AD NUMBER
AD812362
NEW LIMITATION CHANGE
TO Approved for public release, distribution unlimited
FROM Distribution authorized to U.S. Gov't. agencies and their contractors; Administrative/Operational Use; Feb 1967. Other requests shall be referred to US Army Ballistic Research Lab., Aberdeen Proving Ground, MD 21005-5066.
AUTHORITY
US Army Armament Research & Develpment Command ltr dtd 20 Feb 1981

BRL R 1358

8123362

AD

BRL

REPORT NO. 1358

AN EFFICIENCY STUDY OF SEVERAL TECHNIQUES FOR THE NUMERICAL INTEGRATION OF THE EQUATIONS OF MOTION FOR MISSILES AND SHELL

by

Harold J. Breaux

February 1967

This document is subject to special export controls and each transmittal to foreign governments or foreign nationals may be made only with prior approval of Commanding Officer, U.S. Army Ballistic Research Laboratories, Aberdeen Proving Ground, Maryland

U. S. ARMY MATERIEL COMMAND
BALLISTIC RESEARCH LABORATORIES
ABERDEEN PROVING GROUND, MARYLAND

BALLISTIC RESEARCH LABORATORIES

REPORT NO. 1358

FEBRUARY 1967

AN EFFICIENCY STUDY OF SEVERAL TECHNIQUES FOR THE NUMERICAL INTEGRATION OF THE EQUATIONS OF MOTION FOR MISSILES AND SHELL

Harold J. Breaux

Computing Laboratory

RDT & E Project No. 1P523801A287

ABERDEEN PROVING GROUND, MARYLAND

BALLISTIC RESEARCH LABORATORIES

REPORT NO. 1358

HJBreaux/bg
Aberdeen Proving Ground, Md
February 1967

AN EFFICIENCY STUDY OF SEVERAL TECHNIQUES FOR THE NUMERICAL INTEGRATION OF THE EQUATIONS OF MOTION FOR MISSILES AND SHELL

ABSTRACT

The numerical integration of the equations of motion for missiles and shell is a frequently occurring computational problem in government laboratories and in the aerospace industry. Because of the repetitive nature of such computations and the continuing requirement, it is desirable to make these computations as efficiently as possible in order to minimize the computer time spent in their solution. The cost in terms of time required to solve representative problems to a specified accuracy is a measure of the relative efficiency of numerical integration methods and can be determined only by experimentation. This report describes the results of such an experimental study and finds that the Kutta-Merson procedure with automatic and continuous interval adjustment is far superior to the predictor-corrector techniques.

TABLE OF CONTENTS

INTRODUCTION

Prior to the study culminating in this report the Ballistic Research Laboratories Electronic Scientific Computer (BRLESC) had available in its compiler two subroutines for the numerical integration of a system of ordinary differential equations. These subroutines were RKG and RKGMA. The RKG routine is the Runge-Kutta-Gill procedure as described by Romanelli[1]*. RKGMA is a procedure using the Adams fifth order predictor formula with no subsequent correction. The procedure uses the RKG routine as a starter, hence its name, Runge-Kutta-Gill-Modified-Adams**. The reason for using only the predictor cycle in RKGMA as opposed to the classical predictor-corrector methods was that the RKGMA method requires only one derivative evaluation per step and hence runs twice as fast as the conventional method when both methods use the same step size. This allows a step size of h/2 for the same cost as the conventional method using step size h and results in a technique with greater accuracy. This observation was "verified" in a study performed in the Computing Laboratory and led to the incorporation of RKGMA into the BRLESC compiler. Of course all this neglects the very important consideration of stability and in fact Hamming[3] shows that for a simple predictor formula, such an application can be unstable.

The use of RKGMA in the integration of trajectories nevertheless produced a very accurate method and was initially thought to be more

---

*Superscript numbers denote references which may be found on page 37.

**Hildebrand[2] refers to the Adams fifth order predictor-corrector method with iteration as the modified Adam's method. To avoid possible confusion this distinction is noted.

economical than the RKG method. This conclusion was derived as follows: In the absence of any practical estimate of the truncation error in the RKG scheme, it was traditionally used with a constant step size which from experience one knew was adequate and "safe". When RKGMA became available, it was natural to attempt to solve trajectory problems with the same step size traditionally used with the RKG routine. The result was that the method was stable for the particular applications and yielded a method almost four times more economical than RKG. Such a comparison is not complete, however, in that one should also consider accuracy and stability. Subsequent experiments designed to consider these factors led to the conclusion that for trajectory computations RKG was more efficient than RKGMA. These results were attributed to the fact that to maintain stability with RKGMA one had to use a smaller step size which more than offset its four to one advantage with regard to derivative evaluations.

A method for improving the stability properties of a procedure such as RKGMA arose from the findings of Hull and Creemer[4] who reported on a study of the efficiency of predictor-corrector procedures. They found that a method with only one evaluation of the derivative consistently seemed the most efficient type of predictor-corrector technique. This technique maintained the advantage of RKGMA with respect to derivative evaluations, but by modifying the cycling process improved the stability properties. This scheme allows one to apply both the predictor and corrector for the price of one evaluation or the application of the predictor and then the corrector step twice for the cost of two evaluations. In addition, this procedure even with only one evaluation provides an estimate of the truncation error. A study of this scheme, applied to the Adams formulas became the basis of the BRL study. An additional objective of this investigation was to study the formulas of Crane and Klopfenstein[5], who derived a predictor formula which when used with the Adams fifth order corrector yielded a method with increased range of stability. During the study the Kutta-Merson algorithm came to the attention of the author and was added to the study because of its favorable stability properties and because it provides a computable estimate of the truncation error.

The studies described herein were by no means exhaustive; i. e., only fifth order methods were examined, but seem sufficiently important to warrant reporting since few such studies on practical dynamic problems are found in the literature.

The trajectory models used in the study were actual models currently being used by the Ballistic Research Laboratories in computing Firing Tables for various weapon systems. Although the results of this study may to some extent be affected by programming pecularities and the use of a particular computer it is felt that the results are fairly representative of what might be expected in general.

General Discussion

Many different schemes exist for the numerical solution of ordinary differential equations. Because of the widespread need for numerical integration, most computing facilities have as a standard routine, some method of numerical integration which can be utilized as a package by programmers. This leads to a more efficient operation and avoids the occurrence of redundant programming effort. Because of the general use of such a routine or package, it is highly desirable that it be flexible and efficient. The flexibility is measured by the relative ease with which programmers can call for the routine and by the number of tasks to which the routine can be applied with confidence. The efficiency is best measured by the computing time required to solve representative problems to a given degree of accuracy.

The most efficient routines seem to be those which provide an estimate of the truncation error and which adjust the integration interval as the integration proceeds. When a provision for interval adjustment is lacking, the programmer is forced to be conservative in his choice of step size and may even have to solve the same problem several times with different intervals to assure himself that the procedure is stable for the particular problem and the chosen interval.

The truncation error is a measure of the error incurred at a given step and cannot be used to determine precisely the accumlated error resulting after the integration has proceeded for many steps. Despite this, the use of truncation error as a guide to choosing the integration interval has proven quite successful in creating efficient routines.

The computation of trajectories is a frequently occurring problem which requires numerical integration. The Firing Tables Branch of the Computing Laboratory, for example, in 1965 used over 900 hours of machine time, a significant fraction of which was used on the computation of trajectories. These computations usually allow varying intervals as the problem proceeds to attain a fixed truncation error. For this reason a process which can estimate the truncation error and adjust the step size continuously and automatically can be much more efficient than a procedure which does not. The design of such a procedure for general trajectory computations was the major objective of this study. An additional objective was the creation of a standard subroutine which could be used by programmers in general on a variety of problems. Among the integration procedures which have the most widespread use are the two basic categories:

(1) Runge Kutta methods

(2) Predictor-corrector methods

The general features which are discussed extensively in the literature pertain mostly to the following six areas:

(1) Storage requirement

(2) Accuracy

(3) Stability

(4) Ease of starting

(5) Number of derivative evaluations per step

(6) Estimation of truncation error

The Runge Kutta method or the Gill modification evaluated against these features has become almost universally adopted as the best procedure to have as a general purpose standard package. The method is very accurate, has probably the best stability properties of the methods used extensively, is self starting, and the Gill modification of the method requires very little storage of data. The method suffers from the disadvantage of requiring four derivative evaluations to advance the computation one mesh interval. As generally used the method also suffers from the additional disadvantage of not providing an estimate of the truncation error unless used in a multistep mode[1]; i.e., the same step is done twice, once with an interval of h and then followed by two steps of h/2. This method has not found widespread use mainly due to the increased cost of the multistep application. The computable estimates of truncation error found by Merson[6] and more recently by Scraton[7] seem to largely eliminate this problem however.

The predictor-corrector methods are generally considered to be the most efficient since in the conventional application only two derivative evaluations are required and because of the ready availability of a computable estimate of the truncation error. While this may be generally true, there exists a great deal of evidence (this study for example) indicating that for some classes of problems the Runge-Kutta methods are the most economical if used with a technique for interval adjustment.

Despite the extensive body of literature that exists on the subject, no definitive conclusions can be made when attempting to choose a method. Those studies that are found in the literature usually are done for prototype differential equations and it is not generally valid to extend the findings from such studies to the more general dynamic problems that are of greatest interest. Many problems are such however that the most expeditious solution is to use a method available at one's computing facility; i.e., even though the particular method may be very inefficient for a given problem, the fact that it has already been programmed and checked out offsets other considerations. The problem may be a "one-shot" type which can be solved for several integration intervals without incurring much total cost in computation time. The trajectory problem is one, however, which is repetitive, i.e., after a program is written it may be used to compute many trajectories for a given projectile or missile and quite often is used for different projectiles and missiles. In this instance one is concerned with programs which are used almost daily for a period of years and any economies, however small, eventually produce large accumulative savings.

Several techniques are commonly used for choosing the mesh size in this type of problem. One method is merely to use a constant mesh of a magnitude which from experience one has confidence will work. Another technique is to establish some empirical rule by experimentation which keys on the magnitude of a certain variable and adjusts the step size as a function of this variable. The third method is the one mentioned previously; i.e., the integration procedure itself estimates the truncation error and adjusts the step size to keep the truncation error within prescribed bounds. This method requires little knowledge of the problem being solved and therefore is more general and flexible. This last approach was the one pursued in the study to be described.

TRAJECTORY PROBLEM

The trajectory problem can be divided into three broad cases.

A Six degrees of freedom

B Five degrees of freedom

C Three degrees of freedom

The six degrees of freedom problem in its most general form is a system of 12 first order coupled nonlinear differential equations. Six of these equations (3 second order D.E.) govern the motion of the center of gravity and the remaining six govern the attitude of the body. The solutions of the equations governing the attitude and angular rates are oscillatory in nature, the oscillations usually having small periods in the independent variable, time. These small periods require small mesh size to trace the oscillation and to maintain stability. This requirement, coupled with the usually large number of total steps needed to take the trajectory to the terminal point creates significantly long computation times.

The major difference between six degree of freedom trajectory problems and five degree* problems is the choice of coordinate systems specifying the orientation of the body axes. For spinning missiles or shell tne computational problems can become extremely difficult when the kinematics are posed in the six degree scheme. The attitude parameters, for example, oscillate sinusoidally with a frequency equal to the spin rate. This spin rate can often be much higher than the pitch and yaw frequencies and where practical the problem should be restated and solved using five degree kinematics. Both of these simulations usually include additional differential equations (in addition to a basic 12 or 11) for representing mass changes, guidance and control etc.

*Five degree systems have also been called "fixed plane systems"[8] and as the "Frick slip frame"[9].

These problems are usually characterized by very complicated derivative sequences, i.e., the time required to evaluate the derivative is usually large in comparison to the time required to execute the operations pertaining to the integration itself.

The three degree of freedom* problem is a system of six first order ordinary differential equations and represents an approximation to the representations described above. In this simulation only the motion of the center of gravity is represented, and since the oscillations are not present this problem is not nearly so demanding of computation time. The fact that very large numbers of these trajectories are computed, however, creates a need for economizing the solution as much as possible.

METHODS STUDIED

Altogether, eight methods were studied as applied to trajectories, and an additional method (after Scraton) was studied on a prototype system of equations. Six of these methods applied to trajectories were predictor-corrector type schemes and among these six were some which differed only in the sequencing of the successive application of the predictor and the corrector step; the other two methods were the RKG method and the Kutta-Merson procedure.

*Recent work done by the Firing Tables Branch of the Computing Laboratory has been successful in superimposing upon the three degree of freedom trajectory for spinning shell an estimate of the "yaw of repose". This simulation approaches the fidelity of the six and five degree simulation at a cost not much greater than the usual three degree of freedom simulation. This model is described by Lieske and Reiter[10].

DESCRIPTION OF THE ALGORITHMS

The differential equation is assumed to be the form

$$Y' = f(x, Y)$$

and the extension to a system is done in the usual manner. The general formulas for the predictor-corrector scheme are as follows:

$$Y^{(p)}_{n+1} = a_1 Y_n + b_1 Y_{n-1} + c_1 Y_{n-2} + d_1 Y_{n-3} + h(e_1 f_n + f_1 f_{n-1} + g_1 f_{n-2} + k_1 f_{n-3}) \tag{1}$$

$$Y^{(c)}_{n+1} = a_2 Y_n + b_2 Y_{n-1} + c_2 Y_{n-2} + h(e_2 f_{n+1} + f_2 f_n + g_2 f_{n-1} + k_2 f_{n-2}) \tag{2}$$

The superscripts (p) and (c) denote predicted and corrected values respectively. After Hull and Creemer we adopt the notation $P(EC)^m$ to denote the mode of application of a pair of formulas. P represents the prediction step, E the evaluation of derivatives and C the corrector step. m is the number of times the cycle (EC) is performed. In this study two values of m were utilized, (m = 1, 2). The RKGMA procedure as described earlier is represented by the cycle PE but was not studied in this investigation because of the previous observations cited and it provides no estimate of the truncation error. The coefficients for the Adam's formulas in equations (1) and (2) can be found in References 2 and 3 and the Crane-Klopfenstein coefficients are found in Reference 5.

The Kutta-Merson formulas are listed below:

KUTTA-MERSON FORMULAS

$$Y_1 = Y_0 + 1/3\, h\, f(x_0, Y_0)$$

$$Y_2 = Y_0 + 1/6\, h\, f(x_0, Y_0) + 1/6\, h\, f(x_0 + 1/3h, Y_1)$$

$$Y_3 = Y_0 + 1/8\, h\, f(x_0, Y_0) + 3/8\, h\, f(x_0 + 1/3h, Y_2)$$

$$Y_4 = Y_0 + 1/2\, h\, f(x_0, Y_0) - 3/2\, h\, f(x_0 + 1/3h, Y_2) + 2hf(x_0 + 1/2h, Y_3)$$

$$Y_5 = Y_0 + 1/6\, h\, f(x_0, Y_0) + 2/3\, h\, f(x_0 + 1/2h, Y_3) + 1/6hf(x_0 + h, Y_4)$$

The procedure provides two estimates of the solution at the end of the step, namely Y_4 and Y_5. If the differential equation is of the form

$$Y' = f(x, Y) = a\,x + b\,Y + c$$

the truncation error in Y_4 is $1/120\, h^5\, Y^{(5)}$ and in Y_5 is $1/720\, h^5\, Y^{(5)}$. The use of this error estimate for systems of equations, none of which have the above form, is merely a computational expediency which seems to work in most instances and should accordingly be used with caution.

LISTING OF METHODS

Code	Method	Formulas
1	RKG	Runge-Kutta-Gill
2	PEC	Adams-Bashforth Predictor
3	PECE	Adams-Bashforth Predictor Adams-Moulton Corrector
4	PECEC	Adams-Bashforth Predictor Adams-Moulton Corrector
5	PEC	Crane-Klopfenstein Predictor
6	PECE	Crane-Klopfenstein Predictor Adams-Moulton Corrector
7	PECEC	Crane-Klopfenstein Predictor Adams-Moulton Corrector
8	KM	Kutta-Merson

ADJUSTMENT OF STEP SIZE

The utilization of truncation error to adjust step size with the predictor-corrector methods is based on the considerations that follow. See Reference 2. If the true solution at the point n+1 is denoted as Y_{n+1} and all the solution and derivative values occurring in the right hand members of (3) and (4) were known exactly, then

$$Y_{n+1} = Y_{n+1}^{(p)} + E_1 \, h^5 \, Y^{(5)}(\xi_1)/5! \tag{3}$$

$$Y_{n+1} = Y_{n+1}^{(c)} + E_2 \, h^5 \, Y^{(5)}(\xi_2)/5! + k\,h\,(f_{n+1}^{(c)} - f_{n+1}^{(p)}) \tag{4}$$

k is equal to 3/8 in the Adams method and ξ_1 and ξ_2 both lie between x_{n-3} and x_{n+1}. The usual procedure is to assume that the term $Q = k\,h\,(f_{n+1}^{(c)} - f_{n+1}^{(p)})$ is equal to zero, a condition which holds only after iteration. Since the iteration requires going through the derivative evaluations each cycle it is generally more efficient to proceed with smaller h and no iteration. The term Q is assumed to be negligible. To get a computable* estimate of the truncation error after the completion of the corrector step the usual assumption is that h is sufficiently small to equate $Y^{(5)}(\xi_1)$ and $Y^{(5)}(\xi_2)$. This leads to the equation

$$\frac{h^5\, Y^{(5)}(\xi_2)}{5!} = \frac{1}{(E_1 - E_2)} \left(Y_{n+1}^{(c)} - Y_{n+1}^{(p)}\right) \tag{5}$$

The error in the corrector step is then approximated by

$$T \approx \frac{E_2}{(E_1 - E_2)} \left(Y_{n+1}^{(c)} - Y_{n+1}^{(p)}\right) \tag{6}$$

*The ability to explicitly evaluate $Y^{(5)}(\xi_2)$ is not practical in most cases and hence is not considered to be computable.

The Kutta-Merson procedure is not really a predictor-corrector method but the sequencing of operations is quite similar, i.e., the two estimates of the solution at n+1 are obtained and the difference is utilized in estimating the truncation error. The following table lists the quantity E2/(E1-E2) for the various methods.

Method	E2/(E1-E2)	Truncation Error
Adams	1/4	$19/720\ h^5\ Y^{(5)}(\xi)$
Crane-Klopfenstein	1/16	$19/720\ h^5\ Y^{(5)}(\xi)$
Kutta-Merson	1/5	$1/720\ h^5\ Y^{(5)}(\xi)$

The classical approach to step size adjustment is to halve or double as the situation warrants. For example, if T denotes the estimated truncation error then the most common techniques attempt to bound T in the interval

$$C_d \leq T \leq C_h \qquad \begin{array}{l} d \text{ -- double} \\ h \text{ -- halve} \end{array} \qquad (7)$$

where C_h denotes the critical value at which to halve and C_d the value at which to double. The motivation for doubling and halving arises from the problems caused by restarting. For instance the problem of restarting when doubling the interval can be solved by storing additional past values and by selecting the appropriate values for the new step size 2h. When halving one can use interpolation formulas to approximate the required data for the new interval h/2. Such formulas can be found in Reference 11. The routines utilized in this study were started by the RKG procedure and therefore this problem of restarting was not present. Accordingly, the step size was not restricted to halving and doubling but was chosen so as to achieve a fixed truncation error. The step size in the predictor-corrector scheme cannot be varied continuously because such an application would mean that most of the computing time would be spent in the starting phase. For the predictor-corrector scheme the step size was changed only when T exceeded the bounds of (8)

$$C_i \leq T \leq C_d \qquad \begin{array}{l} i \text{ -- increase} \\ d \text{ -- decrease} \end{array} \tag{8}$$

When $T > C_d$ the step was discarded and a decreased step size was computed. When $T \leq C_i$ for five steps, then h was increased. The new size was always chosen so as to move T back to the middle of the interval, (8). The KM process is self starting and hence the step size was adjusted continuously. The new step size is determined from the following consideration. The quantity $Y^{(5)}(\xi)$ is assumed to vary slowly and is assumed constant from one step to the next. The old step size is denoted as h_o and the new as h_n. T_m is the maximum truncation error occurring for those equations in the system. h_n is selected to make the truncation error equal to T_d, the desired truncation error. T_m and T_d are represented by

$$T_m = E_2 \, h_o{}^5 \, Y^{(5)}(\xi) \tag{9}$$

$$T_d = E_2 \, h_n{}^5 \, Y^{(5)}(\xi) \tag{10}$$

Solving (9) and (10) for h_n

$$h_n = h_o (T_d/T_m)^{1/5} \tag{11}$$

Since this is only an estimate of the truncation error it is expected that at the end of a step T_m will exceed T_d as often as it is less than T_d. A criterion has to be established for rejecting and repeating a step when $T_m > T_d$. To avoid excessive repeating of steps, (11) is modified as follows:

$$h_n = h_o (fT_d/T_m)^{1/5} \tag{12}$$

f should be smaller than 1 and in this study a value of 0.1 was used.

SETTING THE TRUNCATION ERROR

In the application to trajectories the control on the step size should take into consideration the following factors: (1) the desired end result is a solution giving the position of the center of gravity with an error no greater than several meters, (2) the accuracy* of the parameters governing attitude is not critical since the attitude merely produces a perturbation or second order effect on the motion of the center of gravity. Hence the consideration here is to set the truncation error at a value which will insure stability; i.e., high accuracy is not too important. For this reason the algorithms which were coded for the various methods were designed so as to have individual controls on each equation. This was executed by having the capability of specifying a T_d for each equation. In the six and five-degree study the controls were effective only on P, Q, R, the angular rates and ψ, θ, ϕ the Euler angles. See appendix.

DISCUSSION OF RESULTS

Six and Five Degrees of Freedom

The results shown in Table 1 are representative of the relative economy and accuracy of the eight integration methods for the six and five degree problems. Trajectory 1 is a missile trajectory containing a six degree phase and a five degree phase. The running time spent in the boost phase (six degrees) is approximately the difference in the running time between trajectory I and trajectory II. Trajectory II contains only a five degree phase. Integration type 1 was the RKG method with fixed step size and was used as the standard of comparison. For

*In the absence of a true solution the accuracy can be defined by comparing results against a standard solution. If a given problem is integrated using a particular method for several step sizes, then for a region of h the solution will not vary appreciably. In this region one can choose the standard value.

integration type 1 the integration interval was held constant at 0.01 during the six degree boost phase and at 0.1 during the five degree phase thereafter. The results could be summarized as follows. Of the predictor-corrector methods listed the results are not very conclusive except to indicate that a method using two evaluations is probably better than a method using only one evaluation. This conclusion is drawn from the observation that the running time and the accuracy are not significantly different and two evaluations would generally be better, due to considerations of safety arising from greater stability. No further attempt was made to choose between the various predictor-corrector methods because of the fact that KM by far seemed the best. This is especially true when the accuracy requirement is made less stringent. For truncation errors of 10^{-3}, KM is almost twice as efficient as any predictor-corrector method.

Figure 1 shows the step size history along trajectory II, using the KM procedure and the Adams predictor-corrector method, PECE. In both instances the truncation error was set to 10^{-3}. The data for KM is plotted only at intervals of five seconds.

TABLE I

TRAJECTORY I

SIX AND FIVE DEGREES

INTEGRATION TYPE	TRUNCATION ERROR	TIME OF FLIGHT ERROR SECONDS	DEFLECTION ERROR METERS	RANGE ERROR METERS	COMPUTATION TIME MIN
1	-	-	-	-	4.16
2	10^{-4}	.0001	.1	.1	2.03
3	"	-.0009	1.0	.2	1.85
4	"	.0008	.4	.1	2.01
5	"	.0001	.0	.1	2.08
6	"	-.0010	1.1	.2	2.04
7	"	.0009	.9	-.1	2.01
8	"	.0001	.0	.0	1.76
1	-	-	-	-	4.16
2	10^{-3}	.0034	2.9	2.1	1.77
3	"	-.0049	3.7	1.8	1.49
4	"	.0032	.4	.2	1.71
5	"	.0014	2.9	2.8	1.83
6	"	-.0044	4.0	1.7	1.72
7	"	.0014	1.1	.6	1.64
8	"	-.0013	.5	.2	.93

SIX DEGREES

8	10^{-4}	-.0003	3.4	.0	5.78
8	10^{-3}	-.0249	22.5	-2.2	3.28

THE TRAJECTORY MODEL ON WHICH THIS DATA WAS COLLECTED WAS THAT OF THE LANCE MISSILE SYSTEM. TRAJECTORY I, WHEN USING THE RKG INTEGRATION AND SIX DEGREES OF FREEDOM WITH A CONSTANT STEP SIZE OF .025 SECONDS REQUIRED APPROXIMATELY 15 MINUTES SOLUTION TIME.

TABLE I
(CONT.)

TRAJECTORY II

FIVE DEGREES

INTEGRATION TYPE	TRUNCATION ERROR	TIME OF FLIGHT ERROR SECONDS	DEFLECTION ERROR METERS	RANGE ERROR METERS	COMPUTATION TIME MIN
1	—	—	—	—	2.84
2	10^{-4}	.0001	.2	.1	1.57
3	"	.0009	1.1	.2	1.34
4	"	.0008	.5	.1	1.50
5	"	-.0001	.1	.1	1.59
6	"	-.0010	1.2	.2	1.42
7	"	.0009	.9	-.1	1.40
8	"	.0000	.1	.0	1.13
1	—	—	—	—	2.84
2	10^{-3}	.0041	3.3	2.1	1.46
3	"	-.0045	3.9	1.8	1.07
4	"	.0031	.4	.2	1.27
5	"	.0022	3.0	3.5	1.53
6	"	-.0043	4.0	1.7	1.33
7	"	.0014	.2	.6	1.33
8	"	.0016	.5	.3	.72

SIX DEGREES

8	10^{-4}	-.0004	3.6	.0	5.17
8	10^{-3}	-.0269	21.8	2.4	2.96

5° TRAJECTORY

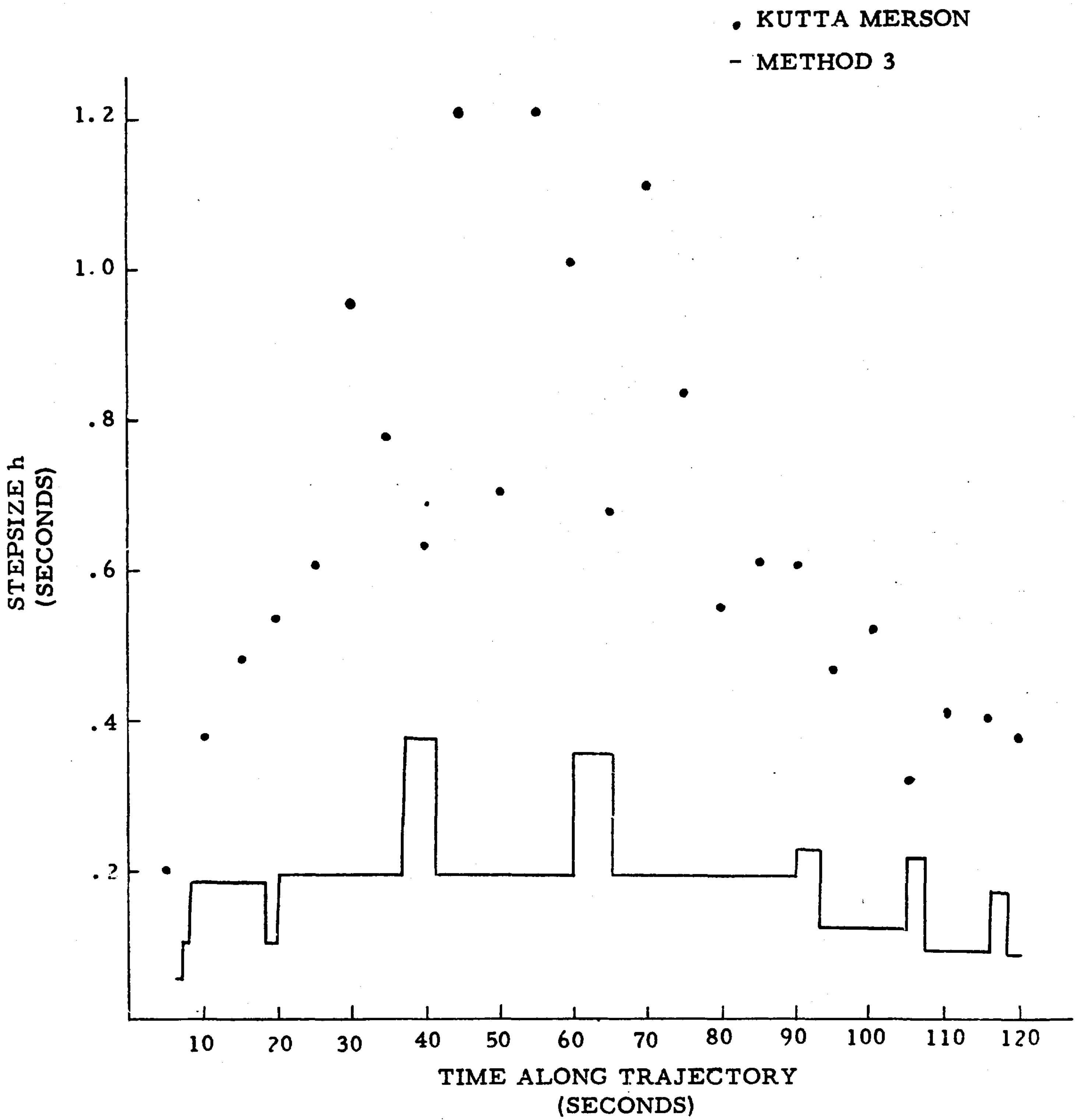

FIGURE 1

Three Degrees

As pointed out previously, the most critical running time problem occurs in the five and six degree trajectory models. Accordingly the emphasis was placed on these problems initially and after KM was selected as the most efficient method, it was applied to the three degree problem. The hope was that Kutta-Merson would prove to be at least as efficient as previously used methods for three degrees thus allowing the use of a single method for all trajectory programs. The equations for the model used in this study are found in Reference 10. In this study KM was compared against a method currently used in the Firing Tables Branch for the mass production of all firing tables for cannon artillery. This method is not widely used elsewhere, but was tailored to this particular application. The method is a self starting predictor-corrector method. The three degree equations are of the form

$\ddot{\vec{X}} = \vec{F}(\vec{X}, \dot{\vec{X}})$ where

$\ddot{\vec{X}}$ is the acceleration vector, $\dot{\vec{X}}$ the velocity and $\vec{X}$ is the position vector. The integration algorithm is as follows.

$$\vec{X}_p = \vec{X}_0 + h\,\dot{\vec{X}}_0$$

$$\dot{\vec{X}}_p = \dot{\vec{X}}_0 + h\,\ddot{\vec{X}}_0$$

$$\ddot{\vec{X}}_p = \vec{F}(\vec{X}_p, \dot{\vec{X}}_p)$$

$$\vec{X}_c = \vec{X}_p - h/2\,(\dot{\vec{X}}_0 - \dot{\vec{X}}_p) + h^2/12\,(\ddot{\vec{X}}_0 - \ddot{\vec{X}}_p)$$

$$\dot{\vec{X}}_c = \dot{\vec{X}}_p - h/2\,(\ddot{\vec{X}}_0 - \ddot{\vec{X}}_p)$$

$$\ddot{\vec{X}}_c = \vec{F}(\vec{X}_c, \dot{\vec{X}}_c)$$

The quantities with the subscript 0 denote values at the beginning of a step. The subscript p denotes predicted values and c denotes corrected values. The predictor formula is that of Euler and has a truncation error of order h^2. The corrector formula for $\vec{X}_c$ is of order h^5 and that of $\dot{\vec{X}}_c$ of order h^2. The fact that the predictor and corrector formulas for X are of different order precludes the estimation of truncation error by the usual methods. Previous techniques for adjusting step size with this method were based on an empirical formula which expressed h as a function of Mach number. Each weapon had its own function, a typical one being as follows:

h	=	.5	0.0	≤	M	<	.764
h	=	.25	0.764	≤	M	<	.936
h	=	.25	0.936	≤	M	<	.960
h	=	.0625	0.960	≤	M	<	.984
h	=	.0625	0.984	≤	M	<	1.016
h	=	.125	1.016	≤	M	<	1.25
h	=	.25	1.25	≤	M	<	4.0

M = Mach Number

The function above was used in the three degree study for the 175mm Gun and a similar function was used for the 155mm Gun. The integration technique described above with step size adjusted as a function of Mach number will be referred to as method A.

An alternative procedure, which has been used on some problems in the Computing Laboratory, is to use the formula $h = C \left| \dot{\vec{X}} / \ddot{\vec{X}} \right|$ to continually adjust h. The constant C is empirical and is determined by experiment. This formula and the integration technique described above will be referred to as method B.

The three degree study consisted of three methods. The first method was the Kutta-Merson procedure, the second was method A and the third was method B. Initially, experiments were conducted to determine control values for the truncation error which would consistently yield good accuracy without sacrificing excessive computation time. The result of these experiments indicated that a good choice of truncation error with the **Kutta** Merson procedure was 0.002 on the three differential equations associated with the velocity and 0.1 on the three equations associated with the position. For Method B a value of C = 0.02 was selected. The three methods were compared on two weapons, the 175mm Gun and the 155mm Gun. Three velocity levels and three quadrant elevations were studied for a total of 18 trajectories for each of the three methods. Each trajectory was run 50 times so that a reliable estimate of the computation time would be obtained. The results are listed in Tables 2 and 3 and can be summarized as follows:

For the 155mm Gun, Method A seems to be as fast as Kutta-Merson or Method B but the accuracy is not acceptable. To reduce the error using Method A would require longer solution time. This is evidenced by the data in Table 3 for the 175mm Gun where the accuracy for Method A is acceptable but the solution time is not competitive with respect to the Kutta-Merson procedure or Method B. In comparing Method B against Kutta-Merson it seems that no significant difference in accuracy or speed is apparent.

The results of this study indicate that Kutta-Merson is a better procedure for the three degree computations than the technique presently used; i.e., the adjustment of h as a function of Mach number, and at least as efficient and accurate as Method B described earlier. Since it is desirable to standardize on a method for various problems within an organization, Kutta-Merson by virtue of being more efficient for six degrees and at least as efficient for three degrees would seemingly be the best method. Another important point at least in the BRL is the following: In some rocket trajectories the simulation is done in stages, five degrees for the first phase, (boost) and three degrees for the second phase (coast).

Kutta-Merson can handle both stages with equal ease.

Figure 2 is a trace of the step size variation as a function of the time along the trajectory for the 155mm Gun. This trajectory had an elevation of 1124 mils and a muzzle velocity of 561 meters/sec. The truncation error was set at 0.01 on all six differential equations of the three degree system.

TABLE 2

THREE DEGREES

KUTTA-MERSON

WEAPON	MUZZLE VEL M/S	QUAD ELEV MILS	COMP TIME FOR 50 TRAJ MIN	TIME OF FLIGHT SECONDS	TIME OF FLIGHT ERROR SECONDS	RANGE M.	RANGE ERROR M	DEFL M	DEFL ERROR M
155	280.0	200	.38	11.015	.000	2870.05	.01	11.24	.04
155	280.0	800	.59	38.557	.001	6684.86	- .09	144.31	.67
155	280.0	1200	1.05	50.116	- .001	4541.49	- .95	270.92	.87
155	378.0	200	.69	13.965	.000	4435.45	.03	21.40	.06
155	378.0	800	.99	48.002	.001	10016.63	- .07	237.25	1.02
155	378.0	1200	1.52	62.594	.000	6916.38	- 1.03	441.10	1.25
155	684.3	200	.79	22.797	.001	9764.53	.29	75.98	.91
155	684.3	800	1.55	68.100	.001	18020.35	- .11	635.09	3.04
155	684.3	1200	2.73	90.005	.000	12530.35	- 3.07	1093.87	2.74

TOTAL=10.29

METHOD A

WEAPON	MUZZLE VEL M/S	QUAD ELEV MILS	COMP TIME FOR 50 TRAJ MIN	TIME OF FLIGHT SECONDS	TIME OF FLIGHT ERROR SECONDS	RANGE M.	RANGE ERROR M	DEFL M	DEFL ERROR M
155	280.0	200	.22	11.010	- .004	2868.74	- 1.24	11.01	- .10
155	280.0	800	.49	38.551	- .005	6683.93	- 1.01	143.72	.26
155	280.0	1200	.60	50.112	- .005	4539.97	- 2.37	270.96	.98
155	378.0	200	.83	13.964	- .001	4435.17	- .15	21.05	- .17
155	378.0	800	.91	47.998	- .003	10016.33	- .42	236.41	.39
155	378.0	1200	1.05	62.592	- .002	6915.09	- 2.26	441.18	1.40
155	684.3	200	1.00	22.793	- .002	9762.55	- 1.57	75.36	.37
155	684.3	800	3.11	68.097	- .007	18019.66	- .76	633.95	1.88
155	684.3	1200	2.94	90.005	.000	12520.28	-12.35	1095.25	3.95

TOTAL=11.15

METHOD B

WEAPON	MUZZLE VEL M/S	QUAD ELEV MILS	COMP TIME FOR 50 TRAJ MIN	TIME OF FLIGHT SECONDS	TIME OF FLIGHT ERROR SECONDS	RANGE M.	RANGE ERROR M	DEFL M	DEFL ERROR M
155	280.0	200	.28	11.012	- .002	2869.32	- .66	10.53	- .58
155	280.0	800	1.00	38.555	- .001	6684.79	- .15	142.56	- .90
155	280.0	1200	1.78	50.116	- .001	4542.19	- .15	269.42	- .56
155	378.0	200	.32	13.961	- .004	4433.93	- 1.39	20.73	- .49
155	378.0	800	1.09	48.000	- .001	10016.77	.02	235.49	.53
155	378.0	1200	1.84	62.592	- .002	6916.90	- .45	439.60	- .18
155	684.3	200	.58	22.792	- .003	9762.33	- 1.79	74.85	- .14
155	684.3	800	1.34	68.095	- .003	18019.48	- .94	633.19	1.12
155	684.3	1200	2.05	89.998	- .007	12529.27	- 3.36	1092.60	1.30

TOTAL=10.28

TABLE 3

THREE DEGREES

WEAPON	MUZZLE VEL	QUAD ELEV	COMP TIME FOR 50 TRAJ	TIME OF FLIGHT	TIME OF FLIGHT ERROR	RANGE	RANGE ERROR	DEFL	DEFL ERROR
	M/S	MILS	MIN	SECONDS	SECONDS	M	M	M	M

KUTTA-MERSON

WEAPON	MUZZLE VEL	QUAD ELEV	COMP TIME FOR 50 TRAJ	TIME OF FLIGHT	TIME OF FLIGHT ERROR	RANGE	RANGE ERROR	DEFL	DEFL ERROR
175	510.5	200	.70	18.643	.001	7248.86	.35	25.76	.43
175	510.5	800	1.27	59.722	.000	15186.80	.05	194.91	.57
175	510.5	1200	1.60	75.540	- .001	10999.15	.51	388.68	- .85
175	704.1	200	.74	24.788	.002	11852.43	.88	64.98	1.26
175	704.1	800	1.53	76.577	.002	22212.09	- .05	410.69	3.34
175	704.1	1200	2.00	100.920	.001	16738.79	.86	697.62	.99
175	914.4	200	.82	30.897	.004	17243.61	1.43	122.26	2.25
175	914.4	800	1.66	96.351	.007	32223.58	.31	882.43	-1.50
175	914.4	1200	2.13	127.643	.000	26104.62	1.99	1291.52	3.86

TOTAL=12.45

METHOD A

WEAPON	MUZZLE VEL	QUAD ELEV	COMP TIME FOR 50 TRAJ	TIME OF FLIGHT	TIME OF FLIGHT ERROR	RANGE	RANGE ERROR	DEFL	DEFL ERROR
175	510.5	200	.90	18.640	- .002	7247.68	- .65	25.04	- .12
175	510.5	800	1.81	59.718	- .003	15185.48	- 1.24	193.56	- .12
175	510.5	1200	1.98	78.534	- .007	10997.56	- 1.16	387.98	-1.22
175	704.1	200	.81	24.783	- .002	11850.24	- 1.12	63.77	.24
175	704.1	800	3.38	76.572	- .003	22210.82	- 1.28	407.95	.85
175	704.1	1200	2.80	100.913	- .006	16737.49	- .52	695.63	- .70
175	914.4	200	.89	30.892	- .001	17240.73	- 1.29	120.68	.84
175	914.4	800	3.43	96.342	- .003	32220.39	- 2.77	876.99	3.02
175	914.4	1200	3.18	127.632	- .010	26104.50	1.63	1287.88	.46

TOTAL=19.18

METHOD B

WEAPON	MUZZLE VEL	QUAD ELEV	COMP TIME FOR 50 TRAJ	TIME OF FLIGHT	TIME OF FLIGHT ERROR	RANGE	RANGE ERROR	DEFL	DEFL ERROR
175	510.5	200	.41	18.636	- .006	7246.09	- 2.24	24.19	- .97
175	510.5	800	1.16	59.717	- .004	15186.23	- .49	192.26	-1.71
175	510.5	1200	1.87	78.533	- .008	10998.49	- .23	387.82	-1.38
175	704.1	200	.54	24.780	- .005	11848.74	- 2.62	62.58	- .95
175	704.1	800	1.30	76.569	- .006	22210.49	- 1.61	406.82	- .28
175	704.1	1200	1.94	100.904	- .015	16737.25	- .76	695.73	- .60
175	914.4	200	.65	30.888	- .005	17239.26	- 2.76	119.38	- .46
175	914.4	800	1.41	96.335	- .010	32219.31	- 3.85	877.87	3.90
175	914.4	1200	2.01	127.617	- .025	26100.67	- 2.20	1288.65	1.23

TOTAL=11.29

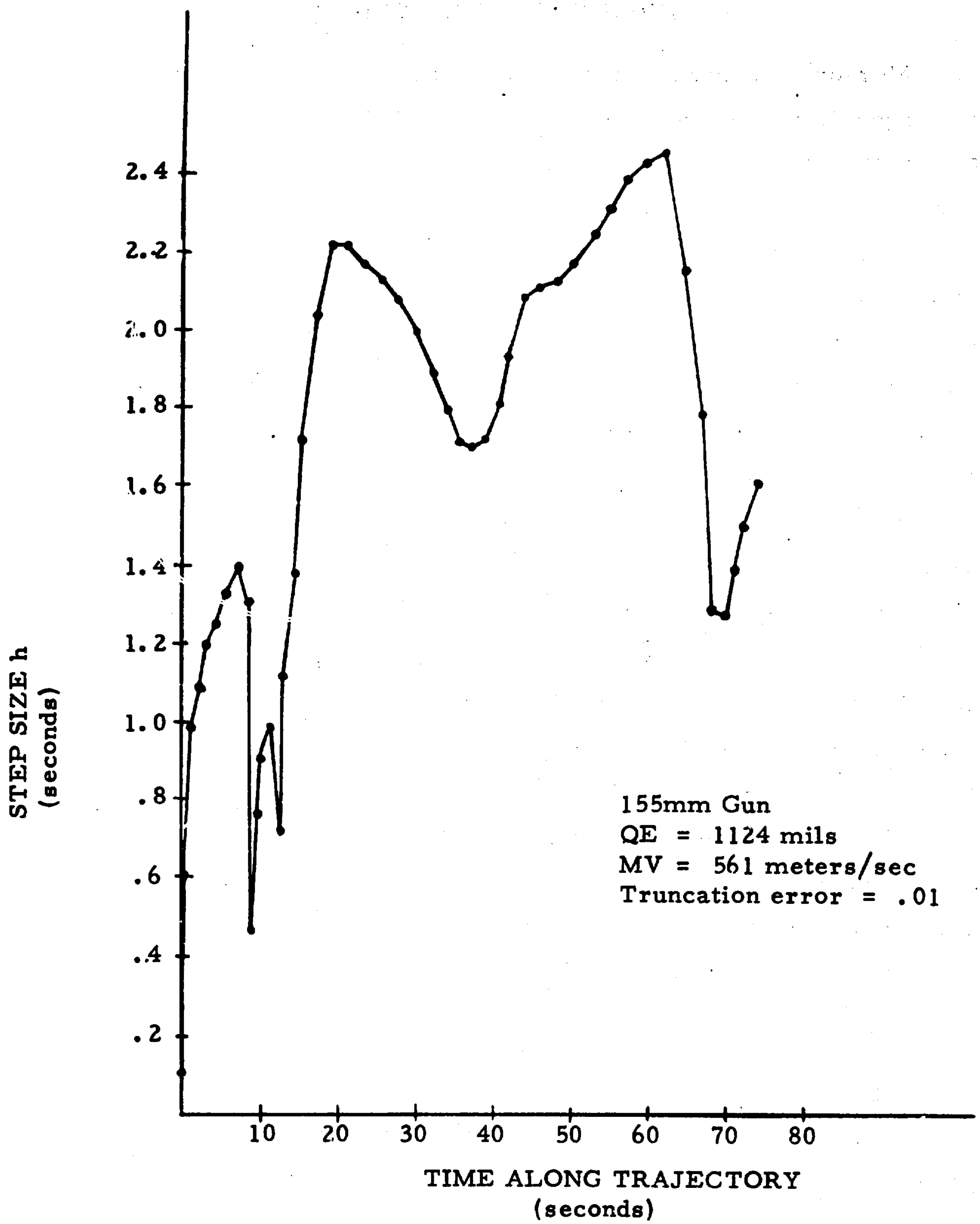
STEP SIZE h
(seconds)
2.4
2.2
2.0
1.8
1.6
1.4
1.2
1.0
.8
.6
.4
.2
10
20
30
40
50
60
70
80
155mm Gun
QE = 1124 mils
MV = 561 meters/sec
Truncation error = .01
TIME ALONG TRAJECTORY
(seconds)

FIGURE 2

EFFICIENCY OF THE METHODS

Merson[6] observed that for dynamic problems, one could use with the Runge-Kutta method, an integration interval perhaps seven times larger than with the Milne predictor. This general observation seems to be borne out by this study.

The results are somewhat surprising in that it is common to attribute to the predictor-corrector methods a greater efficiency than the Runge-Kutta methods. One usually assumes that when the order of two methods is the same and one has say two evaluations in one method and four in the other then the method with two evaluations on the surface is twice as efficient as the other. Hence one is inclined to attribute the contrary results of this study to considerations of stability. This is probably true but two other factors seem important.

The first of these is that to avoid the cost of restarting with the predictor-corrector methods, one has to employ the bounding conditions specified in equation (8). This means that one is not always using the largest step size consistent with the desired truncation error as, e.g., one uses with the KM procedure.

The second factor is accuracy itself. If RKGMA, the conventional Adams method, and KM produced the same accuracy for a given step size then one might choose to assign to each method a theoretical cost index as shown in the second column of Table 3.

TABLE 3

Method	Cost Index	Cost Accuracy Index
RKGMA	1.0	1.0
Adams	2.0	1.2
Kutta-Merson	5.0	1.7

Since the methods do not yield the same accuracy (despite all being fifth order) for the same step size, it is more realistic to define a cost-accuracy index. A cost-accuracy index can be defined as the quantity N/R where N is the number of derivative evaluations per step and R is a step size ratio required to achieve a fixed truncation error. This ratio can best be found by comparing methods against a standard method. For the purpose of comparison let the RKGMA method be the standard with a cost accuracy index of 1.0. KM has an N of 5 and can allow a step size of $(251)^{1/5} \approx 3$ times larger than RKGMA to achieve the same truncation error. This ratio is obtained by equating the truncation error of the two methods and solving for the step size ratio. The cost-accuracy index for the three methods is shown in the third column of Table 3.

Even with the more realistic comparison of a cost-accuracy index the Kutta-Merson procedure shows up as the least efficient method. The contrary results of this study show that for dynamic problems, such as those studied here, efficiency can only be determined by experimentation.

The cost-accuracy index, however, suggests that the additional evaluations do more than contribute to stability, they also increase accuracy, and therefore efficiency. An additional factor which should be noted is the following: In representing the propulsion and aerodynamic characteristics of a missile, one is usually faced with the problem of approximating the data by some curve fitting technique so that it can be utilized in the trajectory problem. Because of the severe nonlinearities of the data and because of practical considerations it is customary to approximate the data by piecewise polynomials. At the juncture of these polynomials the derivatives are not continuous and hence one of the basic assumptions of any integration technique, i.e., the continuity of higher derivatives, is violated. Intuitively it would seem that an integration method which uses information only within one interval (Runge-Kutta) as compared to one which uses information over several intervals (predictor-corrector) would be less affected by this problem. In this study, the data in the six and five degree problems were represented by piecewise linear segments. In the three degree problem piecewise 4th degree polynomials were used.

Since the completion of this study the author has become aware of work done by Scraton[7] who obtains an estimate of the truncation error for the Runge-Kutta method when making the same assumptions as does Merson.

This assumption is that the differential equation is the form

$$Y' = ax + bY + c \qquad (13)$$

where a, b, and c are constants.

In addition Scraton has derived Runge-Kutta type formulas which require five evaluations and which have a computable estimate of the truncation error not based on the above assumptions, i.e., they have general validity. This recent work raises the question as to whether the Runge-Kutta procedure with Scraton's truncation error estimate might not be a better method than the KM procedure since it requires only four derivative evaluations. Furthermore, might not Scraton's new formulas be more attractive because of their general validity. Mr. Glenn Beck of the Computing Laboratory has observed that the KM procedure as used with a prototype differential equation yields greater accuracy than does RK or Scraton's method and the difference in accuracy compensates for the fifth evaluation. The belief is e.g., see Fox[13], that when the system of equations does not follow the form (13), that Merson's truncation error estimate is an overestimate, i.e., it is safe.

CONCLUSIONS

In five and six degree of freedom problems the Kutta-Merson integration procedure as compared against the fifth order predictor-corrector techniques seems to be twice as economical in terms of computation time when applied in a manner which allows continuous interval adjustment. When compared against the RKG procedure with constant step size, the data for this report indicate savings of almost 4 to 1. In most cases the ratio would be higher because one usually cannot justify an extensive stability study to find the largest value of h required to yield the desired accuracy and maintain stability.

For three degree problems the Kutta-Merson procedure seems to be more efficient than presently used techniques and at least as efficient as a proposed modification.

The adoption of Kutta-Merson as the standard subroutine for all trajectory computations can result in significant savings of computation time in addition to the benefits gained from the standardization. In addition it has already proved quite useful on a number of other problems.

ACKNOWLEDGEMENTS

The author is indebted to Mr. Glenn Beck who coded the integration routines and made suggestions pertaining to the various schemes for interval adjustments. The routines were incorporated into the six degree trajectory programs by Mr. Henry Wisniewski who also was of great assistance in obtaining data for this report. The three degree data were obtained by Mr. Donald McCoy. The Kutta-Merson routine was incorporated into the three degree program by Mr. Joseph Hurff.

AUTHOR'S NOTE

This study has resulted in the Kutta-Merson procedure being coded as an integration package which is now part of the BRLESC compiler. The subroutine description can be obtained by contacting the Computation Branch of the Computing Laboratory.

REFERENCES

1. Romanelli, Michael J., Runge Kutta Methods for the Solution of Ordinary Differential Equations, Mathematical Methods for Digital Computers, edited by Ralston and Wilf, John Wiley and Sons Inc.

2. Hildebrand, F. B., Introduction to Numerical Analysis, McGraw Hill Book Company Inc.

3. Hamming, R. W., Numerical Methods for Scientists and Engineers, McGraw Hill Book Company Inc. 1962.

4. Hull, T. E. and Creemer, A. L., Efficiency of Predictor-Corrector Procedures, Journal of the Association for Computing Machinery, April 1963.

5. Crane, R. L. and Klopfenstein, R. W., A Predictor Corrector Algorithm with an Increased Range of Absolute Stability, Journal of the Association for Computing Machinery, Vol. 12, No. 2, April 1965.

6. *Merson, R. H., An Operational Method for Study of Integration Processes, Proceedings of Symposium on Data Processing, Weapons Research Establishment, Salisbury, S. Australia.

7. Scraton, R. E., Estimation of the Truncation Error in Runge-Kutta and Allied Processes, Computer Journal, October 1964.

8. Barnett, Bruce, Trajectory Equations for a Six-Degree-Of-Freedom Missile, Technical Memorandum FRL-TM-25, Picatinny Arsenal, Dover, N. J.

9. Daniels, P., Near Maximum Time Step for Rigid Body Integration Using the Frick Slip Frame , NWL Report 2002, Dahlgren, Virginia.

*This reference is not generally available but has been obtained by the Author through correspondence.

10. Lieske, Robert F., and Reiter, Mary L., Equations of Motion for a Modified Point Mass Trajectory, BRL Report No. 1314.

11. Ralston, Anthony, Numerical Integration Methods for the Solution of Ordinary Differential Equations, Mathematical Methods for Digital Computers, Edited by Ralston and Wilf, John Wiley and Sons Inc.

12. Fox, L., Numerical Solution of Ordinary and Partial Differential Equations, Addison Wesley Publishing Co., Inc.

13. Zaroodny, Serge J., An Exact Re-statement of the Equations of Motion of a Rigid Projectile for use with Modern Computing Machinery, BRL Memorandum Report No. 856, September 1954.

14. Lieske, Robert F. and McCoy, Robert L., Equations of Motion of a Rigid Projectile, BRL Report No. 1244, March 1964.

15. Six Degree of Freedom Flight-Path Study Generalized Computer Program, Technical Documentary Report No. RTD-TDR-64-1, Air Force Flight Dynamics Laboratory, Wright Patterson Air Force Base, Ohio.

APPENDIX

EQUATIONS OF MOTION

List of Symbols

X_o, Y_o, Z_o	Position in inertial space
F_{px}, F_{py}, F_{pz}	Propulsion forces about body axes $\vec{x}$, $\vec{y}$, $\vec{z}$
F_{ax}, F_{ay}, F_{az}	Aerodynamic forces about body axes $\vec{x}$, $\vec{y}$, $\vec{z}$
G_{xo}, G_{yo}, G_{zo}	Components of gravitational vector along inertial axes $\vec{x}_o$, $\vec{y}_o$, $\vec{z}_o$
L_{px}, L_{py}, L_{pz}	Propulsion moments about body axes $\vec{x}$, $\vec{y}$, $\vec{z}$
L_{ax}, L_{ay}, L_{az}	Aerodynamic moments about body axes $\vec{x}$, $\vec{y}$, $\vec{z}$
P, Q, R	Angular rates of missile about body axes $\vec{x}$, $\vec{y}$, $\vec{z}$
I_x, I_y, I_z	Moments of inertia about body axes $\vec{x}$, $\vec{y}$, $\vec{z}$
ψ, θ, ϕ	Euler angles
[C]	Transformation matrix between body and inertial axes
$\vec{x}_o$, $\vec{y}_o$, $\vec{z}_o$	Unit vectors defining the inertial axes
$\vec{x}$, $\vec{y}$, $\vec{z}$	Unit vectors defining the body axes

LISTING OF EQUATIONS

$$m \begin{pmatrix} \ddot{X}_o \\ \ddot{Y}_o \\ \ddot{Z}_o \end{pmatrix} = [C] \begin{pmatrix} F_{px} + F_{ax} & & G_{xo} \\ F_{py} + F_{ay} & + & G_{yo} \\ F_{pz} + F_{az} & & G_{zo} \end{pmatrix} \quad \text{(A1)}$$

$$\begin{aligned} \dot{P} &= 1/I_x \left[L_{px} + L_{ax} - (I_z - I_y)\, QR \right] \\ \dot{Q} &= 1/I_y \left[L_{py} + L_{ay} - I_x\, PR + I_z\, \bar{P}R \right] \\ \dot{R} &= 1/I_z \left[L_{pz} + L_{az} - I_y\, \bar{P}Q + I_x\, PQ \right] \end{aligned} \quad \text{(A2)}$$

$$\begin{cases} \bar{P} = -R \tan\theta & \text{Five Degrees} \\ \bar{P} = P & \text{Six Degrees} \end{cases}$$

$$\begin{aligned} \dot{\theta} &= Q \cos\phi - R \sin\phi \\ \dot{\psi} &= 1/\cos\theta \left[Q \sin\phi + R \cos\phi \right] \\ \dot{\phi} &= \bar{P} + \dot{\psi} \sin\theta \end{aligned} \quad \text{(A3)}$$

$$[C] = \begin{pmatrix} l_1 & m_1 & n_1 \\ l_2 & m_2 & n_2 \\ l_3 & m_3 & n_3 \end{pmatrix}$$

$$\overset{*}{=} \begin{pmatrix} C\theta C\psi & S\theta S\phi C\psi - C\phi S\psi & S\theta C\phi C\psi + S\phi S\psi \\ C\theta S\psi & S\theta S\phi S\psi + C\phi C\psi & S\theta C\phi S\psi - S\phi C\psi \\ -S\theta & S\phi C\theta & C\phi C\theta \end{pmatrix} \quad \text{(A4)}$$

* $C\theta = \cos\theta$, $S\theta = \sin\theta$ etc.

The equations of motion listed above are those used in the six and five degree study. These equations are general and can be used for any missile or projectile which has axial symmetry. The detailed listing of the computations involved in obtaining the propulsion, aerodynamic and gravitational forces and moments are not included, but as pointed out earlier these computations are very lengthy. The transformation matrix relating the body axes and the inertial axes is obtained by an Euler angle system. The equations are posed for programming convenience so that the program can switch from six degrees to five degrees with only minor changes in program logic.

These equations are conventional except for the presence of $\bar{P}$ in the last two equations of (A2). For the six degree system $\bar{P}$ is equal to P. For the five degree system $\bar{P}$ is determined by

$$\bar{P} = -R \tan\theta \qquad \text{(A5)}$$

In addition, for the five degree system $\phi = \dot{\phi} = 0$ in (A3) and (A4).

The singularity which can occur in the $\dot{\psi}$ equation when θ approaches 90° is avoided by utilizing a "switch over" set of intermediate axes $\theta_c \le \theta \le 90°$.

It should be noted that alternative techniques for obtaining [C] are often used. A method which seems to be used as often as the Euler angle method is to integrate the direction cosine derivatives directly by the equations

$$\begin{aligned} \dot{l}_i &= R\, m_i - Q\, n_i \\ \dot{m}_i &= \bar{P}\, n_i - R\, l_i \qquad i = 1, 2, 3 \\ \dot{n}_i &= Q\, l_i - \bar{P}\, m_i \end{aligned} \qquad \text{(A6)}$$

This method avoids the evaluation of the trigonometric terms in [C] at the expense of the additional differential equations. The experience of various Government agencies and the industry with this technique however is that [C] tends to gradually deviate from orthogonality (especially for spinning missiles) and accentuates instability. This problem is avoided by inserting into the program an orthogonality loop such as that described in Reference 15 which at each step of the integration forces [C] back to a condition of orthogonality. Equations (A6) can be utilized with both five and six degree systems. For five degree systems $\bar{P}$ is found from the equation

$$\bar{P} = -R \tan l_3 / m_3 \tag{A7}$$

An interesting observation is that conventional six degree systems such as that described by Zaroodney[11] which utilize (A6) for finding [C] can be modified to five degrees by simply replacing P with $\bar{P}$ as noted in (A6) and defining $\bar{P}$ as in (A7).

The rigid body system of equations most widely used in firing table computations at BRL is that described by Lieske and McCoy[12] and due originally to H. L. Reed. This system is a five degree one and integrates the direction cosine derivatives but does not employ the "body axes" as is usually done, i.e., all integrations are done in inertial space. The method enjoys the advantage of being simple but suffers the disadvantage of being unconventional. The missile engineer with the customary education in the aeronautical schools usually prefers to express the forces and moments in terms of parameters associated with the "body axes" such as P, Q, R, the angular rates, α, β, the angle of attack and angle of sideslip and ψ, θ, ϕ, the Euler angles. This seems especially true for guided missile systems which have "on board" inertial platforms which in some way or other are "tied" to the body axes.

Unclassified
Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) U. S. Army Ballistic Research Laboratories Aberdeen Proving Ground, Md	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

3. REPORT TITLE AN EFFICIENCY STUDY OF SEVERAL TECHNIQUES FOR THE NUMERICAL INTEGRATION OF THE EQUATIONS OF MOTION FOR MISSILES AND SHELL

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

5. AUTHOR(S) (First name, middle initial, last name)

Breaux, Harold J.

6. REPORT DATE February 1967	7a. TOTAL NO. OF PAGES 46	7b. NO. OF REFS 15
8a. CONTRACT OR GRANT NO. b. PROJECT NO. RDT & E 1P523801A287 c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) Report Number 1358 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)	

10. DISTRIBUTION STATEMENT This document is subject to special export controls and each transmittal to foreign governments or foreign nationals may be made only with prior approval of Commanding Officer, U.S. Army Ballistic Research Laboratories, Aberdeen Proving Ground, Maryland

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY U.S. Army Materiel Command Washington, D.C.

13. ABSTRACT

The numerical integration of the equations of motion for missiles and shell is a frequently occurring computational problem in government laboratories and in the aerospace industry. Because of the repetitive nature of such computations and the continuing requirement, it is desirable to make these computations as efficiently as possible in order to minimize the computer time spent in their solution. The cost in terms of time required to solve representative problems to a specified accuracy is a measure of the relative efficiency of numerical integration methods and can be determined only by experimentation. This report describes the results of such an experimental study and finds that the Kutta-Merson procedure with automatic and continuous interval adjustment is far superior to the predictor-corrector techniques.

DD FORM 1 NOV 65 1473 REPLACES DD FORM 1473, 1 JAN 64, WHICH IS OBSOLETE FOR ARMY USE.

Unclassified
Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Numerical Integration						
Equations of Motion						
Automatic Interval Adjustment						
Ordinary Differential Equations						