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 common solution is to linearize around reference states [15]: 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. This operation is performed by the *Kalman filter with reference track* by GENFIT2.

2.5. Deterministic annealing filter

The deterministic annealing filter (DAF) [17] 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.

2.6. Generalized broken lines fitter

The generalized broken lines (GBL) method of track fitting [18] 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 [19], has been implemented in GENFIT2. RAVE is a detector-independent toolkit developed for the CMS experiment [20]. It deals with both finding (pattern recognition) and fitting (statistical estimation) of vertices, taking 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. Its implementation into PandaRoot is now pending.

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 Fig. 1). The new `GenfitTools` structure was created to allow the use of both GENFIT and GENFIT2 as fitting tool, and then to compare results.

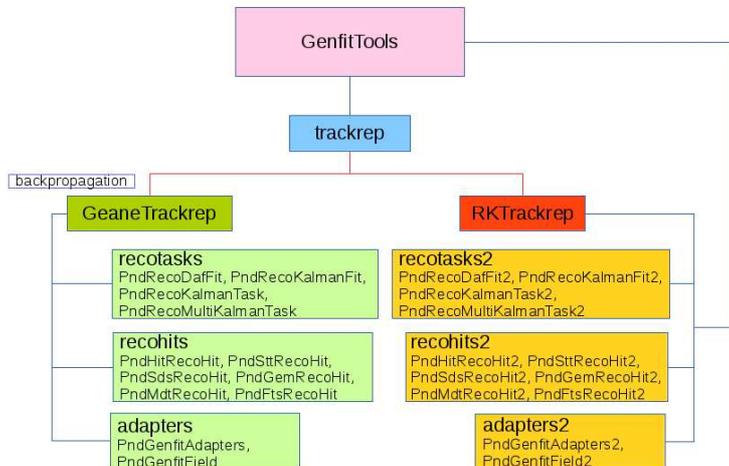


Figure 1. Structure of the **GenfitTools** package in the PandaRoot trunk-revision 28747.

GenfitTools includes several classes:

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

The `adapters`, `recohits`, and `recotasks` refer to the GENFIT interface to the PandaRoot classes, and the `adapters2`, `recohits2`, `recotasks2` refer to the GENFIT2 interface to the PandaRoot classes². The user must provide the track objects in the standard `PndTrack` format, 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 **PANDA MC Box Generator** package. Five particle hypotheses have been tested: electrons, muons, pions, kaons, and protons. We have generated separately single particle events for each type, using different transverse momentum (p_T) values. We tested particles at a fixed polar angle $\theta = 60^\circ$. The magnetic field map used in these tests was a realistic one, as shown in Fig. 2. This is obtained by converting TOSCA [21] calculations into PandaRoot format. Figure 3 shows the efficiency change of a single pion event reconstruction, simulated with the **MC Box Generator** package, when changing 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 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 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.

² The above mentioned `recohits` class takes care of the hit reconstruction in different detectors; in the `trackrep` class the information about track representation is accessible; the `recotasks` class contains the functions needed to reconstruct tracks; `adapters(2)` is the PandaRoot class where the transformation of GENFIT (GENFIT2) track to `PndTrack` is performed, and vice versa.

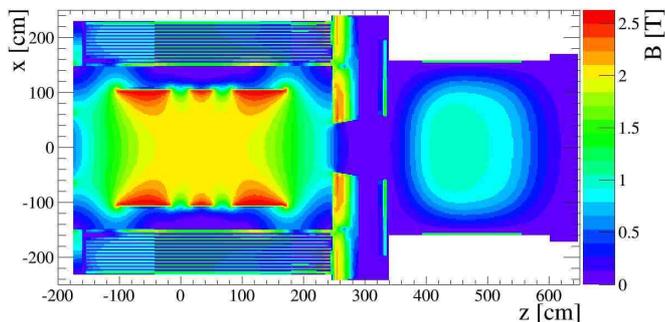


Figure 2. Magnetic field scheme in $\overline{\text{PANDA}}$. In the central part of the detector, the B field is constant and equal to 2T, and in the dipole area the maximum bending power is 2Tm.

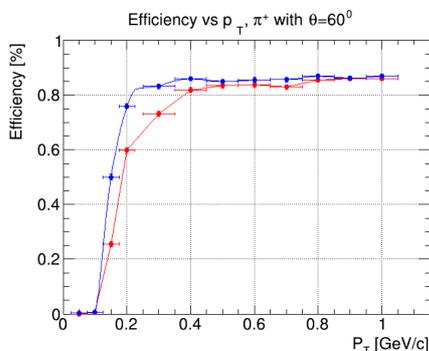


Figure 3. Efficiency versus the transverse momentum of charged pions emitted at $\theta = 60^\circ$. The MC Box Generator package produces single particle events. The blue points show the performance of the GENFIT2 package in the PandaRoot trunk-revision 28747 in comparison with GENFIT (red plot). A clear improvement is observed for $p_T < 350 \text{ MeV}/c$ when using GENFIT2.

A transverse momentum cut larger than $50 \text{ MeV}/c$ is set up: the reason is that at least one 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 $48 \text{ MeV}/c$ for every tracked particle in the $\overline{\text{PANDA}}$ detector.

Figure 4 shows the Gaussian fit to the pull distributions of the track parameters d_0 and z_0 , for $p_T = 1.0 \text{ GeV}/c$. 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 pull is defined as the difference between the generated value, subtracted from the reconstructed value of the variable, then divided by its error. The Gaussian fit is centered at zero and the width is close to the value one, as expected.

4. 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 with electron reconstruction were shown during the workshop in Ref. [23].

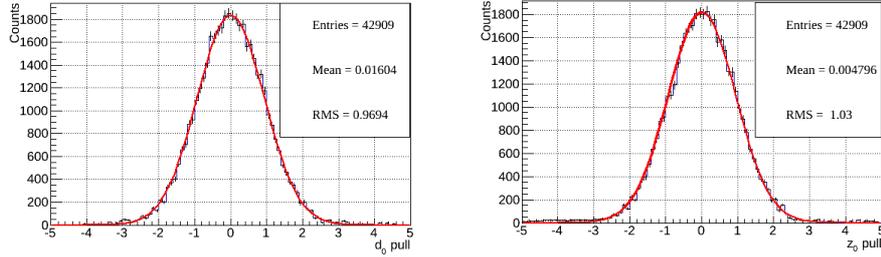


Figure 4. Generated values are subtracted from the reconstructed values of the variables d_0 and z_0 , and then divided by their error. The plots obtained for $p_T = 1.0$ GeV/ c show the pull distributions.

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 $\overline{\text{PANDA}}$ 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.

5. 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], to which most of Section 2 of this report is inspired, for kind clarifications and fruitful discussions, and for their hard work in developing this toolkit.

6. Bibliography

- [1] Rauch J and Schlüter T 2015 *J. Phys.: Conf. Ser.* **608** 012042
- [2] Höppner C, Neubert S, Ketzer B and Paul S 2010 *Nucl. Instr. Meth. A* **620** 518
- [3] Spataro S 2012 *J. Phys.: Conf. Ser.* **396** 022048
- [4] Al-Turany M, Bertini D, Karabowicz R and Uhlig F 2008, *J. Phys.: Conf. Ser.* **119** 032011
- [5] W. Erni *et al* 2009 *Preprint hep-ex/0903.3905*
- [6] Gaponenko I *et al* 2000 *Comp. Phys. Comm.* **140** 406
- [7] Bilka T *et al* 2015 *IEEE Trans. Nucl. Sci.* **62** 1
- [8] Blobel V 2006 *Nucl. Instr. Meth. A* **566** 5
- [9] Brune R *et al* 1993 *CERN Program Library Long Write-up W* **5013** 467
- [10] Brun R and Rademakers F 1996 *AIHENP96 Workshop (Lausanne)* **389** 81
- [11] Agostinelli S *et al* 2003 *Nucl. Instr. Meth. A* **506** 250
- [12] Highland V L 1975 *Nucl. Instr. Meth.* **129**
- [13] Lynch G R and Dahl O I 1991 *Nucl. Instr. Meth. B* **58** 6
- [14] Lista L 2016 *Statistical Methods for Data Analysis in Particle Physics* Springer Publishing, 11 West 42nd Street, New York 10036, USA
- [15] Frühwirth R, Regler M, Bock R K, Grote H and Notz D 2000 *Data Analysis Techniques for High-Energy Physics* Cambridge University Press
- [16] Anderson B D O and Moore J B 1979 *Optimal Filtering* Dover Publications
- [17] Frühwirth R and Strandlie A, *Track Fitting with Ambiguities and Noise: A Study of Elastic Tracking and Nonlinear Filters* 1999 Computer Physics Communications 197
- [18] Kleinwort C 2012 *Nucl. Instr. Meth. A* **673** 107
- [19] Waltenberger W 2011 *IEEE Trans. Nucl. Sci.* **58** 434
- [20] S. Chatrchyan *et al* 2008 *J. Instr.* **3** 361
- [21] The TOSCA Reference Manual 1988 *Vector Fields Limited 068814* 24 Bankside, Kidlington, Oxford, OX51JE, England
- [22] Tadel M 2010 *J. Phys.: Conf. Ser.* **219** 42055
- [23] *Connecting-the-dots-2016 workshop*, <https://indico.hephy.oeaw.ac.at/event/86/overview>; proceedings published by Prencipe E 2016 *Eur. Phys. J. Web Conf* **127** 00013