Track propagation for different detector and magnetic field setups in Acts

Fabian Klimpel on behalf of the ATLAS collaboration
CERN, Espl. des Particules 1, 1211 Meyrin, Room 40-3-D16, Switzerland
E-mail: fklimpel@cern.ch

Abstract. Track finding and fitting are among the most complex parts of event reconstruction in high-energy physics, and usually dominate the computing time in a high luminosity environment. A central part of track reconstruction is the transport of a given track parametrisation (i.e. the parameter estimation and associated covariance matrices) through the detector, respecting the magnetic field setup and the traversed detector material. While track propagation in a sparse environment (e.g. tracking detector with layers) can be sufficiently well approximated by considering discrete interactions at several positions, the propagation in a material dense environment (e.g. calorimeters) is better served by a continuous application of material effects. Recently, a common tracking software project (Acts), originally from the Common Tracking code of the ATLAS experiment, has been developed in order to preserve the algorithmic concepts from the LHC start-up era and prepare them for the high luminosity era of the LHC and beyond. The software is designed in an abstract, detector independent way and prepared to allow highly parallelised execution of all involved software modules, including magnetic field access and alignment conditions. Therefore the propagation algorithm needs to be both flexible and adjustable. The implemented solution using a fourth order Runge-Kutta-Nyström integration and its extension with continuous material integration and eventual time propagation is presented and the navigation through different geometry setups involving different environments is demonstrated.

1. Introduction

The software package Acts [1, 2] (A Common Tracking Software) is a detector-independent, highly parallelisable, continuously tested and documented project of contributors from different collaborations and experiments. It combines ideas, algorithms and techniques from various sources in order to provide a long term maintainable and highly performant standard for tracking software in high energy physics and beyond.

2. Concept of track propagation

Track propagation is a very frequent operation in the track reconstruction and describes the transport of a track parametrisation and the associated covariance matrices through the detector geometry, taking interactions with the material and the magnetic field into account as shown in figure 1. The propagation is essential for many parts of the track reconstruction, from hit finding, where it is used for road building, to track fitting. Particularly for the latter operation, the transport of the associated covariance matrices is necessary.
3. Propagator design and interplay

The propagation of a track parametrisation through a detector geometry requires an interplay of individual semantically different components. An overview is given in figure 2. The managing component is the Propagator which creates the data storage of the track parametrisation which is called State and calls the individual components in a cyclic manner. At each round the trajectory propagates a step further. This State object also guarantees that eventually cached information stays contained within a thread and thus that the software supports fully parallel execution.

The Navigator keeps the detector geometries and predicts upcoming objects. The search for upcoming objects that may be hit by the particle is performed with a straight line approximation of the trajectory as shown in figure 3, and reports the distance to this object. It handles the entering and leaving of objects and provides the properties of the object such that they can be inquired and taken into account for the propagation. Responsible for the mathematical transport of the track parameterisation and covariance matrices is the Stepper class, which can be a straight line stepper, a helical stepper, or a numerical integration based stepper, the latter is the most general case. This part will be explained in more detail in section 4.

The ActionList is a list of user-defined manipulations and actions on the State between the steps. This could involve e.g. the addition of uncertainties onto the covariance matrix by (multiple) scattering at a material surface, or the search for fitting measurements at a sensitive surface and their inclusion in the track parametrisation as performed in the Kalman filter [4] procedure. The AbortList, on the other hand, is a user-defined list of abort conditions of the propagation, e.g. if the particle left the detector.

All these components, the individual Steppers, the Navigator and the Actors and Aborters in the associated lists, can be extended or exchanged with different modules. The compatibility is guaranteed by compile-time checks to reveal a highly virtual interface structure.

4. Track parameter propagation

The position $\vec{r}$ of a particle in vacuum per unit path length $s$ is described by the Lorentz equation

$$\frac{d^2 \vec{r}}{ds^2} = \frac{q}{p} \left( \frac{d\vec{r}}{ds} \times \vec{B}(\vec{r}) \right)$$

(1)

with the electrical charge $q$, the absolute value of the momentum $p$ and the magnetic field $\vec{B}(\vec{r})$. Although this is a simplified problem compared to real track propagation, this equation
Figure 2. Conceptual design of the Propagator and the components that it steers. At the beginning of the propagation, it creates a State object, a data container that is filled by data of the Navigator and the Stepper and passed around all components in a cyclic manner as the track propagates step-wise further. All components are stateless and therefore the moving State provides an explicit thread-local track parameter propagation.

has no general analytical solution for non-constant magnetic fields, the propagation of the track parameters is performed step-wise by the Stepper. An overview of the working principle is shown in figure 4.

The Stepper object performs a numerical integration using the Runge-Kutta-Nyström integration [5] of fourth order for the most general case. Since a numerical calculation is always related to a certain integration error, a step evaluation is accepted if the error is small enough, otherwise the step needs to be adjusted and the evaluation be repeated. This is called adaptive step estimation.

The reconstruction of each track and its origin is done in small steps through the detector. These include calculations of the local magnetic field, material effects and the uncertainties on track parameters. To evaluate the \((n+1)\)th step of the track parameters \(\vec{y}_n\) over a distance \(h\), the Runge-Kutta-Nyström integration of fourth order requires the evaluation of

\[
\vec{y}_{n+1} = \vec{y}_n + \frac{h}{6} \left( \vec{k}_1 + 2\vec{k}_2 + 2\vec{k}_3 + \vec{k}_4 \right)
\]

\[
\vec{y}_{n+1} = \vec{y}_n + h\vec{y}_n + \frac{h^2}{6} \left( \vec{k}_1 + \vec{k}_2 + \vec{k}_3 \right)
\]

which depends on the calculation of \(\vec{k}_i\) [6]. These variables allow the inclusion of any physical effect on the trajectory e.g. by involving the magnetic field evaluated at a certain position. The
Figure 3. Navigation through a detector geometry. It can be approximated as a collection of volumes and contained surfaces. Within the navigation only the next surface and the distance to it needs to be evaluated. The estimation is performed by a straight-line approximation of the particle’s trajectory and updated multiple times.

propagation of the track parameters Jacobian $J_n$ can be expressed via using the transport matrix [7] $D_n$ such that $J_{n+1}$ is given by

$$J_{n+1} = D_n \cdot J_n.$$  

This Jacobian transport is implemented in the STEP algorithm [6, 7]. In Acts it is implemented and performed in the StepperExtensionList. Within this list, a user can define arbitrary effects on the particle’s trajectory and its covariance. For that purpose an Extension is called to calculate $k_i$ and the transport matrix $D_n$. This calculation is thereby not restricted to a single Extension as each could include different effects. In the end a combination of all is delivered to the Stepper, which combines the single components to evaluate the parameters at the $(n+1)$th step.

Since not all definable effects are applicable at each step, e.g. there is no energy loss due to Bethe-Bloch ionisation in vacuum, an Auctioneer concept is introduced. This object allows a step-by-step evaluation of the responsibility of the Extensions. The rules of the auction at each step are that each Extension receives all available data of the track propagation, the State object, and bids upon the data. The Auctioneer decides, based on the collection of all gathered bids, which and how many Extension(s) get a surcharge granted. Since the Extensions are user-defined, their purpose and applications in the context of track propagation cannot be foreseen, i.e. an Extension does not have to calculate a single physical effect but could be a combination of effects and could therewith have an intersect with other Extensions. In order to make the implementation and application as flexible as possible, the Auctioneer is, like the Extensions, user-defined.

5. Summary

The structure and interplay of the components shown in figure 2 provide a setup for the track parameter propagation which enables the search and inclusion of measurements into a parametrisation. Furthermore, the Stepper concept using the StepperExtensionList and the Auctioneer allow the implementation of a single propagation setup that is able to propagate the parametrisation through the whole detector while it can handle all possible physical effects on the trajectory based on the configuration. Therewith a single State object which represents
Figure 4. Concept of an adaptive Runge-Kutta-Nyström integration based step propagation of a track parametrisation. The user definable interface allows the inclusion of further effects on the particle’s trajectory, e.g. by magnetic fields and detector material. An Auctioneer allows additional steering of the application of the individual effects.

the parametrisation of a single particle can be propagated without any additional objects, data container conversions and with a central steerable interface for all objects and functions. A Propagator setup is thereby not fixed to a single detector setup or particle type but can be implemented for an arbitrary experiment. A loosely coupled structure allows a user specific implementation and replacement of components to include all required effects of the particle’s trajectory, a specific detector geometry and magnetic field. The centralised interface and the templated structure of experiment specific components allow a modular customisation of Acts.

Copyright 2019 CERN for the benefit of the ATLAS Collaboration. Reproduction of this article or parts of it is allowed as specified in the CC-BY-4.0 license.

References
[1] Acts code repository, version 0.09.02 https://gitlab.cern.ch/acts
[2] Acts website http://acts.web.cern.ch/ACTS/
[3] Salzburger A 2007 The atlas track extrapolation package (Preprint ATL-SOFT-PUB-2007-005)
[4] Kalman R E 1960 Transaction of the ASME-Journal of Basic Engineering 82D 35–45
[5] Nyström E J 1925 Acta Soc. Sci. Fenn. 50 1–55
[6] Lund E, Bugge L, Gavrilenko I and Strandlie A 2009 Track parameter propagation through the application of a new adaptive runge-kutta-nyström method in the atlas experiment (Preprint ATL-SOFT-PUB-2009-001)
[7] Lund E, Bugge L, Gavrilenko I and Strandlie A 2009 Transport of covariance matrices in the inhomogeneous magnetic field of the atlas experiment by the application of a semi-analytical method (Preprint ATL-SOFT-PUB-2009-002)