AT Developer Primer

September 20, 2016

Abstract

We explain the structure of the AT code, oriented towards someone wanting to implement a new pass method.

1 Overview

The AT is a toolbox of Matlab functions that can tracks electrons through different magnetic elements. One uses Matlab for the various operations one wants to do, but for speed reasons, the underlying tracking engine is written in compiled C. To develop a new element or pass method for an existing pass method, one must thus understand this interface between C and Matlab.

2 Ringpass and Linepass

The two functions used for tracking in AT are **ringpass** and **linepass**. **linepass** takes a lattice structure and electron coordinates and tracks through the structure. **ringpass** does the same, but also takes an input for the number of turns, and tracks through the lattice repeatedly, for this many turns.

ringpass is a Matlab function that accepts the lattice and initial conditions and passes them to `atpass`. `atpass` is a C function with a Mex Function interface to Matlab.

3 atpass

The function `atpass` reads in the lattice and then does the tracking through each element. Each element has a field called 'PassMethod' which specifies the integration routine. This file must exist as a compiled C function. `atpass` calls the pass method via the function 'passFunction' .

```
ExportMode int* passFunction(const mxArray *ElemData, int *FieldNumbers,
                             double *r_in, int num_particles, int mode)
```

4 Structure of a pass method

Each integrator is C code. It has several functions by which it may be accessed. There is a MexFunction, a PassFunction, and a TrackFunction. First we describe the MexFunction.

4.1 MexFunction

References