original

C.G.T.M. #-
V. Whitis
October 1965

TRACK

A Computer Program for Integration of a Particle Track Through an Inhomogeneous, Three-Dimensional Magnetic Field

This paper describes an easy-to-use ALGOL 60 subroutine developed for the integration of the second order differential equation of the motion of a charged particle through a magnetic field, $\bar{B}$. The differential equation has the form

$$\frac{d\bar{u}}{ds} = k\bar{u} \times \bar{B} \quad , \quad k = \frac{\pm 0.03}{p}$$

$$\bar{u} = \frac{d\bar{x}}{ds}$$

where s is the distance along the particle path, $\bar{x} = (x,y,z)$ is the vector distance of the particle from the origin, $\bar{u}$ is the unit vector of directional cosines, directed along the particle velocity, and p is the particle momentum. The code is written as a procedure which, when given a set of x values, will return the corresponding y and z coordinates of a given particle as it moves through an arbitrary magnetic field. The momentum is considered to be constant, but if momentum loss should be considered, then any function $p(s)$ can be inserted.

The integration utilizes a predictor-corrector scheme adapted to second order equations. After a prediction of the $\bar{u}$ and $\bar{x}$ parameter values at the $(i + 1)$ step based on derivatives at the i step, a correction of the $(i + 1)$ values is made by utilizing the predicted and the i parameter values.

These calculations may be expressed by

Predictions

$$\begin{cases} \dfrac{d\bar{u}_i}{ds} = k\bar{u}_i \times \bar{B}_i \\[2ex] \bar{u}'_{i+1} = \bar{u}_i + \dfrac{d\bar{u}_i}{ds}\Delta s \\[2ex] \bar{x}'_{i+1} = \bar{x}_i + (\bar{u}_i + \bar{u}'_{i+1})\,\Delta s/2 \end{cases}$$

Corrections

$$\begin{cases} \bar{u}_{i+1} = \bar{u}_i + \left[\dfrac{d\bar{u}_i}{ds} + \dfrac{d\bar{u}_{i+1}}{ds}(\bar{u}_i,\, \bar{x}_i,\, \bar{u}'_{i+1},\, \bar{x}'_{i+1})\right]\Delta s/2 \\[2ex] \bar{x}_{i+1} = \bar{x}_i + (\bar{u}_i + \bar{u}_{i+1})\,\Delta s/2 \end{cases}$$

A check on the step by step error is made by observing the differences between the predicted and corrected values of the coordinates. If these differences lie outside some fixed range, the step size is adjusted accordingly. Thus, while adequate accuracy is demanded, an unnecessarily small step size is ruled out.

All of the essential input and output quantities are found in the argument list of the procedure. The magnetic field is specified at the discretion of the user in a separate procedure. The calling sequence of TRACK is as follows:

```
TRACK (XS,P,H1,F1,CHQ,YS,ZS,SS,NO);.
```

The input variables are:

XS[0:NO] - A set of coordinate values on the beam axis at which the particle coordinates are desired (including the initial point XS[0]) [meters]

P - The initial particle momentum [Bev/C]

H1 - The initial scattering angle θ [radians]

F1 - The initial orientation angle φ [radians]

CHQ - The particle charge (ordinarily +1. or -1.)

YS[0] - The initial y coordinate [meters]

ZS[0] - The initial z coordinate [meters]

NO - The number of coordinate points desired as output

The following output results:

YS[1:NO] - The y-coordinates corresponding to XS[1:NO] [meters]

ZS[1:NO] - The z-coordinates corresponding to XS[1:NO] [meters]

SS[0:NO] - The arc length along the track measured from SS[0] = 0. [meters]

NO - The number of coordinate points actually computed (may be different from the input number)

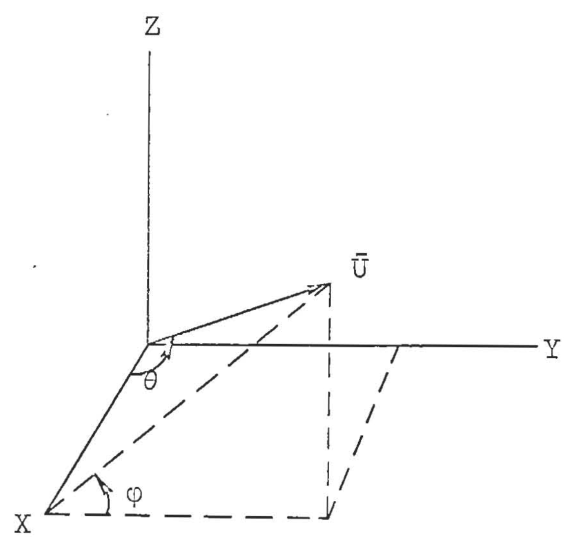

Fig. 1

Coordinate System

The coordinate system used is a right handed one which uses two angles θ and φ to define the initial direction of the particle motion, the vector $\bar{u}$. θ is the scattering angle measured from the principal axis, X, to the vector, $\bar{U}$. φ is the orientation angle measured positively from the XY plane to the line dropped perpendicularly to the X axis from $\bar{u}$ (the azimuthal angle about the x axis).

The procedure is limited to monotonically increasing input values of x . This means that XS[V] must increase with V . The integration procedure is quite capable of integrating backwards in x ; however, the user presumably does not know at what point x begins to decrease and thus cannot give a proper sequence of double valued x's for input. If the computed x begins to decrease ($\theta > 90^\circ$) before attaining the final value XS[NO], then NO is reset to the last value attained by the XS index. An error message is then printed: "THE TRACK HAS SPIRALED BEFORE X = XS[V]". Now NO ← V - 1.

The magnetic field is specified by a procedure FIELD (X, Y, Z, BX, BY, BZ) which must be declared by the user for the execution of TRACK. The structure of FIELD is straightforward:

```
FIELD (X,Y,Z,BX,BY,BZ) ;
REAL X,Y,Z,BX,BY,BZ     ;
BEGIN
     BX ← f(X,Y,Z)  ;
     BY ← g(X,Y,Z)  ;
     BZ ← h(X,Y,Z)          END ; .
```

The BX, BY, BZ are, of course, the x, y, and z components of the magnetic field expressed as functions of the coordinates. For example, a constant field orthogonal to the X-Y plane could be written BX $\leftarrow$ 0.; BY $\leftarrow$ 0.; BZ $\leftarrow$ CONST;. The field components could also be extracted from tables if the user chose to write the appropriate FIELD procedure.

In a test run of TRACK with a particle exhibiting a radius of curvature of one meter in a uniform field, a comparison was made with the exact helix solution. The error in the y component of position was less than 0.001% of the arc length at a quarter circle of arc. This low error was obtained with a step size greater than 0.5 cm. Even lower error tolerances may be imposed by reducing the predictor-corrector error allowed in the program. The error is continually monitored and the step size adjusted accordingly to maintain error tolerance with a minimum consumption of computer time.

TRACK will integrate a particle track through $\pi/2$ radians of arc in one or two seconds.