Ballistic Research Laboratories

TECHNICAL NOTE NO. 736

[redacted]

SEPTEMBER 1952

The Iterated Simpson Method of Numerical Integration

W. BARKLEY FRITZ

The Ballistic Research Laboratories Technical Note is designed for use within the laboratories or for issuing available information, when the occasion demands speed.

The contents of this paper are of the nature of advance information and may be extended or otherwise revised.

ABERDEEN PROVING GROUND, MARYLAND

BALLISTIC RESEARCH LABORATORIES

TECHNICAL NOTE NO. 736

September 1952

THE ITERATED SIMPSON METHOD OF NUMERICAL INTEGRATION

W. Barkley Fritz

Project No. TB3-0007K of the Research and Development Division, Ordnance Corps

ABERDEEN PROVING GROUND, MARYLAND

BALLISTIC RESEARCH LABORATORIES

TECHNICAL NOTE NO. 736

WBFritz/bts
Aberdeen Proving Ground, Md.
September 1952

THE ITERATED SIMPSON METHOD OF NUMERICAL INTEGRATION

ABSTRACT

The paper reviews the well known Iterated Simpson method for obtaining the numerical solution of the initial value problem for a general system of n first order ordinary differential equations. The formulae are developed and two examples are given illustrating their use.

This page was intentionally left blank

I. INTRODUCTION

Early in 1950 a new numerical procedure for solving systems of ordinary differential equations on computing machines was announced by R. F. Clippinger and B. Dimsdale of the Computing Laboratory. The method was first discussed in the Coding Notes prepared for the Coding Course given by the Ballistic Institute (September 1949 to February 1950). The procedure is also mentioned as Clippinger's Method in the lecture notes of L. H. Thomas. However, since neither set of notes is available for general distribution it was felt desirable to publish this note.

This method, which has been called "Iterated Simpson", is of the fourth order, that is the error (excluding round-off) goes to zero as the fifth power of the step size. It is thus equivalent to approximating the solutions locally (on intervals of length 2h) by polynomials of the fourth degree. These polynomials differ from the Taylor expansions of the dependent variables about the left endpoints of the interval of integration by terms of order h^5 only, if the right members of the differential equations possess a sufficient number of bounded derivatives.

Like the Runge-Kutta method, it proceeds from initial data at one point without need for data at other points. This is a distinct advantage for large scale digital computers since it limits the storage required for the dependent variables, and simplifies changes in the grid size when required. If the functions possess discontinuities, they may be placed at the grid points and the method may still be used.

Another advantage is the smaller amount of computer memory required to solve a system of given order, when compared with other methods of comparable accuracy. It has been claimed that systems of roughly 60% higher order can be solved with this method than with the widely used methods of Moulton, Milne, Adams, etc. However, the method of Runge-Kutta can be shown to be as efficient as Iterated Simpson with respect to storage requirements, for instance, Proceedings of the Cambridge Philosophical Society, Vol. 47 Part 1 1950, "A Process for the Step by Step Integration of Differential Equations in an Automatic Digital Computing Machine". - S. Gill, pp 96-109. Further study will indicate whether the method of Runge-Kutta as used by S. Gill should replace Iterated Simpson entirely.

Further it should be emphasized that the usefulness of the Iterated Simpson method is restricted largely to high speed digital computers, since considerable computation time is sacrificed in order to gain these advantages. However, for a fast machine with limited storage this sacrifice may well be compensated for by the smaller storage required. A distinct advantage over Runge-Kutta is that the iteration makes a single step practically self checking.

II. PROCEDURE

Let the system of n first order ordinary differential equations be

$$(1) \qquad y_p' = f_p(x, y_1, y_2, \ldots, y_n) \qquad (p = 1, 2, \ldots, n)$$

which can be written in vector form as $y' = f(x,y)$ where y, y' and f represent vectors. The initial conditions are

$$(2) \qquad y(x_o) = y_{i-1} \text{ at } x = x_o$$

We derive formulae for y at a point x which is a distance $2h$ from x_o by expanding $y(x) = y_i$, $y'(x) = y_i'$, $y(x_o) = y_{i-1}$, and $y'(x_o) = y_{i-1}'$ about $\bar{x} = x_{i-1} + h$ and eliminating $y = y(\bar{x})$, $\bar{y}'$, $\bar{y}''$, $\bar{y}'''$, and $\bar{y}^{iv}$ from the expansions:

$$(3.1) \quad y_i = \bar{y} + h\bar{y}' + (h^2/2)\,\bar{y}'' + (h^3/6)\,\bar{y}''' + (h^4/24)\,\bar{y}^{iv} + O(h^5)$$

$$(3.2) \quad y_{i-1} = \bar{y} - h\bar{y}' + (h^2/2)\,\bar{y}'' - (h^3/6)\,\bar{y}''' + (h^4/24)\,\bar{y}^{iv} + (h^5)$$

$$(3.3) \quad y_i' = \bar{y}' + h\bar{y}'' + (h^2/2)\,\bar{y}''' + (h^3/6)\,\bar{y}^{iv} + O(h^4)$$

$$(3.4) \quad y_{i-1}' = \bar{y}' - h\bar{y}'' + (h^2/2)\,\bar{y}''' - (h^3/6)\,\bar{y}^{iv} + O(h^4),$$

where $O(h^m)/h^m$ is a function of h which remains bounded as h → 0.

Adding (3.1) and (3.2) we get

$$(4.1) \quad y_i + y_{i-1} = 2\bar{y} + h^2\bar{y}'' + O(h^4)$$

Subtracting (3.2) from (3.1) we get

$$(4.2) \quad y_i - y_{i-1} = 2h\bar{y}' + h^3/3\;\bar{y}''' + O(h^5)$$

Adding (3.3) and (3.4) we get

$$(4.3) \quad y_i' + y_{i-1}' = 2\bar{y}' + h^2\bar{y}''' + O(h^4)$$

Subtracting (3.4) from (3.3) we get

$$(4.4) \quad y_i' - y_{i-1}' = 2h\bar{y}'' + h^3/3\;\bar{y}^{iv} + O(h^5).$$

Next we eliminate $\bar{y}'''$ by multiplying (4.3) by $-h/3$ and adding it to (4.2); Collecting terms we get

(5.1) $y_i = y_{i-1} + \frac{h}{3} (y'_{i-1} + 4\bar{\bar{y}}' + y'_i) + O(h^5)$

Equation (5.1) is merely Simpson's Rule which is known to be correct to fourth order.

Likewise we eliminate $\bar{\bar{y}}''$ by multiplying (4.1) by 2 and adding to - h times (4.4). Divide this relation through by 4 and we get

(5.2) $\bar{\bar{y}} = y_i /2 + y_{i-1} /2 + h/4 (y'_{i-1} - y'_i) + O(h^4)$

The numerical procedure for determining y(x) consists in taking, as a first approximation by Euler,

(6) $y_{io} = y_{i-1} + 2hy'_{i-1}, \quad y'_{i-1} = f(x_{i-1}, y_{i-1}),$

and defining the (j + 1)st approximation to y_i by the formulae:

(7.1) $\bar{\bar{y}}_j = y_{ij} /2 + y_{i-1} /2 + h/4 (y'_{i-1} - y'_{ij})$ and

(7.2) $y_{i,j+1} = y_{i-1} + h/3 (y'_{i-1} + 4\bar{\bar{y}}'_j + y'_{ij})$.

All y' in (7.1) and (7.2) are defined by $y' = f(x,y)$, in particular $\bar{\bar{y}}'_j = f(\bar{\bar{x}}, \bar{\bar{y}}_j)$.

Before listing a few examples we summarize how the method is used:

a. First use Euler integration to get an initial approximation to the values of the dependent variables at the new point.

b. Next use these values and the values at the initial point to obtain approximations for the values of the dependent variables at the point half way between the new point and the old point.

c. Use these intermediate values and the values at the old point to obtain a better approximation to the new point.

d. Steps b. and c. may be repeated as many times as desired or until the changes in the dependent variables are sufficiently small.

In the examples considered in the next section, two iterations were found to be sufficient to obtain the desired accuracy. In other examples more than two iterations were required. It should be noted that the ordering of the steps is such that only the last computed values at the new point or the values at the intermediate point need be kept in storage in addition to the values at the old point.

III. EXAMPLES

We illustrate the procedure by means of two examples; both were coded for and run on the Eniac.

a. Bessel Functions of Order zero and one.

The method was first applied to the solution of the equations (8.1) $y' = -z$ and (8.2) $z' = y - z/x$ with initial conditions: (8.3) $y(0) = 1$, $z(0) = 0$. The solution of this system is

$y = J_o(x)$ (zero order Bessel function)

$z = J_1(x)$ (first order Bessel function)

It was found possible to obtain nine correct significant figures, comparing the results with the Harvard Tables. The error due to round-off was only a maximum of 5 in the 10th figure after 500 steps of integration (an estimated 30,000 multiplications and divisions). This is only 10 times the error involved in introducing a number into the Eniac.

The algorithm for Iterated Simpson applied to the problem (8.1) - (8.3) and used on the Eniac follows:

As before we used (6) for the initial approximation: (or for the two equations we write)

(8.4) $y_{io} = y_{i-1} + 2hy'_{i-1}$

(8.5) $z_{io} = x_{i-1} + 2hz'_{i-1}$

The Iterated Simpson relations are

(8.6) $2/3\ \bar{\bar{y}}_{ij} = y_{ij}/3 + y_{i-1}/3 + h/6\ (z_{ij} - z_{i-1})$

(8.7) $2/3\ \bar{\bar{z}}_{ij} = z_{ij}/3 + x_{i-1}/3 + h/6 \left[x_{i-1} - (y_{ij} - z_{ij}/x)\right]$

(8.8) $y_{i,j+1} = y_{i-1} - h/3\ (z_{i-1} + 4\bar{\bar{z}}_{ij} + z_{ij})$

(8.9) $z_{i,j+1} = z_{i-1} + h/3 \left[z'_{i-1} + 4\ (\bar{\bar{y}}_{ij} - \bar{\bar{z}}_{ij}/x) + y_{ij} - z_{ij}/x\right]$

$2/3\ \bar{\bar{y}}_{ij}$ is computed rather than $\bar{\bar{y}}_{ij}$ to aid in handling number size and to simplify the number of operations required.

b. Integration of the trajectory equations.

The method has been applied to the integration of the particle theory trajectory equations with t as the independent variable. The algorithm follows:

For the initial approximation we write

$$y'_{io} = y'_{i-1} - (E_{i-1}y'_{i-1} + g)\,2h \tag{9.1}$$

$$x'_{io} = x'_{i-1} - (E_{i-1}x'_{i-1})\,2h \tag{9.2}$$

$$y_{io} = y_{i-1} + 2hy'_{i-1} \tag{9.3}$$

$$x_{io} = x_{i-1} + 2hx'_{i-1}. \tag{9.4}$$

The Iterated Simpson relations are as follows:

$$\bar{\bar{y}}'_{ij} = 1/2(y'_{i-1} + y'_{ij}) + h/4\,(E_{ij}y'_{ij} - E_{i-1}y'_{i-1}) \tag{9.5}$$

$$\bar{\bar{x}}'_{ij} = 1/2(x'_{i-1} + x'_{ij}) + h/4\,(E_{ij}x'_{ij} - E_{i-1}x'_{i-1}) \tag{9.6}$$

$$\bar{\bar{y}}_{ij} = 1/2(y_{i-1} + y_{ij}) + h/4\,(y'_{i-1} - y'_{ij}) \tag{9.7}$$

$$\bar{\bar{x}}_{ij} = 1/2(x_{i-1} + x_{ij}) + h/4\,(x'_{i-1} - x'_{ij}) \tag{9.8}$$

$$y'_{i,j+1} = y'_{i-1} - h/3\,(E_{i-1}y'_{i-1} + 4\bar{\bar{E}}_{ij}\bar{\bar{y}}'_{ij} + E_{ij}y'_{ij} + 6g) \tag{9.9}$$

$$y_{i,j+1} = y_{i-1} + h/3\,(y'_{i-1} + 4\bar{\bar{y}}'_{ij} + y'_{ij}) \tag{9.10}$$

$$x'_{i,j+1} = x'_{i-1} - h/3\,(E_{i-1}x'_{i-1} + 4\bar{\bar{E}}_{ij}\bar{\bar{x}}'_{ij} + E_{ij}x'_{ij}) \tag{9.11}$$

$$x_{i,j+1} = x_{i-1} + h/3\,(x'_{i-1} + 4\bar{\bar{x}}'_{ij} + x'_{ij}) \tag{9.12}$$

A discussion on bounds on errors is contained in the original Coding Notes of Clippinger and Dimsdale, as well as in the Lecture Notes on "Ordinary Differential Equations: June - August 1950", prepared by Mark Lotkin, which are available only for local BRL distribution.

W. Barkley Fritz
W. Barkley Fritz

DISTRIBUTION

No. of Copies	Organization
2	Chief of Ordnance Department of the Army Washington 25, D. C. Attn: ORDTB - Bal Sec
5	British - ORDTB for distribution
2	Canadian Joint Staff - ORDTB for distribution
2	Chief, Bureau of Ordnance Department of the Navy Washington 25, D. C. Attn: Re3
2	Commander Naval Proving Ground Dahlgren, Virginia
2	Chief of Naval Research Technical Information Div. Library of Congress Washington 25, D. C.
2	Commander Naval Ordnance Laboratory White Oak Silver Spring, Maryland
2	Commander Naval Ordnance Test Station Inyokern P. O. China Lake, California Attn: Technical Library & Editorial Section
1	Superintendent Naval Postgraduate School Monterey, California
1	Director Air University Library Maxwell Air Force Base Alabama
5	Director Armed Services Technical Information Agency Documents Service Center U. B. Building Dayton 2, Ohio Attn: DSC - SA

UNCLASSIFIED

ATI 171 002 (Copies obtainable from ASTIA-DSC)

Aberdeen Proving Ground, Ballistic Research Lab., Md.
(Technical Note No. 736)

The Iterated Simpson Method of Numerical Integration - Project No. TB3-0007K

Fritz, W. Barkley Sept'52 9pp.

Computers - Problem coding	Engineering Practices (49)
Integration	Computers (4)
Iterative processes	

UNCLASSIFIED

○

050750