Implications of Finite Time Stepping for Transient Sensor Models

Martin Schubert, Beatriz González

FH Regensburg*, Seybothstr. 2, D-93053 Regensburg, Germany, e-mail: martin.schubert@e-technik.fh-regensburg.de

ABSTRACT

Modeling of sensor devices has to take into account not only variable voltages and currents but also variable device parameters. This makes transient differentiation and integration more complex than for circuits with constant device characteristics. The problem is increased when behavioral models allow for unrealistic idealizations such as discontinuous functions. To obtain convergent solutions that make sense we must understand the little differences between what we want to do, what the software can do, what it will do and what we have simulated at the end.

Keywords: Simulation, Shape Functions, Charge Conservation, Finite Time Stepping, Analog Circuit Modeling.

I. INTRODUCTION

During the development of a VHDL based mixed signal simulator [1] the authors had the chance to see the problem of analog and mixed signal simulation from three points of view: (1) HDL specification, (2) software development and (3) software application. This communication intends to give the user insight in those problems of software development, that might have effect his modeling work.

A distinction between micro modeling and macro modeling is made by assuming that micro modeling is done by the software developer and macro modeling by the user of the tool. It was found to be advantageous when both sides had some understanding of the problems, goals and limitations of the other.

Chapter II illustrates difficulties arising from the transition of the infinitesimal expression $s'(t) = ds/dt$ to it's finite differences approximation $s'(t) \equiv (s(t+h) - s(t))/h$. As market emphasis is on capacitive rather than inductive sensors, test setup for simulation tools are proposed for circuits containing capacitors. Transfer to inductors is straight forward.

Chapter III illustrates limitations of time step selection and Chapter IV lists some desirable features of macro model behavioral waveforms.

* University of Applied Sciences.

II. COMBINING MICRO AND MACRO MODELS

A. Shape Functions

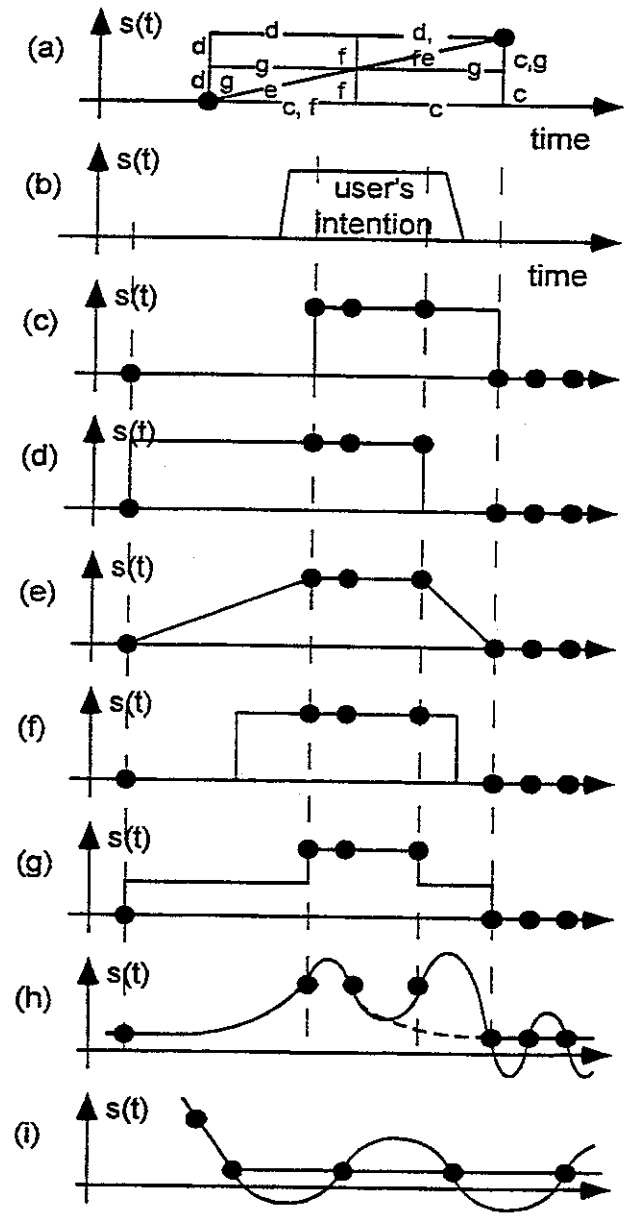

Figure 1: Different shape functions connecting discrete points

The representation of an arbitrary function on a computer is done with a finite number of discrete points. To define the function between these points an agreement regarding interpolation must be found. This is typically done using shape functions.

Fig. 1(a) shows five possibilities, labeled (c-g), to realize the transition between two timepoints using straight lines. Fig. 1(b) shows a user defined quantity, (e.g. voltage, current, resistor, capacitor, inductor, etc.). The simulator samples the waveform of Fig. 1(b) by evaluating the function for a set of time points. The selection of these points is due to the global situation. Figs. 1(c-g) show the shape of simulated pulses according to the transition schemes shown in Fig. 1(a). Obviously, there are differences between the user's intention and the simulated waveforms. Fig. 1(h) uses a second order polynomial to yield a continuous first derivative. The oscillation illustrates that more continuity does not guarantee more accuracy. It can accidentally deliver acceptable (dashed line) or unacceptable results (solid line). In Fig. 1(i) oscillation is caused by a discontinuous first derivative. Non-linear polynomials carry the risk of oscillation and so the risk of negative integrals for some time steps as shown in (h,i), which may be physically impossible.

A discontinuity can hardly be modeled with a continuous function. As derivatives are also functions, the order of continuity of a macro model (waveform) defined by the user should be equal to or larger than the shape function's order of continuity. This suggests simple, linear shape functions.

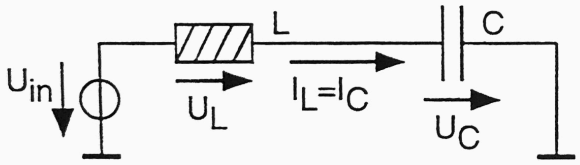

Figure 2: A simple circuit but difficult for shape functions.

To find a physically consistent shape function consider the LC circuit in Fig. 2. Let L, C be constant and make the following assumptions:

1. Voltages U(t) are modeled using n^{th} order polynomials.
2. As the current through the inductor, $I_L(t)$, is obtained by integration of $U_L(t)$, polynomials of order (n+1) are required for current shape functions, $I_L(t)$.
3. As the voltage across the capacitor, $U_C(t)$, is obtained by integration of $I_C(t)=I_L(t)$, polynomials of order (n+2) are required for the voltage shape functions, U(t).

Obviously, points 1 and 3 are contradictory for a finite order n. Could an exponential shape function be a solution? It is difficult to handle, has problems with constant slopes, requires continuity conditions, etc. These arguments are contradictory to the need for simple, linear shape functions. These problems arise while modeling primitive situations.

Analog circuit simulation tools integrate a signal $s(t)$ represented by a number of discrete points $s_n = s(t_n)$ either with the linear backward Euler integration formula

$$s_{n+1} = s_n + h \cdot \dot{s}_{n+1} \tag{1}$$

or with the second order trapezoidal model

$$s_{n+1} = s_n + h \cdot (\dot{s}_{n+1} + \dot{s}_n) / 2 \tag{2}$$

Both models are stable for any timestep $h > 0$ [2]. The trapezoidal model uses the signal derivative of the last time point, $\dot{s}_n$. So it assumes a continuous first derivative.

B. Charge Conservation for Variable Capacitors

Sensors typically transduce a variable physical quantity into a variable electronic device parameter such as resistance, capacitance or inductance. The simulation of time dependent electronic component characteristics is more complicated than computing constant components. Fundamental laws of electricity deliver $U_R = R \cdot I_R$ for resistors, $Q_C = C \cdot U_C$ for capacitors and $\Phi_L = L \cdot I_L$ for inductors, where U, I, Q and Φ represent voltage, current, charge and magnetic flux, respectively. Simple differentiation delivers

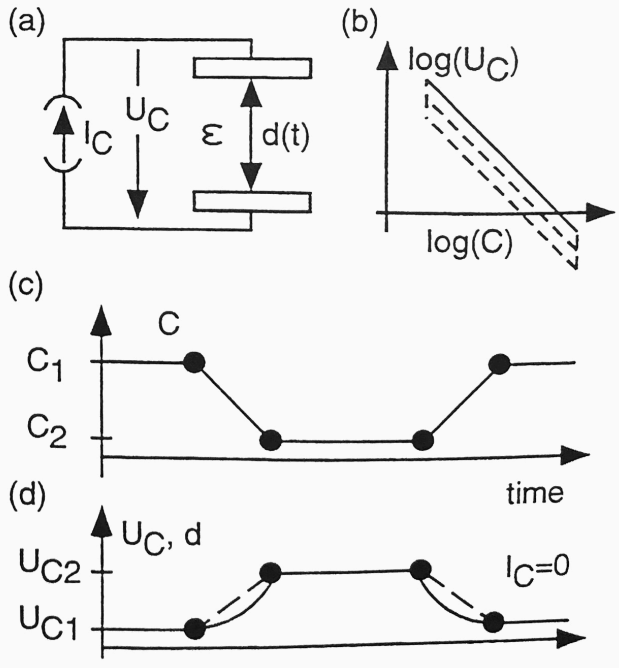

Figure 3: **(a)** Variable capacitor, **(b)** $\log(U_C)$ vs. $\log(C)$ with $I_C = 0$, **(c)** $C(t)$ and **(d)** $U_C(t)$ or distance $d(t)$.

$$\frac{dU_R}{dt}=\frac{d}{dt}(I_R R)=I_R\frac{dR}{dt}+R\frac{dI_R}{dt} \tag{3}$$

$$I_C=\frac{dQ_C}{dt}=\frac{d}{dt}(C\cdot U_C)=C\frac{dU_C}{dt}+U_C\frac{dC}{dt} \tag{4}$$

$$U_L=\frac{d\Phi_L}{dt}=\frac{d}{dt}(L\cdot I_L)=L\frac{dI_L}{dt}+I_L\frac{dL}{dt}. \tag{5}$$

Consider the capacitive sensor in Fig. 3(a). Charge it using I_C and then set $I_C=0$ to obtain constant charge $Q_C=CU_C$. The logarithm of this formula delivers

$$\log(U_C(t))=\log(Q_C)-\log(C(t)). \tag{6}$$

With constant charge Q_C and variable capacitance $C(t)$ the above equation produces a line as indicated by the solid line in Fig. 3(b). In cases where charge is being lost, the characteristic is indicated by the field denoted by dashed lines in Fig. 3(b).

Charge conservation is a difficult task for the simulation of capacitive sensor devices. Discretization of Eq. (4) delivers the finite differences expression

$$I_C\equiv C_{eff}\frac{\Delta U_C}{\Delta t}+U_{C,eff}\frac{\Delta C}{\Delta t} \tag{7}$$

creating the problem of proper selection of C_{eff}, U_{eff}. Applying the boundary condition $I_C=0$ we find

$$\Delta U_C=-\frac{U_{C,eff}}{C_{eff}}\Delta C. \tag{8}$$

Due to the conservation of charge, stepping the value of C back and forth by ΔC results in a corresponding variation in voltage as shown in Fig. 3(c). Therefore, the error:

$$\Delta U_{C,err}=\left(\frac{U_{C,eff,2}}{C_{eff,2}}-\frac{U_{C,eff,1}}{C_{eff,1}}\right)\Delta C. \tag{9}$$

has to be minimized. This requires in the specific situation of Fig. 3(c) that

$$\frac{U_{C,eff,2}}{C_{eff,2}}=\frac{U_{C,eff,1}}{C_{eff,1}}. \tag{10}$$

However, if C returns back to the same value after several steps, such an easy model as Eq. (10) is no more possible.

To test a simulator we charge a Capacitor C_{sens} modeled as

$$C_{sens}=C_0(1-\varepsilon(t)) \tag{11}$$

to a constant, non-zero charge and apply a transient $\varepsilon(t)$. The condition $C_{sens}>0$ must be fulfilled. For symmetry reasons, symmetric waveforms may cause error cancellation. A model that forces any simulator to its limits will result in oscillating capacitor values according to Fig. 4. A sinusoidal signal is weighted with a hyperbolic tangent step.

$$\varepsilon(t)=\frac{1}{2}\left[1+\tanh\frac{t-t_o}{\tau}\right]\cdot\sin(\omega(t)\cdot t). \tag{12}$$

When $\varepsilon(t)\to 1$ then $C_{sens}(t)\to 0$ and therefore $U_C(t)=Q_C/C_{sens}(t)\to\infty$. This situation is difficult to simulate while conserving charge. To avoid error cancellation due to symmetries, the sinusoidal signal in Fig. 4 is modulated such that the falling edge is three time faster than the rising edge.

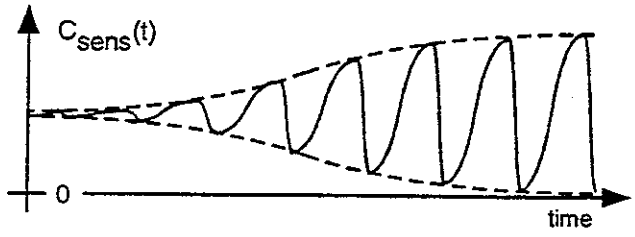

Figure 4: oscillating capacitor approaching zero to test the simulator's charge conservation capabilities.

C. Accuracy Considerations

To simulate accuracy, resolution, linearity, etc. of a sensor, we should first know the respective limitations of our tool.

Accuracy is difficult to define. For a signal $s(t)$ oscillating with amplitude A around an offset B according to

$$s(t)=A\sin(\omega t)+B \tag{13}$$

a relative accuracy can be defined as

$$c_R=\frac{|A|}{|A|+|B|}\le relacc. \tag{14}$$

Especially problematic are oscillations around $B=0$. This situation is expected for $U_L(t)$ in Fig. 5(b). In this case

Eq. (14) will always indicate 100% inaccuracy. Oscillations due to truncation noise can hold a solver in an endless loop when trying to fulfill a given relative accuracy criterion. For this case, an additional absolute accuracy requirement

$$c_A = |A| \le absacc \tag{15}$$

must be fulfilled. To check this two accuracy criteria, the Gaussian weight

$$G(t,t_0,\sigma) = \exp\left(\frac{(t-t_0)^2}{2\sigma^2}\right) \tag{16}$$

is applied as a window function to a sinusoidal oscillation according to

$$s(t) = 1V \cdot \left[G(t,t_0,\sigma)\cdot \sin(\omega t) + B\right]. \tag{17}$$

In Fig. 5(a) horizontal lines indicate the amplitude below which U_C or U_L do not follow the input signal any more because one of the accuracy criteria masks their activity.

Applying $s(t)$ according to Eq. (17) with $B=1$ to the circuit in Fig. 5(b), $U_L(t)$ and $U_C(t)$ oscillate around 0 and $1V$, respectively. Then, U_L, U_C allow one to observe when a signal is considered to be constant due to the absolute and the relative accuracy criterion, respectively. $RC = \omega^{-1} = R/L$ is recommended.

If the above mentioned data about accuracy criteria is unknown, then it may be assumed that a sensor does not work while in fact an accuracy criterion masks its activity.

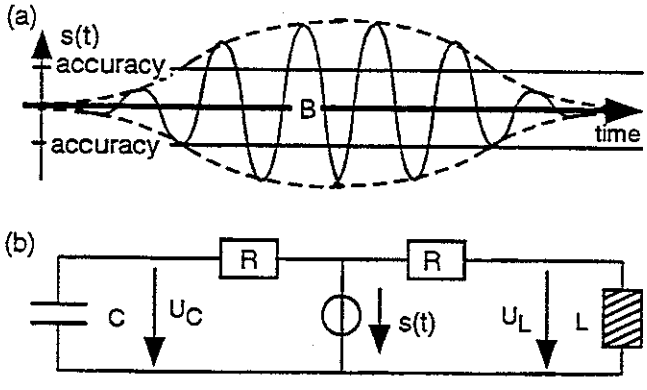

Figure 5: (a) Gaussian shaped sinusoidal waveform with offset and indicated bandwidth of accuracy tolerance. (b) With $B=1V$, the voltages U_L, U_C allow one to check the absolute and relative accuracy, respectively.

The truncation noise in the simulator can be seen from the maximum amplitude of a signal $s(t)$ according to Eq. (17) with $B=1$, while another signal $s_2(t) = s(t) - 1V$ remains $s_2(t) = 0$. Typical real number precision are some 7 or 15 decimal places, depending on the compiler.

The minimum absolute signal value that can be represented on a given simulator can be observed with a signal $s(t)$ according to Eq. (17) with $B=0$. The Gaussian weight will suppress any signal activity after some time. Typical absolute real minima are some 10^{-38} or 10^{-308}.

Table 1 gives the damping effect of a Gaussian weight relative to its maximum in standard deviations (σ) from the maximum. Checking for absolute resolution we should keep in mind that some simulators can represent numbers down to 10^{-308} (i.e. 37.6σ) while other simulations allow for numbers down to down to 10^{-38} (i.e. 13.2σ). When computing relative accuracies we see from table 1 that a mantissa length of 15 or 7 decimals produces truncation noise in the order of 8.3σ or 5.7σ, respectively.

Table 1: Damping effect of a Gaussian weight

$\left\|\frac{x-x_0}{\sigma}\right\|$	$G(x,x_0,\sigma) = \exp\left(\frac{(x-x_0)^2}{2\sigma^2}\right)$
0	1
1	0.6065
5.7	10^{-7}
8.3	10^{-15}
13.2	10^{-38}
37.6	10^{-308}

D. Hysteresis

Fig. 6 shows U_L, U_C following $s(t)$ of Fig. 5(b) stepping from $0 \rightarrow 1V$ and from $2V \rightarrow 1V$. An accuracy criterion that indicates convergence dependent on a sufficiently small step ΔU_C may cause hysteresis. The effect is increased when a soft step is generated by a dependent source $s((U_x(t))$, where $U_x(t)$ can be the output of an RC circuit.

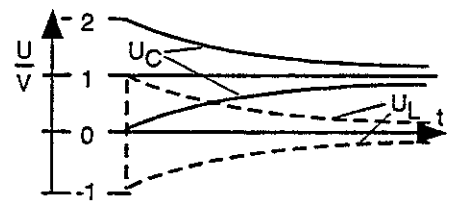

Figure 6: U_L, U_C following input voltage steps with hysteresis.

III. Finite Time Stepping

A. Numerical Efficiency Considerations

Numerical efficiency depends strongly on appropriate time stepping. Larger time steps are, together with accuracy and stability, important reasons to use backward instead of forward Euler integration, even though the numerical effort of solving implicit equations in any time point is significantly higher.

More versatile time stepping is also one of the driving forces for the development of simulation techniques like block iteration, waveform relaxation or event driven simulation. To achieve numerical efficiency any simulator must try to solve a problem within a given accuracy using a minimum number of time steps.

The computation of time steps depends typically on slope and/or bending of a curve, i.e. on its first and/or second order derivative. Therefore, a constant signal with $s'=s''=0$ is difficult for time step computation.

B. Directly Controlled Waveforms

Some simulators allow for a specific set of functions to be used as stimuli. In this case omission of events as illustrated in Fig. 7 can be excluded because the simulator has the information about critical time points.

If the user can define an arbitrary function which has to be sampled by the simulator, a time-step prediction model will return a maximum time step if all derivatives are zero or very close to zero as illustrated in Fig. 7(a). If a large time step reaches over the edge of a stimuli signal, e.g. in the interval $h_3 = t_3 - t_2$ in Fig 7(a), the simulator can realize that something has happened. It can locate the event more exactly for example by interval division. However, much computational power is needed to distinguish a small discontinuity as shown in time point t_x in Fig. 7(a) from a fast edge. In such cases it is helpful when the user can force the simulator to compute solutions in specific time points.

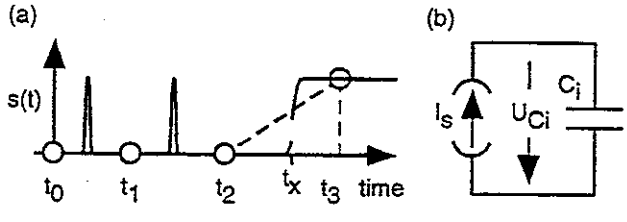

Figure 7: **(a)** Time stepping from constant signal values is difficult. A step function can always be recognized. Spikes my be omitted. **(b)** Test circuit to integrate current spikes.

Even more problematic are thin spikes as swallowed by the intervals $h_1 = t_1 - t_0$ and $h_2 = t_2 - t_1$ in Fig. 7(a). The simulator has no chance to detect the omission of the pulses when only the data computed in the time points t_0, t_1, t_2 is available. If the user cannot give hints, the maximum time step must be chosen small enough to detect the pulses. An example of such pulses are current spikes coming from photo diodes receiving very short laser pulses.

A simulator's reaction in the situation described in Fig. 7(a) can be tested by integrating current pulses into a capacitor as illustrated in Fig. 7(b). An appropriate test function for the current spike is the derived hyperbolic tangent pulse

$$I_S(t) = \frac{I_{S0}}{\tau}\left(1 - \tanh^2 \frac{t - t_0}{\tau}\right). \tag{18}$$

This pulse, similar to the Gaussian curve, has an infinite number of derivatives and a known closed form analytical solution:

$$U_{Ci}(t) = \int_{-\infty}^{t} \frac{I_S(t')}{C_i} dt' = \frac{I_{S0}}{C_i}\left(1 + \tanh \frac{t - t_0}{\tau}\right) \tag{19}$$

To check the relative accuracy of the integration of the current spike in Fig. 7(b) the difference between the numerical (index 'num') and analytical (index 'ana') solution is computed:

$$E_{relacc} = \frac{U_{Ci,ana}(t) - U_{Ci,num}(t)}{U_{Ci,ana}(t)}. \tag{20}$$

The simulation time points of the problematic events discussed in this chapter are foreseeable. If the user has the chance to give information to the simulator about critical time points, the solver's task can be significantly facilitated.

C. Indirectly Controlled Waveforms

Also a simulator that has a limited set of available stimuli functions and that allows the user to give hints about crucial time points can be operated in situations which are difficult for a time stepper to handle.

Fig 8(a) shows a circuit frequently used to measure micro machined capacitors. The current I_{ref} begins to charge a capacitor C_S while the MOSFET M has a high input impedance. The linear rise of the voltage U_{Cs} across the capacitor provokes the simulator to use large time steps. When U_{Cs} exceeds the reference voltage U_{ref} the operational amplifier triggers the monostable MF that switches the MOSFET to a low impedant state. In this time

point the time constant of node n_1 changes rapidly and the capacitor is discharged to a near zero voltage. In the ideal case $\Delta t = t_2 - t_1 = 0$ would be expected in Fig. 8(c). Care must be taken to distinguish between that part of Δt that results from device delay and the other part that results from the time stepper's inaccuracy. Let t_{0last} be the time point of the monostable's last falling edge. The noise in $t_{rise} = t_2 - t_{0last}$ originating from the time stepper's inaccuracy can be made visible by plotting U_{Cs} versus $t^* = t - t_{0last}$.

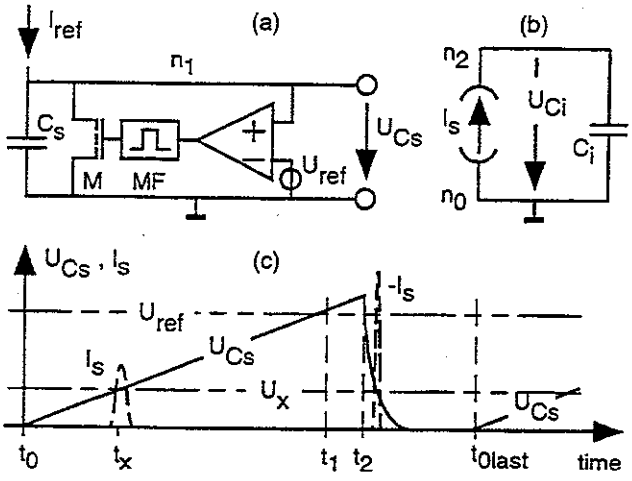

Figure 8: (**a**) Oscillator appended by (**b**) a voltage controlled current source. (**c**) Voltages and current vs. time.

Although the kink in t_2 is difficult to determine exactly, a valid simulator will always note that an event has occurred. More difficult to find is the current peak in $t = t_x$ which is modeled again using a derived hyperbolic tangent pulse:

$$I_S(t) = \frac{Q_{S0}}{U_\tau}\left(1 - \tanh^2 \frac{U_{Cs} - U_x}{U_\tau}\right) \cdot U'_{Cs}(t) \qquad (21)$$

with $U_\tau << U_{ref}$ and with Q_{S0} being a constant. To distinguish accidental from systematic inaccuracies, U_x should vary slowly between 0 and U_{ref} while several periods of the oscillation are simulated. If $U_{Ci0} = 0$ then

$$U_{Ci}(t) = \int_{t_0}^{t} \frac{I_S(t)}{C_i} dt + U_{Ci0}$$

$$= \frac{Q_{S0}}{C_i}\left(\tanh \frac{U_{Cs}(t) - U_0}{U_\tau} - \tanh \frac{U_{Cs}(t_0) - U_0}{U_\tau}\right). \qquad (22)$$

A figure of merit is computed from the difference between the known analytical result and the numerically computed voltage $U_{Ci}(t)$. (The definition of a relative error would require to specify a $U_{Ci0} > 0$.) The ideal is $U_{Ci,err} = 0$:

$$U_{Ci,err} = U_{Ci,num}(t) - U_{Ci,ana}(t) \qquad (23)$$

To obtain a figure of merit for charge conservation and/or accuracy of numerical integration the numerically performed integral over the current through the capacitor C_S is compared to the known charge in C_S. In the ideal case $Q_{Cs,err} = 0$ should be found:

$$Q_{Cs,err} = \int_{t_0}^{t} I_{Cs,num}(t')dt' - C_S\left(U_{Cs}(t) - U_{Cs}(t_0)\right) \qquad (24)$$

IV. Five Demands for Analog Modeling

Analyzing a number of practical situations we would like our test functions and simulator to fulfill the following five demands: (1) A zero input signal should be really zero and not any tiny value e.g. from an exponential function. (2) We would like to have an infinite number of continuous derivatives all over the abscissa, (3) have control over the first and/or second derivative in specific time points and (4) be able to request solutions exactly in user specified time points (5) We would like to know the waveform's integral from analytical calculus.

The tangent step was proposed by Vogelsong [3]. More detailed investigations about analog waveforms suitable for behavioral modeling will be published elsewhere [4].

V. Summary

A number of problems coming along with numerical modeling and simulation of arbitrary analog signals were pointed out. On the basis of these difficulties circuit setups are proposed that allow to test a simulator's capability in handling such situations.

References

[1] M. Schubert, "Mixed Analog-Digital Signal Modeling Using Event-Driven VHDL", X Brazilian Symp. on Integrated Circuit Design - SBCCI'97, Porto Alegre, Brazil, Aug. 25-27, 1997.

[2] William C. McCalla, Fundamentals of Computer-Aided Circuit Simulation, Kluwer Academic Publishers, 1997.

[3] R. S. Vogelsong, "Tradeoffs in Analog Behav. Model Develop.: Managing Accuracy and Efficiency", BMAS, Washington, 1997.

[4] B. Gonzáles, M. Schubert, "Analog Waveform Behavioral Modeling", submitted to SBCCI'98.