SIMULATION
http://sim.sagepub.com

The treatment of inputs in real-time digital simulation

Dennis S. Bernstein

SIMULATION 1979; 33; 65
DOI: 10.1177/003754977903300206

Keywords: digital integration, digital simulation techniques

The treatment of inputs in real-time digital simulation

by

Dennis S. Bernstein
Applied Dynamics International
3800 Stone School Road
Ann Arbor, Michigan 48104

DENNIS S. BERNSTEIN received his BS degree in applied mathematics (1977) from Brown University in Providence, Rhode Island. He recently obtained his MS in control from the Program in Computer, Information and Control Engineering at the University of Michigan, where he is currently working toward his PhD. Since 1978 he has been employed as a member of the technical staff of Applied Dynamics International.

ABSTRACT

Real-time simulation often requires a computer to interact with external hardware in a dynamic loop. Data must be sampled from input signals and incorporated into the numerical integration algorithm to evaluate the derivatives of the state variables. Also, data from the integration must be available for external use with a minimum of delay. This paper will show that, because of these requirements, certain numerical integration routines are best suited for real-time simulation. A new fourth-order Runge-Kutta routine designed for real-time use is presented along with its derivation.

INPUT-OUTPUT DIFFICULTIES IN CERTAIN INTEGRATION SCHEMES

Consider the real-time integration of the system

$$\dot{y}(t) = f[t, y(t), r(t)]$$

where $y(t)$ represents the state variables and $r(t)$ is the set of external input variables. To use a numerical integration routine we assume, as is usually the case, that inputs can be sampled before each evaluation of the state-variable derivatives and that the evaluations of the derivatives take most of the computing time. We also assume that the numerical integration interval h is chosen to be as small as possible. A lower bound on the size of h is usually determined by the speed of the digital computer when it is being taxed to keep up with real time. Thus, in order to keep the integration accuracy at an acceptable level, the smallest possible value must often be used.

Suppose that the integration is performed with the classical fourth-order Runge-Kutta method. This routine has the form

$$y_{n+1} = y_n + \frac{1}{6}h\,[f_{n0} + 2f_{n1} + 2f_{n2} + f_{n3}]$$

$$f_{n0} = f[t_n,\ y_n,\ r(t_n)]$$

$$f_{n1} = f[t_n + \tfrac{1}{2}h,\ y_n + \tfrac{1}{2}hf_{n0},\ r(t_n + \tfrac{1}{2}h)]$$

$$f_{n2} = f[t_n + \tfrac{1}{2}h,\ y_n + \tfrac{1}{2}hf_{n1},\ r(t_n + \tfrac{1}{2}h)]$$

$$f_{n3} = f[t_n + h,\ y_n + hf_{n2},\ r(t_n + h)]$$

In real time each evaluation of the state variable derivatives must be accomplished in $h/4$ time units. Note that the method requires the input value $r(t_n + \frac{1}{2}h)$ to evaluate f_{n1}. This evaluation, however, must be initiated at time $t_n + h/4$. Similarly, the fourth derivative evaluation, which begins at time $t_n + 3h/4$, requires $r(t_n + h)$, the value of the input at the end of the interval. For these two derivative evaluations some form of extrapolation is needed to approximate the required input data when it is needed, i.e., $h/4$ time units before it is available. The extrapolation procedure, besides increasing the computing overhead, results in reduced accuracy, especially when the inputs contain high-frequency components. This problem has previously been referred to in the literature, e.g., on page 8 of Reference 2.

An alternative to the extrapolation of inputs is to perform the classical fourth-order Runge-Kutta integration $h/4$ time units behind real time and then extrapolate the output $h/4$ time units ahead to compensate for the delay.

The same sort of timing problem arises with the Adams-Moulton predictor-corrector routines. For these methods the end-of-interval input value $r(t_n + h)$ is required at time $t_n + h/2$, requiring extrapolating ahead by $h/2$ time units.

ELIMINATING EXTRAPOLATION

There are integration routines which, when used for real-time simulation, do not require extrapolation of either input or output values. The Adams-Bashforth predictors and certain Runge-Kutta methods are examples.

It is well known that infinitely many different Runge-Kutta formulas exist for each order of 2 or greater. We are particularly interested in the second-order formula (page 47 of Reference 4).

$$y_{n+1} = y_n + hf_{n1}$$

$$f_{n0} = f[t_n, y_n, r(t_n)]$$

$$f_{n1} = f[t_n + \tfrac{1}{2}h, y_n + \tfrac{1}{2}hf_{n0}, r(t_n + \tfrac{1}{2}h)]$$

and the third-order formula (page 49, *ibid.*)

$$y_{n+1} = y_n + \frac{1}{4}h(f_{n0} + 3f_{n2})$$

$$f_{n0} = f[t_n, y_n, r(t_n)]$$

$$f_{n1} = f[t_n + \frac{1}{3}h, y_n + \frac{1}{3}hf_{n0}, r(t_n + \frac{1}{3}h)]$$

$$f_{n2} = f[t_n + \frac{2}{3}h, y_n + \frac{2}{3}hf_{n1}, r(t_n + \frac{2}{3}h)]$$

These formulas require sampled input data at equally spaced intervals, and hence no extrapolation is needed for real-time simulation. We note that these two routines and the classical fourth-order formula are "purely iterative" in that the evaluation of the derivative f_{ni}, $i \geq 1$, requires only the preceding value $f_{n,i-1}$. Compared to a method that is not purely iterative, a routine of this form has the advantages of fast execution and minimal storage requirements.

The situation is not so fortunate for a fourth-order method. It is shown on page 207 of Reference 3 that any four-stage, fourth-order Runge-Kutta formula requires $r(t_n + h)$ for the evaluation of the fourth derivative. Since this derivative evaluation must begin at time $t_n + 3h/4$, we see that there cannot exist a four-stage, fourth-order Runge-Kutta method that does not require extrapolation when used for real-time simulation.

DERIVATION OF A NEW FORMULA

We now derive a five-stage Runge-Kutta method that has fourth-order accuracy but does not require extrapolation. We shall choose the coefficients to be small rational numbers with a minimal least common denominator to facilitate numerical computation. Also, zero values will be chosen wherever possible to reduce execution time and storage requirements.

The derivation proceeds as follows: a five-stage Runge-Kutta method has the form

$$y_{n+1} = y_n + h \sum_{k=0}^{4} c_k f_{nk}$$

where

$$f_{n0} = f(t_n, y_n, r_n)$$

$$f_{nk} = f(t_n + a_k h, y_n + h\hat{f}_{nk}, r_{nk}) \; ; \qquad k = 1, \ldots, 4$$

$$\hat{f}_{nk} = \sum_{\ell=0}^{k-1} b_{k\ell} f_{n\ell}$$

$$r_{nk} = r(t_n + a_k h)$$

Real-time operation requires that

$$a_k = \frac{k}{5} \quad ; \quad k = 1, 2, 3, 4 \tag{1}$$

To realize a fourth-order method, the remaining fifteen parameters must satisfy twelve equations. These equations are given on pages 17 and 18 of Reference 1. Using (1), we can write these equations as follows:

I $\quad c_0 + c_1 + c_2 + c_3 + c_4 = 1$

II $\quad c_1 + 2c_2 + 3c_3 + 4c_4 = \frac{5}{2}$

III $\quad c_1 + 4c_2 + 9c_3 + 16c_4 = \frac{25}{3}$

IV $\quad c_1 + 8c_2 + 27c_3 + 64c_4 = \frac{125}{4}$

V $\quad c_2 b_{21} + c_3(b_{31} + 2b_{32}) + c_4(b_{41} + 2b_{42} + 3b_{43}) = \frac{5}{6}$

VI $\quad c_2 b_{21} + c_3(b_{31} + 4b_{32}) + c_4(b_{41} + 4b_{42} + 9b_{43}) = \frac{25}{12}$

VII $\quad c_2 b_{21} + 3c_3(b_{31} + 2b_{32}) + 4c_4(b_{41} + 2b_{42} + 3b_{43}) = \frac{25}{8}$

VIII $\quad c_3 b_{21} b_{32} + c_4(b_{21} b_{42} + b_{31} b_{43} + 2b_{32} b_{43}) = \frac{5}{24}$

IX $\quad b_{10} = \frac{1}{5}$

X $\quad b_{20} + b_{21} = \frac{2}{5}$

XI $\quad b_{30} + b_{31} + b_{32} = \frac{3}{5}$

XII $\quad b_{40} + b_{41} + b_{42} + b_{43} = \frac{4}{5}$

By algebraic manipulation we find that the minimal least common denominator of the coefficients $c_0, \ldots, c_4$ satisfying I, II, III, and IV is 24. If we choose

$$c_4 = \frac{5}{12}$$

the other coefficients are

$$c_0 = -\frac{1}{24}$$

$$c_1 = \frac{5}{8}$$

$$c_2 = -\frac{5}{24}$$

$$c_3 = \frac{5}{24}$$

If we now fix b_{21} and b_{43}, the one nonlinear equation (VIII) becomes linear and Equations V, VI, VII, and VIII form a system of four linear equations in four unknowns. For all possible choices of b_{21} and b_{43} the coefficient matrix of the resulting system is singular (rank = 3). However, Equations V - VIII can be solved, and they have infinitely many solutions if and only if b_{21} and b_{43} satisfy

$$3b_{21} - 12b_{21}b_{43} + 2b_{43} = 1$$

Choosing

$$b_{21} = 0$$

and

$$b_{43} = \frac{1}{2}$$

to satisfy the above constraint and using the last degree of freedom to choose

$$b_{32} = 0$$

we obtain

$$b_{31} = 1$$

$$b_{41} = 0$$

$$b_{42} = 0$$

The foregoing steps lead to the following integration formula:

$$y_{n+1} = y_n + \frac{1}{24}h\,(-f_{n0} + 15f_{n1} - 5f_{n2} + 5f_{n3} + 10f_{n4})$$

$$f_{n0} = f[t_n,\ y_n,\ r(t_n)]$$

$$f_{n1} = f[t_n + \tfrac{1}{5}h,\ y_n + \tfrac{1}{5}hf_{n0},\ r(t_n + \tfrac{1}{5}h)]$$

$$f_{n2} = f[t_n + \tfrac{2}{5}h,\ y_n + \tfrac{2}{5}hf_{n0},\ r(t_n + \tfrac{2}{5}h)]$$

$$f_{n3} = f[t_n + \tfrac{3}{5}h,\ y_n - \tfrac{2}{5}hf_{n0} + hf_{n1},\ r(t_n + \tfrac{3}{5}h)]$$

$$f_{n4} = f[t_n + \tfrac{4}{5}h,\ y_n + \tfrac{3}{10}hf_{n0} + \tfrac{1}{2}hf_{n3},\ r(t_n + \tfrac{4}{5}h)]$$

For the execution of this routine, note that f_{n2} need not be stored. After its computation it can be entered into the equation for y_{n+1}. Also, the memory space for f_{n1} can be used to store f_{n3}. Unfortunately, no five-stage fourth-order formula exists that is purely iterative. We can see this by setting

$$b_{20} = b_{30} = b_{31} = b_{40} = b_{41} = b_{42} = 0$$

and using V - XII. Other routines can be derived, however, by considering tradeoffs among execution speed, memory requirements, and integration accuracy.

Table 1

Comparison of truncation error coefficients

Coefficient	Proposed real-time formula	Classical fourth-order formula
T_1	$\frac{-1}{120}$	$\frac{-1}{120}$
T_2	$\frac{-1}{240}$	$\frac{1}{480}$
T_3	0	$\frac{-1}{240}$
T_4	$\frac{-1}{1\,800}$	$\frac{-1}{720}$
T_5	0	$\frac{1}{120}$
T_6	$\frac{-1}{1\,200}$	$\frac{-1}{480}$
T_7	$\frac{-1}{480}$	$\frac{1}{160}$
T_8	$\frac{-1}{400}$	$\frac{1}{480}$
T_9	$\frac{-1}{3\,600}$	$\frac{1}{2\,880}$
$\sum_{i=1}^{9} \lvert T_i \rvert$	0.019	0.035
$\left(\sum_{i=1}^{9} T_i^2\right)^{\frac{1}{2}}$	0.009 9	0.015

ACCURACY OF THE PROPOSED ROUTINE

Truncation error coefficients are useful indicators of the accuracy of an integration routine. These coefficients multiply the leading terms in the Taylor expansion of the local, or one-step, integration error. The relative importance of the various terms in the expansion, and hence of their respective coefficients, depends on the system being integrated. Two rough indicators of accuracy that are often used to compare Runge-Kutta formulas are the sum of the absolute values and the square root of the sum of the squares. These values are given in Table 1 along with those of the classical fourth-order formula for comparison. The numbering of the coefficients corresponds to page 10 of Reference 1.

CONCLUSION

We have pointed out that certain second- and third-order Runge-Kutta formulas may be well suited for real-time simulation since they do not require extrapolation. It is also important that these formulas have approximately the same accuracy as the other well-known Runge-Kutta methods of their orders (see, for example, Reference 4).

The proposed integration routine requires five derivative evaluations per integration step as compared to four per step for the classical fourth-order Runge-Kutta method. Since most of the computing time is used to evaluate the derivatives, maintaining real-time speed with the proposed routine requires a step size that is 25% larger than that used for the classical fourth-order method. Assuming that both routines have approximately the same truncation error properties, this represents an increase in the local truncation error by a factor of $(5/4)^5 \approx 3$ for asymptotically small h. Thus, since the proposed routine eliminates the need for extrapolation, its use is recommended for real-time simulation when the inputs contain high-frequency components and the derivative evaluations do not present an excessive burden to the computer.

REFERENCES

1 FEHLBERG, E.
Low Order Classical Runge-Kutta Formulas with Step-Size Control and Their Application to Some Heat Transfer Problems
NASA Technical Report R-315 1969

2 HOWE, R.M. FOGARTY, L.E.
Computer Considerations for Real Time Simulation of a Generalized Rotor Model
NASA Technical Report CR-2877 1977

3 KOPAL, Z.
Numerical Analysis
Wiley New York 1955

4 LAPIDUS, L. SEINFELD, J.H.
Numerical Solution of Ordinary Differential Equations
Academic Press New York 1971

SCS Regional Council election results

SOCIETY FOR COMPUTER SIMULATION
WESTERN REGIONAL COUNCIL
STEERING COMMITTEE 1979-80

CHAIRMAN

Edward C. Deland
Division of Thoracic Surgery
University of California
Los Angeles, CA 90024
(213) 825-0859

VICE CHAIRMAN (SOUTH)

James E. Mericle
Electronic Associates, Inc.
Suite 100
23860 Hawthorne Blvd.
Torrence, CA 90505
(213) 375-0588

VICE CHAIRMAN (NORTH)

Martin H. Dost
IBM Corporation G-12-141
5600 Cottle Road
San José, CA 95193
(408) 256-2298

SECRETARY

Edward Y. Lambert
Pacific Missile Test Center
896 Calle La Roda
Camarillo, CA 93010
(805) 982-7820

TREASURER

Stefan S. Kruc
The Aerospace Corporation
P.O. Box 92957
Los Angeles, CA 90009
(213) 648-6034

MEMBER-AT-LARGE

Leon Levine
Department of Computer Science
3531 Boelter Hall
University of California
Los Angeles, CA 90024
(213) 825-6735 or 4943

MEMBER-AT-LARGE

Richard R. Hansen
42957 Chicory
Lancaster, CA 93534
(805) 277-4509

MEMBER-AT-LARGE

Hans F. Meissinger
TRW Defense and Space Systems
Space Vehicles Division
One Space Park M/S R5-2031
Redondo Beach, CA 90278
(213) 535-1451

NATIONAL REPRESENTATIVE

Walter J. Karplus
Department of Computer Science
3732 Boelter Hall
University of California
Los Angeles, CA 90024
(213) 825-2929

NATIONAL REPRESENTATIVE

Stefan S. Kruc
The Aerospace Corporation
P.O. Box 92957
Los Angeles, CA 90009
(213) 648-6034

TECHNICAL EDITOR (SOUTH)

Man T. Ung
428 W. Acacia Avenue
El Segundo, CA 90245
(213) 322-0346

TECHNICAL EDITOR (NORTH)

William D. Cameron
N-243-5 NASA Ames Research Center
Moffett Field, CA 94035
(415) 965-5169

SCS MANAGER OF OPERATIONS

Chip G. Stockton
P.O. Box 2228
La Jolla, CA 92038
(714) 259-3888 or
(emergency only) (714) 279-6799

PAST CHAIRMAN

Charles R. Estes
Estes Enterprises
3042 Palo Verde Circle
Camarillo, CA 93010
(213) 889-2532