*Proceedings of the 2005 Winter Simulation Conference*
*M. E. Kuhl, N.M. Steiger, F. B. Armstrong, and J. A. Joines, eds.*

# CONSTRUCTING MULTI-POINT DISCRETE EVENT INTEGRATION SCHEMES

James Nutaro
Oak Ridge National Laboratory
Oak Ridge, TN 37830, U.S.A.

## ABSTRACT

The systematic study of discrete event numerical integration schemes can be greatly aided by an understanding of their general form. This paper describes the structure of DEVS models that can be used to construct multi-point discrete event integration methods. The structure is shown to be sufficient for describing two known methods. The utility of the structure is illustrated by the construction of a new, second order accurate, multi-point discrete event integrator.

## 1 INTRODUCTION

Computer simulation of differential equations requires that some element of the problem be approximated by discrete quantities. There are two dimensions that can be made discrete; time and state. By discretizing time, the differential equation is approximated by a difference equation (i.e., a discrete time system). Here, the solution is calculated at fixed points in time. By discretizing state, the differential equation is approximated by a discrete event system. In this instance, events correspond to jumps through the discrete state space of the approximation.

The essential aspect of a discrete time approximation is that the difference equation maps a discrete time set to a continuous state set. The time discretization of the differential equation need not be regular. It may even be revised in the course of a calculation. Regardless, the elementary features of a discrete time base and continuous state space remain.

The basic aspect of a discrete event approximation is opposite that of a discrete time approximation. The approximating discrete event system is a function from a continuous time set to a discrete state set. The state discretization need not be uniform, and it may even be revised as the computation progresses.

The existence of general forms for discrete time approximations of differential equations has greatly aided the systematic development of that field. A systematic development of discrete state approximations could similarly benefit from an explicit description of form.

This paper discusses a form for multi-point discrete event approximations of ordinary differential equations. The form is restricted to methods that have piecewise constant input and output trajectories, employ only first derivatives, and use a uniform discretization of the state space. None the less, it is general enough to describe known multi-point integration schemes, and it provides a structure through which new methods can be discovered. The latter purpose is illustrated by a new, near second order accurate, integration scheme.

## 2 THE GENERAL FORM OF DISCRETE EVENT INTEGRATORS

The Discrete Event System Specification (DEVS) is the foundation for the development of discrete event integrators. DEVS is a general theory of discrete event systems (Zeigler, Praehofer, and Kim 2000). It has two specific advantages for this application (Nutaro and Sarjoughian 2004). First, it rigorously characterizes weakly causal and multi-input discrete event systems. These properties regularly occur in continuous systems, and they must be dealt with explicitly and consistently in any discrete approximation.

Second, DEVS provides a rigorous characterization of multi-component system structure. Implicit in this characterization is a natural and consistent treatment of encapsulated state space representations and simultaneous events. These are both regular features of continuous systems, and they must be treated precisely in discrete approximations.

Consider a differential equation in the form

$$\dot{x}(t) = f(x(t), u(t)).$$

The trajectory x(t) describes the state of the system, and u(t) is the input to the system. The system input can be vector valued, if this is needed. This system can be described by the two component model shown in Figure 1 (see (Zeigler, Sarjoughian, and Praehofer 2000), (Zeigler, Praehofer, and Kim 2000), and (Kofman 2004) for other descriptions of this decomposition). Systems of equations can be modeled by connecting several of these elementary blocks. For example,

the system

$$\dot{v}(t) = f(v(t), w(t))$$
$$\dot{w}(t) = g(w(t), v(t))$$

is described by the coupled system shown in Figure 2.

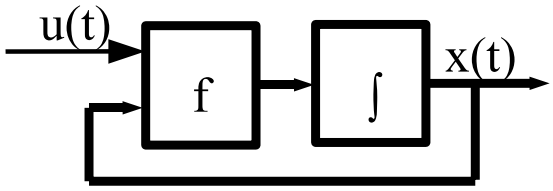


Figure 1: An Ordinary Differential Equation with State Variable x(t) and Input u(t)

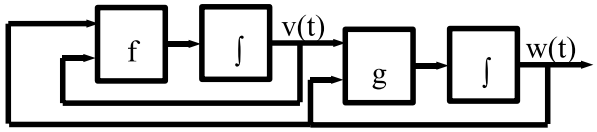


Figure 2: A Pair of Coupled Ordinary Differential Equations

There are two types of dynamic components shown in Figures 1 and 2. These are differential functions (i.e., the functions f and g) and integrators. Both types of components can be described by atomic DEVS models.

### 2.1 Differential Functions

An atomic model of a differential function has one input for each function argument. The state of the model includes one state variable for each function argument, and one a state variable that describes the next event time. This value is set to zero when the model receives an input, and this causes the system to immediately produces its new value as output. Otherwise, the next output time is $\infty$ (i.e., nothing has changed).

Formally, the input, output, and state set for the model are

$$X = \{1, 2, \ldots, n\} \times \mathbb{R},$$
$$Y = \mathbb{R}, \text{ and}$$
$$S = \mathbb{R}^n \times \{0, \infty\}.$$

The value $(i, v_i) \in X$ represents a change to the function argument $x_i$. The state set consists of the last input value for each function argument and the time advance. For brevity, the symbol $\bar{q}$ will be used to represent the model state $(x_1, \ldots, x_n, \sigma)$. Where it is convenient, $\bar{x}$ will be used in place of the state variables $x_1, x_2, \ldots, x_n$.

The initial state of the model has $\bar{x}$ equal to the initial values of its function arguments and $\sigma$ equal to zero. An input event to the model indicates that the value of one of the function arguments has changed. This will cause the model to update its state variables to reflect the new input value. The model immediately outputs the new value of the function. Absence of input indicates that the function arguments have not changed, and so no output is needed. This is formalized in the model output, time advance, and state transition functions

$$ta(\bar{q}) = \sigma,$$
$$\lambda(\bar{q}) = f(\bar{x}),$$
$$\delta_{int}(\bar{q}) = (\bar{x}, \infty),$$
$$\delta_{ext}(\bar{q}, e, x^b) = (\bar{x}', 0) \text{ where}$$
$$\begin{cases} x_i' = v_i, & \text{if } (i, v_i) \in x^b \\ x_i' = x_i, & \text{otherwise} \end{cases}$$
and
$$\delta_{con}(\bar{q}, x^b) = \delta_{ext}(\bar{q}, \sigma, x^b)$$

As an example, consider the single argument function f(x(t)). The model for this function has two state variables; $x_1$ and $\sigma$. The atomic model that describes this function is defined by

$$X = Y = \mathbb{R},$$
$$S = \mathbb{R} \times \{0, \infty\}$$
$$ta((x_1, \sigma)) = \sigma,$$
$$\delta_{int}((x_1, \sigma)) = (x_1, \infty),$$
$$\delta_{ext}((x_1, \sigma), e, x) = (x, 0),$$
$$\delta_{con}((x_1, \sigma), x) = \delta_{ext}((x_1, \sigma), \sigma, x), \text{ and}$$
$$\lambda((x_1, \sigma)) = f(x_1).$$

Because this is a single input/single output system, $x^b$ has been replaced by $x$ in the external and confluent transition functions.

### 2.2 Single Point Discrete Event Integrators

A single point discrete event integrator uses the most recent input (i.e, the most recent approximation of the derivative) to approximate f(x(t)). The integrator is a single input, single output system. Input events represent changes in the value of the derivative (i.e., in f(x(t))). Output events give the value of the integrated function (i.e., x(t)) when it changes by a significant amount. More precisely, an output event occurs whenever x(t) changes by the integration quantum D.

The state set of a single point discrete event integrator is given by

$q_l$, the last output value of the integrator,
$q$, the current value of the integral,

$\dot{q}$, the last known value of the derivative, and
$\sigma$, the time until the next output event.

The integrator's input and output events are real numbers. The value of an input event is the derivative at the time of the event. An output event gives the value of the integral at the time of the output.

The discrete event integrator generates output events when the integral of the input changes by D. More generally, if $\Delta q$ is the desired change, [$t_0$,T] is the interval in which the change occurs, and f(x(t)) is the first derivative of the system, then

$$\int_0^T f(x(t_0 + t))\, dt = F(T) = \Delta q. \tag{1}$$

The function F(T) gives the *exact* change in x(t) over the interval [$t_0$,T]. Equation (1) is used in two ways. If F(T) and $\Delta q$ are known, then the time advance of the discrete event integrator is found by solving for T. If F(T) and T are known, then the next state of the integrator is given by q+F(T), where T is equal to the elapsed time (in the external transition function) or time advance (in the internal transition function).

A single point scheme approximates f(x(t)) with a piecewise constant function. At any particular time, the value of the approximation is given by the state variable $\dot{q}$. Using $\dot{q}$ in place of $f(x(t_0 + T))$ in equation (1) gives

$$\int_0^T \dot{q}\, dt = \dot{q}T.$$

When $\dot{q}$ and T are known, then the function

$$\hat{F}(T, \dot{q}) = \dot{q}T \tag{2}$$

approximates F(T). Because T must be positive (i.e., we are simulating forward in time), the exact inverse of equation (2) can not be used to compute the time advance. However, the absolute value of the inverse,

$$\hat{F}^{-1}(\Delta q, \dot{q}) = \begin{cases} \frac{\Delta q}{|\dot{q}|} & \text{if } \dot{q} \neq 0 \\ \infty & \text{otherwise} \end{cases} \tag{3}$$

is suitable for this purpose.

The state transition, output, and time advance functions of the single point discrete event integrator can be defined in terms of equations (2) and (3). This gives

$$\begin{aligned}
&\delta_{int}((q_l, q, \dot{q}, \sigma)) = \\
&\quad (q + \hat{F}(\sigma, \dot{q}), q + \hat{F}(\sigma, \dot{q}), \dot{q}, \hat{F}^{-1}(D, \dot{q})), \\
&\delta_{ext}((q_l, q, \dot{q}, \sigma), e, x) = \\
&\quad (q_l, q + \hat{F}(e, \dot{q}), x, \hat{F}^{-1}(D - |q + \hat{F}(e, \dot{q}) - q_l|, x)), \\
&\delta_{con}((q_l, q, \dot{q}, \sigma), x) = \\
&\quad (q + \hat{F}(\sigma, \dot{q}), q + \hat{F}(\sigma, \dot{q}), x, \hat{F}^{-1}(D, x)), \\
&\lambda((q_l, q, \dot{q}, \sigma)) = q + \hat{F}(\sigma, \dot{q}), \quad \text{and} \\
&ta((q_l, q, \dot{q}, \sigma)) = \sigma.
\end{aligned}$$

With this definition, $\hat{F}$ computes the next value of the integral using the previous value, the approximation of f(x(t)) (i.e., $\dot{q}$), and the time elapsed since the last state transition. The time that will be needed for the integral to change by an amount D is computing using $\hat{F}^{-1}$. The arguments to $\hat{F}^{-1}$ are the distance remaining (i.e., D minus the distance already traveled) and the speed with which the distance is being covered (i.e., the approximation of f(x(t))).

Some general properties of this single point scheme are described in (Zeigler, Sarjoughian, and Praehofer 2000), (Kofman 2004), and (Nutaro 2003). The definition of the time advance function given here corresponds to the definition used in (Kofman 2004) when the hysteresis is set to D. The global error for this method is proportional to D, and the method is exact when x(t) is a line (i.e., when f(x(t)) is a constant).

### 2.3 Two Point Discrete Event Integration Schemes

If the function f(x(t)) in equation (1) is approximated using the previous two values of the derivative, then the method is called a two point scheme. Two point schemes require the state variables

$q$, $q_l$, and $\sigma$, just as before,
$\dot{q}_1$ and $\dot{q}_0$, the last two values of the derivative, and, possibly,
h, the time interval between $\dot{q}_1$ and $\dot{q}_0$.

Two two point methods are considered here. The first method approximates f(x(t)) in equation (1) with the line connecting the points $\dot{q}_1$ and $\dot{q}_0$. The distance moved by x(t) in the interval [h,h+T] can be approximated by

$$\int_h^{h+T} \frac{\dot{q}_1 - \dot{q}_0}{h} + \dot{q}_0\, dt = \frac{\dot{q}_1 - \dot{q}_0}{2h} T^2 + \dot{q}_1 T = \Delta q.$$

The functions

$$\hat{F}_1(T, \dot{q}_1, \dot{q}_0, h) = \frac{\dot{q}_1 - \dot{q}_0}{2h} T^2 + \dot{q}_1 T, \quad (4)$$

and

$$\hat{F}_1^{-1}(\Delta q, \dot{q}_1, \dot{q}_0, h) = \Delta T, \quad (5)$$

where $\Delta T$ is the smallest positive root of

$$|\frac{\dot{q}_1 - \dot{q}_0}{2h} T^2 + \dot{q}_1 T| = \Delta q$$

and $\infty$ if such a root does not exist, can be used to define the state transition, output, and time advance functions (which will be done in a moment). Equations (4) and (5) are exact when x(t) is a quadratic.

Another approximation to f(x(t)) can be obtained with the piecewise constant function

$$a\dot{q}_1 + b\dot{q}_0, \quad a + b = 1. \quad (6)$$

If $x(t)$ is the line $mt + b$, then $f(x(t)) = m$, $(am + bm) = (a + b)m = m$, and so this approximation is exact. Integrating equation (6) over the interval [0,T] gives the approximating functions

$$\hat{F}_2(T, \dot{q}_1, \dot{q}_0) = (a\dot{q}_1 + b\dot{q}_0)T, \text{ and} \quad (7)$$

$$\hat{F}_2^{-1}(\Delta q, \dot{q}_1, \dot{q}_0) = \frac{\Delta q}{|a\dot{q}_1 + b\dot{q}_0|}. \quad (8)$$

This approximation does not require the state variable h. This scheme is discussed in detail in (Nutaro 2003).

For brevity, let $\bar{q}$ denote the state of the integrator, and let $\bar{dq}$ denote the variables $\dot{q}_1, \dot{q}_0$ or $\dot{q}_1, \dot{q}_0, h$ as needed. Which is intended will be clear from the context in which it is used. The time advance function for a two point schemes is given by

$$ta(\bar{q}) = \sigma,$$

and the output function is defined by

$$\lambda(\bar{q}) = \hat{F}(\sigma, \bar{dq}).$$

If equations (4) and (5) are used to define the integration scheme, then the resulting state transition functions are

$$\delta_{int}(\bar{q}) = (q + \hat{F}_1(\sigma, \bar{dq}), q + \hat{F}_1(\sigma, \bar{dq}), q_1, q_1, \sigma, \hat{F}_1^{-1}(D, \dot{q}_1, \dot{q}_1, \sigma)),$$

$$\delta_{ext}(\bar{q}, e, x) = (q_l, q + \hat{F}_1(e, \bar{dq}), x, q_1, e, \hat{F}_1^{-1}(D - |q + \hat{F}_1(e, \bar{dq}) - q_l|, x, \dot{q}_1, e)), \text{ and}$$

$$\delta_{con}(\bar{q}, x) = (q + \hat{F}_1(\sigma, \bar{dq}), q + \hat{F}_1(\sigma, \bar{dq}), x, q_1, \sigma, \hat{F}_1^{-1}(D, x, \dot{q}_1, \sigma)).$$

When equations (7) and (8) are used to define the integrator, then the state transition functions are

$$\delta_{int}(\bar{q}) = (q + \hat{F}_2(\sigma, \bar{dq}), q + \hat{F}_2(\sigma, \bar{dq}), q_1, q_1, \hat{F}_2^{-1}(D, \dot{q}_1, \dot{q}_1)),$$

$$\delta_{ext}(\bar{q}, e, x) = (q_l, q + \hat{F}_2(e, \bar{dq}), x, q_1, \hat{F}_2^{-1}(|q + \hat{F}_2(e, \bar{dq}) - q_l| - D, x, \dot{q}_1)), \text{ and}$$

$$\delta_{con}(\bar{q}, x) = (q + \hat{F}_2(\sigma, \bar{dq}), q + \hat{F}_2(\sigma, \bar{dq}), x, q_1, \hat{F}_2^{-1}(D, x, \dot{q}_1)).$$

The scheme that is constructed using equations (4) and (5) is similar to the QSS2 method in (Kofman 2004), except that the input and output trajectories used here are piecewise constant rather than piecewise linear.

The scheme constructed from equations (7) and (8) is nearly second order accurate when a and b are chosen correctly. If we select $a = \frac{3}{2}$ and $b = -\frac{1}{2}$, then the error in the integral of 6 is

$$E = (f(x_1) - \frac{3f(x_1)}{2} + \frac{f(x_0)}{2})T + \frac{1}{2}T^2 \frac{d}{dt} f(x_1) + \sum_{n=3}^{\infty} \frac{1}{n!} \frac{d}{dt}^{(n+1)} f(x_1) T^n. \quad (9)$$

To make this nearly second order accurate, we want the terms that depend on T and $T^2$ to be as small as possible. Let h be the time separating $x_1$ and $x_0$ (i.e., $x_1 = x(t_1)$ and $x_0 = x(t_0)$ and $h = t_1 - t_0$), and let $\alpha = \frac{T}{h}$, the ratio of the current time advance to the previous time advance. It follows that $T = \alpha h$. The function $\frac{d}{dt} f(x_1)$ can be approximated by

$$\frac{d}{dt} f(x_1) \approx \frac{f(x_1) - f(x_0)}{h}. \quad (10)$$

Substituting equation (10) into equation (9) and dropping the high order error terms gives

$$E \approx \alpha h (\frac{f(x_1) - f(x_0)}{2} + \alpha \frac{f(x_0) - f(x_1)}{2}). \quad (11)$$

Equation (11) approaches zero as $\alpha$ approaches 1. It seems reasonable to assume T and h become increasingly similar as D is made smaller. From this assumption, it follows that the low order error terms in equation (9) vanish as D shrinks.

Figure 3 shows the absolute error in the computed solution of $\dot{x} = -x$, x(0)=1 as a function of D. The simulation was ended at t=1.0, and $\alpha$ and absolute error were recorded at that time. Figure 4 shows $\alpha$ as a function of D, and the plot clearly shows that $\alpha$ approaches 1 as D becomes small. The absolute error is proportional to $D^2$. This agrees nicely with the hypothesis that, as $\alpha$ goes to unity, the low order error terms in equation (9) to drop out.

The particular choices of $a = \frac{3}{2}$ and $b = -\frac{1}{2}$ are critical for obtaining second order errors. To illustrate this, Figure 5 shows the error, as a function of D, for several possible choices of a and b. Inserting these choices into equation (11) clearly shows that the low order error terms remain while $h > 0$. This is reflected in the Figure 5, where the error is seen to be linear in D.

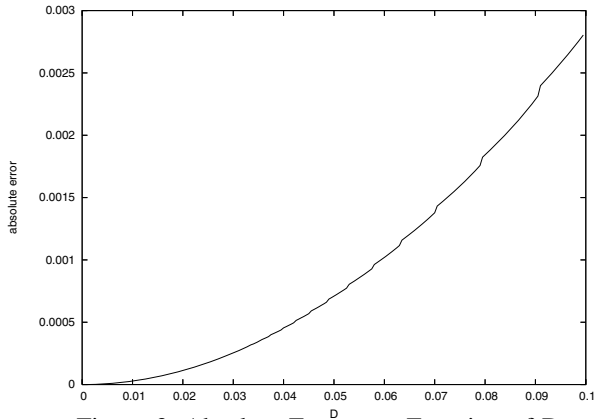


Figure 3: Absolute Error as a Function of D

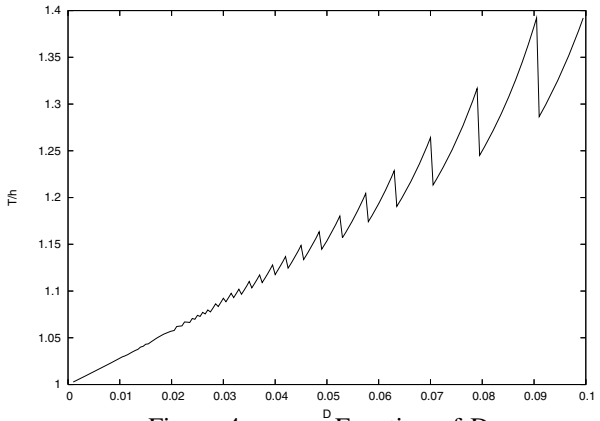


Figure 4: $\alpha$ as a Function of D

It is worth noting that, when the time step is fixed, equation (7), with a = $\frac{3}{2}$ and b = $-\frac{1}{2}$, is the second order Adams-Bashforth method (see, e.g., (Ralston and Rabinowitz 1978)). The second order Adams-Bashforth scheme is obtained by integrating the line connecting $f(x(t_1))$ and $f(x(t_0))$ when $t_1 - t_0 = h$. The coefficients a = $\frac{3}{2}$ and b = $-\frac{1}{2}$ emerge naturally from the integral when the integration interval is [h,2h].

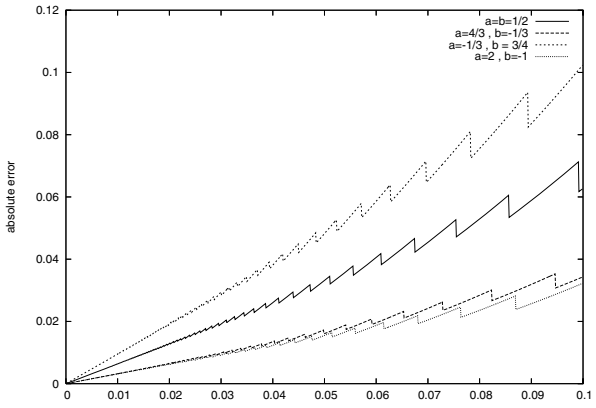


Figure 5: Error as a Function of D for Different a and b

This approach to deriving higher order schemes is interesting because it does not rely on finding the roots of a polynomial. While it is, in principle, possible to create third and even fourth order methods that rely on finding the roots of a polynomial (see, e.g., (Kofman 2005a) and (Kofman 2005b)), this approach must fail for polynomials of degree five. Moreover, the computational cost of the root finding problem may out pace the savings from reduced quantum sizes when higher order methods of this type are used. If, instead, it is possible to construct third, fourth, or even higher order schemes that are similar to equations (7) and (8), then the root finding problem can be entirely avoided.

### 2.4 General Multi-point Schemes

The one and two point schemes can be easily generalized to describe methods that use an arbitrary number of past derivative values. The critical element of these schemes are the approximations $\hat{F}$ and $\hat{F}^{-1}$. The function $\hat{F}$ describes how the state variables $q$ and $q_l$ evolve through time. The function $\hat{F}^{-1}$ predicts how long the system, evolving as described by $\hat{F}$ and assuming no change in the derivative, will need to move a particular distance. The fact that time must move forward prevents $\hat{F}^{-1}$ from being a true inverse of $\hat{F}$. Instead, $\hat{F}$ and $\hat{F}^{-1}$ must satisfy the somewhat weaker relationship

$$\hat{F}(\hat{F}^{-1}(\Delta q, \bar{dq}), \bar{dq}) = \Delta q \; sgn(\hat{F}(\hat{F}^{-1}(\Delta q, \bar{dq})), \bar{dq}),$$

where

$$sgn(x) = \begin{cases} 1 & \text{if } x > 0 \\ -1 & \text{if } x < 0 \\ 0 & \text{if } x = 0 \end{cases} .$$

It can be readily verified that equations (2) and (3), (4) and (5), and (7) and (8) satisfy this relationship.

Errors produced by integration scheme in this form can be seen from two different points of view. On the one hand, there is the error in the state of the computed system at a particular time. This point of view is shown in Figure 3. Classical methods in error analysis prefer this viewpoint, and it amounts to asking how well $\hat{F}$ satisfies 1 when T is known.

The second point of view focuses on how well $\hat{F}^{-1}$ predicts when the system will travel a given distance. More formally, we ask how well $\hat{F}^{-1}$ predicts a T to satisfy equation (1) when $\Delta q$ is fixed. From this vantage point, there is an error in the time at which the computed system enters a particular state. Figure 6 shows the absolute value of the difference between the computed and actual times at which the solution to $\dot{x} = -x$, x(0) = 1, is closest to $e^{-1}$. The calculation was performed using equations (7) and (8). The general shape of the curve is a quadratic in D, just as in Figure 3.

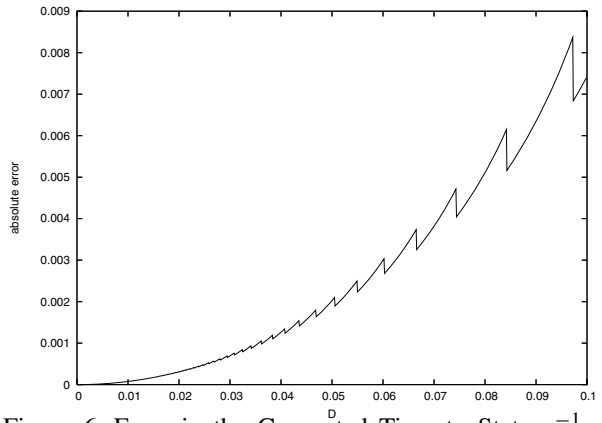


Figure 6: Error in the Computed Time to State $e^{-1}$ as a Function of D.

## 3 CONCLUSIONS

The functions $\hat{F}$ and $\hat{F}^{-1}$ are the essential elements of this form for discrete event integrators. Here, too, is the difficulty in analyzing discrete event integration schemes. $\hat{F}$ can be constructed by integrating interpolating polynomials, truncating Taylor series, or any number of well known methods (an excellent introduction can be found in (Ralston and Rabinowitz 1978)). The construction of $\hat{F}^{-1}$ has not been studied extensively, and its interaction with the approximation $F$ to produce truncation and global errors is not, in general, well understood.

The form presented here is restricted to schemes that have piecewise constant input/output trajectories, rely only on first derivatives, and use a fixed integration quantum size. Schemes using higher order derivatives are suggested, but not fully developed, in (Kofman 2003). A method using piecewise polynomial input and output trajectories is described in (Kofman 2004) and (Wainer and Giambiasi 2005). An adaptive quantization scheme is presented in (Bolduc and Vangheluwe 2003).

In spite of the restrictions on the form presented in this paper, it is hoped that the explicit exposition of form will elicit broader interest in discrete event numerical methods. The identification of the functions $\hat{F}$, $\hat{F}^{-1}$, and their relationship as critical unknowns should help to focus research efforts, and to promote discourse between researchers working with discrete event systems and those working with classical numerical methods.

## REFERENCES

Bolduc, J.-S., and H. Vangheluwe. 2003, July. Mapping odes to devs: Adaptive quantization. In *Proceedings of the 2003 Summer Simulation MultiConference (SCSC'03)*, 401–407. Montreál, Canada.

Kofman, E. 2003. *Discrete event based simulation and control of continuous systems*. Ph. D. thesis, Universidad Nacional de Rosario, Rosario, Argentina.

Kofman, E. 2004. Discrete event simulation of hybrid systems. *SIAM Journal on Scientific Computing* 25 (5): 1771–1797.

Kofman, E. 2005a. A third order discrete event method for continuous system simulation. part i: Theory. Technical Report LSD0501, School of Electronic Engineering, Universidad Nacional de Rosario, Rosario, Argentina.

Kofman, E. 2005b. A third order discrete event method for continuous system simulation. part ii: Applications. Technical Report LSD0502, School of Electronic Engineering, Universidad Nacional de Rosario, Rosario, Argentina.

Nutaro, J. 2003. *Parallel discrete event simulation with application to continuous systems*. Ph. D. thesis, University of Arizona, Tuscon, Arizona.

Nutaro, J., and H. Sarjoughian. 2004. Design of distributed simulation environments: A unified system-theoretic and logical processes approach. *SIMULATION* 80 (11): 577–589.

Ralston, A., and P. Rabinowitz. 1978. *A first course in numerical analysis, second edition*. Mineola, New York: Dover Publications.

Wainer, G. A., and N. Giambiasi. 2005, February. Cell-devs/gdevs for complex continuous systems. *SIMULATION* 81 (2): 137–151.

Zeigler, B. P., H. Praehofer, and T. G. Kim. 2000. *Theory of modeling and simulation, 2nd edition*. Academic Press.

Zeigler, B. P., H. Sarjoughian, and H. Praehofer. 2000, September. Theory of quantized systems: Devs simulation of perceiving agents. *Cybernetics and Systems* 31 (6): 611–647.

## AUTHOR BIOGRAPHY

**JAMES NUTARO** is employed at the Oak Ridge National Laboratory in Oak Ridge, TN. His e-mail address is <nutarojj@ornl.gov>.