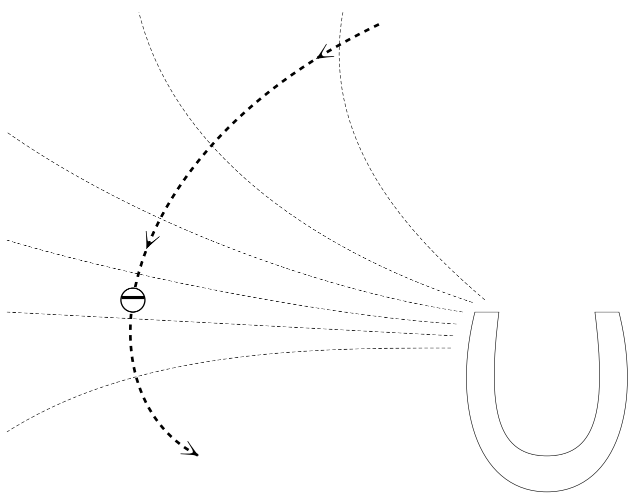
MISN-0-128

THE NUMEROV ALGORITHM FOR MAGNETIC FIELD TRAJECTORIES



Project PHYSNET Physics Bldg. Michigan State University East Lansing, MI

THE NUMEROV ALGORITHM FOR

MAGNETIC FIELD TRAJECTORIES

by

Peter Signell

Title: **The Numerov Algorithm for Magnetic Field Trajectories**

Author: Peter Signell, Michigan State University

Version: 2/1/2000 Evaluation: Stage 0

Length: 1 hr; 8 pages

Input Skills:

1. Use the Taylor series to expand a function about a point (MISN-0-4).

Output Skills (Knowledge):

K1. Derive the recurrence relation for the Numerov Algorithm, to second order and in two dimensions, in a form suitable for use in obtaining the trajectory of a charged particle in an arbitrary magnetic field. Show all steps in the derivation.

K2. Derive equations for insertion of initial position and velocity in the Numerov Algorithm and communicate a method of obtaining a particular desired accuracy.

Post-Options:

1. "Trajectory of a Charged Particle in a Magnetic Field: Cyclotron Orbits (a computer project)" (MISN-0-127).

THIS IS A DEVELOPMENTAL-STAGE PUBLICATION
OF PROJECT PHYSNET

The goal of our project is to assist a network of educators and scientists in transferring physics from one person to another. We support manuscript processing and distribution, along with communication and information systems. We also work with employers to identify basic scientific skills as well as physics topics that are needed in science and technology. A number of our publications are aimed at assisting users in acquiring such skills.

Our publications are designed: (i) to be updated quickly in response to field tests and new scientific developments; (ii) to be used in both classroom and professional settings; (iii) to show the prerequisite dependencies existing among the various chunks of physics knowledge and skill, as a guide both to mental organization and to use of the materials; and (iv) to be adapted quickly to specific user needs ranging from single-skill instruction to complete custom textbooks.

New authors, reviewers and field testers are welcome.

THE NUMEROV ALGORITHM FOR MAGNETIC FIELD TRAJECTORIES

by

Peter Signell

1. Introduction and Description

Many problems in science and engineering can not be solved in terms of known functions, even when the under lying equation is known. Such a problem is the trajectory of a charged particle in a non-uniform magnetic field. For such cases one must resort to general numerical techniques: one of the most common is examined in this module.

2. Study Material

The force on a charged particle in a magnetic field is the Lorentz force:

$$\vec{F} = q\vec{v} \times \vec{B} \tag{1}$$

where q is the charge of the particle, $\vec{v}$ is its velocity, and $\vec{B}$ is the value of the magnetic field at the present location of the particle. The present force on the particle is $\vec{F}$. The force influences the particle's trajectory through Newton's Second Law:

$$\vec{F} = m\vec{a} \tag{2}$$

where m is the particle's mass.

In our case $\vec{B}$ will always be at right angles to $\vec{v}$, as is obvious from Eq. (1), hence $\vec{v}$ can change $\vec{B}$'s direction but not its magnitude. We will restrict ourselves to motion in the x-y plane by putting $\vec{v}$ there initially and putting $\vec{B}$ in the z-direction:

$$\vec{B} = B(x,y)\hat{z}$$

$$\vec{v} = x'\hat{x} + y'\hat{y} \tag{3}$$

where a prime denotes derivative with respect to time: $x' \equiv dx/dt$.

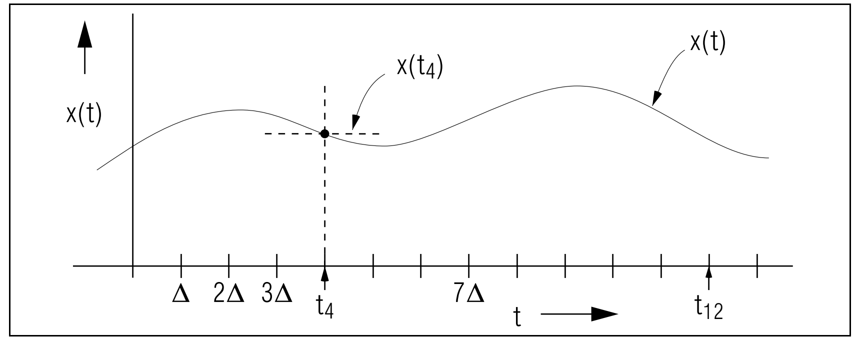


Figure 1. A function $x(t)$, specified at equally spaced values of t.

Equating the forces in Eqs. (1) and (2) and taking components (you do it) we get:

$$\begin{aligned} mx'' &= qy'B \\ my'' &= -qx'B \\ mz'' &= 0 \end{aligned} \tag{4}$$

Forget the third (z) equation since its solution does not couple to those of the x- and y-equations. Note that the x- and y-equations are "coupled", in that x'' involves y' and y'' involves x'. If we define $a(x,y) \equiv (q/m)B(x,y)$ then Eqs. (4) can be written:

$$x'' = ay'$$

$$y'' = -ax'$$

or, equivalently,

$$v_x'(t) = a(t)v_y(t); v_y'(t) = -a(t)v_x(t)\,. \tag{5}$$

In the Numerov method we deal with the solution functions $x(t)$, $y(t)$, $v_x(t)$, and $v_y(t)$, as a series of numbers at "net-point" times that are integrally spaced:

$$t_n = n\Delta\,.$$

This is illustrated in Fig. 1 for $x(t)$.

We then write:

$$x_n \equiv x(t_n) \equiv x(n\Delta)$$
$$y_n \equiv y(t_n) \equiv y(n\Delta)$$
$$v_{x,n} \equiv v_x(t_n) \equiv v_x(n\Delta)$$
$$v_{y,n} \equiv v_y(t_n) \equiv v_y(n\Delta)$$

and our Eqs. (3) and (5) become:

$$\begin{aligned} x'_n &= v_{x,n} \\ y'_n &= v_{y,n} \\ v'_{x,n} &= a_n v_{y,n} \\ v'_{y,n} &= -a_n v_{x,n}\,. \end{aligned} \tag{6}$$

These are four coupled equations.

We now connect the consecutive values of the x's and v's by making Taylor's Series expansions of each of them. For example:

$$x(t+\Delta) = x(t) + \Delta x'(t) + \frac{\Delta^2}{2!}x''(t) + \ldots$$

We will choose a sufficiently small so that terms beyond the second will be negligible compared to the first two terms. Then in our net-point notation and using Eq. (6):

$$\begin{aligned} x_{n+1} &= x_n + \Delta v_{x,n} \\ v_{x,n+1} &= v_{x,n} + \Delta a_n v_{y,n} \end{aligned} \tag{7}$$

▷ You derive the equations for y_{n+1} and $v_{y,n+1}$.

Given the $t = 0$ position and velocity components,

$$x_0;\ y_0;\ v_{x,0};\ v_{y,0}$$

we can use the four "recurrence" relations (7) to generate the four position and velocity components at time $t = \Delta$:

$$x_1;\ y_1;\ v_{x,1};\ v_{y,1}\,.$$

Putting the latter back into the right hand side of the recurrence relations, we get the values at time $t = 2\Delta$. Continuing this process, we can find

the trajectory as far into the future as we wish. We must only be careful to put the correct value of $\vec{B}$ into a at each space-point (x_n,y_n).

Finally, how does one know what size time interval Δ to use? One could attempt to assess the importance of successive terms in the Taylor's Series, but a more reliable method is to decrease Δ until the predicted trajectory stabilizes; that is, until it does not change significantly when Δ is made even smaller. However, one must be aware that if Δ is continually made even smaller, a point will be reached where the errors will start increasing due to the computer's finite-word-size limit.

The algorithm, then, consists of:

1. recurrence relations
2. method of assuring desired accuracy
3. insertion of initial conditions.

Acknowledgments

Preparation of this module was supported in part by the National Science Foundation, Division of Science Education Development and Research, through Grant #SED 74-20088 to Michigan State University.