Verification of Analog and Mixed-Signal Circuits using Hybrid System Techniques

Abstract

In this paper we demonstrate a potential extension of formal verification methodology in order to deal with analog and mixed-signal circuits. In particular, we focus on verifying time-domain properties. The time-dependent behavior of an analog circuit can be described by a system of differential algebraic equations. To analyze such circuits under all possible input signals and all values of parameters, we extend the reachability analysis techniques developed in the context of hybrid (discrete-continuous) control systems [4]. Moreover, these techniques can be readily applied to the verification of mixed-signal circuits since hybrid systems can naturally be used as a mathematical model for such circuits.

1 Introduction

Formal verification has become part of the development cycle of digital circuits. Its advantage relative to more traditional methods of simulation/testing lies in its exhaustiveness: It can guarantee that a system behaves correctly in the presenece of *all* its external inputs, whose number can be infinite or too large to be covered by individual simulations. Of course, this advantage does not come at no cost and verification algorithms are more complex and costly than simple simulation. An extension of verification methodology in order to deal with analog and mixed-signal circuits is far from being straightforward due to the following reason. The mathematical model for digital circuits is that of a discrete event dynamical systems (automata, transition systems) where the inputs are sequence of binary inputs, and the behavior of the circuit induced by these inputs are binary sequences corresponding to paths in the transition graph. Hence digital verification can be realized using graph search algorithms. In contrast, the mathematical model of an analog circuit is that of a continuous dynamical system defined typically by differential algebraic equations where inputs are real-valued signals defined over the real time axis, and the behaviors they induce are trajectories in the continuous state space of the system. A typical verification task is to prove that the circuit behaves correctly for all possible input signals and that none of them drives the system into a bad state, for example a state where one of the components reaches saturation. When restricting to a finite number of input signals, numerical simulation can be used to verify the behavior of the circuit. However, simulation is suitable only for previously sized circuits while many tasks in circuit design require the ability to determine the influences of element parameters on circuit performances. To this end, symbolic analysis methods (which involves computing a symbolic expression that describes the behavior or characteristic of a circuit) have been developed and successfully applied to linear or linearized circuits [9]. Extensions of symbolic methods to nonlinear circuits are mainly based on simplification and reduction to linear or weakly nonlinear systems, which is often limited by accuracy trade-offs (see e.g. [17, 16]).

In this paper we address the problem of verification of analog and mixed-signal circuits. In particular, we focus on verifying *time-domain properties* of circuits with parameter tolerances or under-specified design parameters. The time-dependent behavior of an analog circuit can be described by a system of differential algebraic equations. To analyze an analog circuit under all possible input signals and values of parameters, we extend the reachability analysis techniques for ordinary differential equations, which have been developed in the context of hybrid (discrete-continuous) control systems and implemented in the verification tool **d/dt** [4]. Moreover, these techniques can be readily applied to the verification of mixed-signal circuits since hybrid systems can indeed be used as a mathematical model for such circuits.

There have been several previous works on formal verification of analog circuits (see for example [10, 12, 14]). The work closest to this paper is [12], in which an analog system is approximated by a discrete system for which classical model-checking algorithms can be applied. The discrete system is obtained by partitioning the state space into boxes (each of which corresponds to a state of the discrete model). The transition relation is then determined by reachability relation between the boxes which is approximated by simulating trajectories from some test points in each box.

The rest of the paper is organized as follows. In Section 2 we present our approach to the verification of nonlinear analog circuits. The approach is illustrated by means of a low pass filter circuit. In Section 3 we show the application of the approach to mixed-signal circuits by means of a Δ-Σ modulator circuit.

2 Verification of nonlinear analog circuits

2.1 Approach

Mathematically, the behavior of a nonlinear analog circuit can be described by a set of differential algebraic equations (DAE):

$$F(x(t), \dot{x}(t), u(t), p) = 0 \qquad (1)$$

where $x : \mathbb{R}^+ \to \mathbb{R}^n$ denotes the state variables (internal voltages, currents, and outputs), $p \in P \subset \mathbb{R}^k$ is the parameter vector, and $u : \mathbb{R}^+ \to U$ is the input signal. We assume a set $\mathcal{U}$ of admissible input signals consisting of piecewise continuous functions taking values in a bounded and convex set $U \subseteq \mathbb{R}^l$. In this model the input is uncertain, which allows to model external disturbance and noise. A parameter can be a resistor value, a transistor saturation current, etc. The equations (1) result from applying Kirchhoff's laws to the whole circuit and the characteristics equations to the basic elements. Such circuit equations can be automatically generated, and among existing techniques to do so the Modified Nodal Analysis (MNA) [8] is the most frequently used.

To verify time-domain properties of the circuit, such as the transient behavior, one needs to characterize the set of solutions of (1) under all possible inputs $u(t)$ and all parameter values p. For safety properties (stating that the circuit never reaches a bad state), it suffices to compute the set of states reachable by all trajectories of the system. Let us formally define the reachable set notion.

We denote by $\gamma(t, x_0, u(\cdot), p)$ the solution of (1) with the initial condition $x(0) = x_0$ under the input signal $u(\cdot) \in \mathcal{U}$ and a parameter $p \in P$. Given a set of initial conditions X_0 and $T > 0$, the reachable set from Z_0 during the time interval $[0, T]$ is defined as:

$$R(T, X_0) = \{\gamma(t, x_0, u(\cdot), p) \quad | \quad t \in [0, T] \ \wedge \ x_0 \in X_0 \ \wedge u(\cdot) \in \mathcal{U} \ \wedge \ p \in P\}.$$

Note that, unlike in simulation, reachability computations can handle uncertainty in initial conditions. Reachability techniques for ordinary differential equations (ODE) have been developed in the context of hybrid systems verification and control (see for example [1] and references therein). However, an extension of these techniques to DAEs is not easy since DAEs differ from ODEs both in theoretical and numerical properties, which is classified by the *index* concept (for an introduction see [7]). The coupled system of ODEs and nonlinear equations as in (2) and (3) means that the solution has to lie on a manifold. The differential index of (1) is the minimal number of differentiations required to solve for the derivatives $\dot{y}$. The DAEs originating from practical electronic circuits are usually of index 1 or 2, and it is known that the problem of numerically solving the DAEs with index 2 or higher is ill-posed.

In this work, we assume that the circuit equations are DAEs of index 1. In particular, we shall study the equivalent semi-explicit form of (1):

$$\begin{aligned} \dot{x}(t) &= f(x(t), y(t), u(t), p) \qquad (2) \\ 0 &= g(x(t), y(t), u(t), p) \qquad (3) \end{aligned}$$

Note that the implicit DAE system (1) can be trivially transformed into the above form as follows: $\dot{x}(t) = z(t)$ and $0 = F(x(t), z(t), u(t), p)$.

If the Jacobian $g_y(x, y) = \partial g / \partial y$ is invertible in a neighborhood of the solution, then by differentiating the algebraic equation we obtain

$$\dot{y} = -g_y^{-1} g_x f \qquad (4)$$

In this case, the DAE system is of index 1. In a simpler case, where $\partial F / \partial \dot{x}$ in (1) is regular and the algebraic equation (3) disappears, (1) is simply an ODE system (i.e. of index 0).

In the following we focus on the problem of computing reachable sets of DAEs with index 1. A trivial way to do so is to transform it into an ODE composed of (2) and (4) using the above-described differentiation and then apply the existing techniques for ODEs. However, the drawback of this approach is that the solution may drift away from the algebraic constraint. To remedy this, we can retain all the original equations and their derivatives and interprete the DAE as an ODE on the manifold defined by (3). Geometric integration using projection is a standard approach for numerical simulation of ODEs on manifolds [11]. In the sequel we show how to combine this idea with reachability methods for ODEs to compute the reachable set of ODEs on manifolds.

We consider an ODE on a manifold[1]:

$$\begin{aligned} \dot{z}(t) &= h(z(t)) \qquad (5) \\ 0 &= g(z(t)) \qquad (6) \end{aligned}$$

To compute the reachable set, as mentioned earlier, we employ the projection idea to avoid the drift off phenomenon, and the differential part is handled by a reachability method for ODEs. Let $\mathcal{M}$ be the manifold defined by (6). We denote by Φ the reachability operator that takes as input a set Z and a time step

[1] For simplicity of presentation, u and p are not included, but the analysis can be generalized to systems with inputs and parameters.

$r > 0$ and returns the reachable set from Z by the ODE during the interval $[0, r]$. The idea is to compute the reachable set of the ODE (5) for each time step and project it on the manifold $\mathcal{M}$, as shown in Algorithm 1 where Z_0 is the set of initial conditions and $\Pi_{\mathcal{M}}$ is the operator of projecting a set on $\mathcal{M}$. For the computation of Φ, we use the method implemented in the tool **d/dt** [4, 3]. Basically, this method approximates the reachable set of an ODE on a step-by-step basis using polyhedra. For an affine system, given a convex polyhedron R, the set R' of states reachable from R at time r can be computed as the convex hull of the points reachable at time r from the vertices of R. Then, the set of states reachable during the whole time interval $[0, r]$ is approximated by the convex hull $conv(R \cup R')$ which is enlarged by an appropriate amount to ensure conservative approximation. If the ODE is nonlinear, it is first approximated by a piecewise affine system (using interpolation over a simplicial partition of the state space) and then the method for affine ODEs can be applied. This method was also extended to hybrid systems, i.e. systems with different continuous mode whose dynamics are described by ODEs and switching between modes is described by discrete transitions (see [2] for a detailed description of the model).

Algorithm 1 Computation of $R(T, Z_0)$

$R_0 = Z_0$
for $k = 0, 2, \ldots, \lfloor T/r \rfloor$ **do**
  $\hat{R}_{k+1} = \Phi(r, R_k)$
  $R_{k+1} = \Pi_{\mathcal{M}}(\hat{R}_{k+1})$
end for

We proceed to discuss the computation of the projection operator $\Pi_{\mathcal{M}}$. Since the sets $\hat{R}_k$ in Algorithm 1 are represented by convex polyhedra, $\Pi_{\mathcal{M}}$ can be computed by solving a constrained optimization problem. More concretely, let $V(\hat{R}_k) = \{v^1, \ldots, v^m\}$ denote the set of vertices of $\hat{R}_k$. For each vertex v^i in $V(\hat{R}_k)$, we define: $v^i_{\mathcal{M}} = \arg\min_z |z - v^i|$ subject to g(z)=0, where $|\cdot|$ is the Euclidian norm. Then, $\Pi_{\mathcal{M}}(\hat{R}_k) = conv\{v^1_{\mathcal{M}}, \ldots, v^m_{\mathcal{M}}\}$. We notice that if (6) is linear, $\Pi_{\mathcal{M}}$ can be easily computed using linear algebra. It is clear that $\Pi_{\mathcal{M}}(\hat{R}_k)$ does not always lie entirely on $\mathcal{M}$ but its distance to $\mathcal{M}$ can be made as small as desired. Indeed, we can prove that, as in the case of numerical simulation, the projection does not change the convergence order of the computation method for Φ, which is quadratic. The algorithm is illustrated by Figure 1.

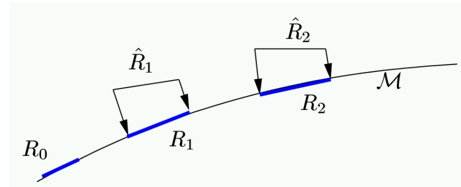

Figure 1: Combining projection and reachability computations for ODEs

We now briefly discuss some computational issues. Reachability computation, while allowing to analyze all possible behaviors of a system, is much more expensive than simulation, and complexity reduction methods are necessary for scalability purposes. Nonlinearities in analog circuits are often restricted to the device models. As an example, in Chua's circuit, to observe oscillations, it is not necessary to use an exact *arctan* shape for the voltage-current characteristics of the nonlinear resistor and it can be approximated by a piecewise linear function. The approximation of a complex function by a piecewise simpler one, which we call 'hybridization', can be used in this context as a means of complexity reduction. On the other hand, discontinuities are present in many circuit element characteristics, but they are often 'smoothened' to avoid numerical instability, and hence the use of piecewice continuous functions sometimes allows better modeling. We shall use this hybridization idea in the example that follows.

2.2 Example: A Biquad lowpass filter

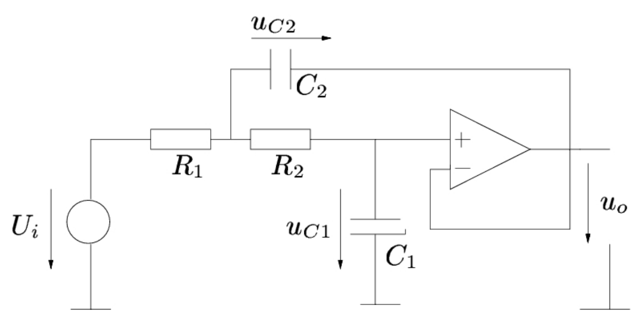

Figure 2: Lowpass filter

We now illustrate the approach with a second order biquad low pass filter circuit, shown in Figure 2. This example is taken from [12]. The circuit equations are as follows:

$$\dot{u}_{C1} = \frac{u_{C2} + u_o - u_{C1}}{C_1 R_2}$$

$$\dot{u}_{C2} = \frac{U_i - u_{C2} - u_o}{C_2 R_1} - \frac{u_{C2} + u_o - u_{C1}}{C_2 R_2} \quad (7)$$

$$u_o - V_{max} \tanh\left(\frac{(u_{C2} - u_o) V_e}{V_{max}}\right) + U_{om} = 0 \quad (8)$$

$$U_{om} = \mathcal{V}(i_0) \quad (9)$$

$$i_o = -C_2\, \dot{u}_{C2}$$

The algebraic constraints (8-9) come from the characteristics of the operational amplifier where u_o is the output voltage and U_{om} corresponds to the output voltage decrease caused by the output current i_o. In this circuit, U_i (input voltage), V_e, V_{max} (maximal source voltage) are parameters. Denoting $x = (u_{C1}, u_{C2})$ and $y = u_o$, the circuit equations can be put in the semi-explicit form (2-4). Assuming that the Jacobian $g_y(x, y)$ has bounded inverse in a neighborhood of the solution (which can indeed be verified for a concrete circuit), by differentiating (8) the DAE can then be transformed into an ODE on a manifold as in (5-6) with state variables $z = (u_{C1}, u_{C2}, u_o)$.

As mentioned earlier, to reduce the complexity of reachability computation, we shall use piecewise linear approximation. First, the nonlinear characteristics $U_{om} = \mathcal{V}(i_o)$ which is

$$\begin{aligned}\mathcal{V}(i_o) = K_1 i_o + 0.5\sqrt{K_1 i_o^2 - 2K_2 i_o I_s + K_1 I_s^2 + K_2} \\ -0.5\sqrt{K_1 i_o^2 + 2K_2 i_o I_s + K_1 I_s^2 + K_2}.\end{aligned}$$

can be approximated by a piecewise affine function of the form:

$$\mathcal{V}(i_o) = \begin{cases} K_1 i_o + K_3 & \text{if } i_o \leq I_s \\ 0 & \text{if } -I_s < i_o < I_s \\ K_1 i_o - K_3 & \text{if } i_o \geq I_s \end{cases}$$

Therefore, the original system is approximated by a hybrid system with 3 continuous modes. The conditions for staying with a mode and for switching between modes are determined by the value interval of i_o. For example in order to stay with the continuous mode corresponding to (10) the state variables (u_{C1}, u_{C2}, u_o) should satisfy: $i_o = -C_2\,\dot{u}_{C2} \leq I_s$, which together with (7) gives $-\dfrac{U_i - u_{C2} - u_o}{R_1} - \dfrac{u_{C2} + u_o - u_{C1}}{R_2} \leq I_s$. Note that the hyperbolic tangent function in (8) is retained. The property to verify is the absence of overshoots. For the highly damped case (where $C_1 = 0.5e{-}8$ and $C_2 = 2e{-}8$), Figure 3 shows the projection of the reachable set on u_{c1} and u_{C2}. The initial set is defined by a box: $u_{C1} \in [-0.3, 0.3]$, $u_{C2} \in [-0.3, 0.3]$ and $u_o \in [-0.2, 0.2]$. From the figure, one can see that u_{C1} indeed remains in the range $[-2, 2]$.

3 Application to mixed-signal circuits

In this section we illustrate the application of hybrid systems to the modeling and analysis of mixed signal circuit by mean of an example of a Δ-Σ modulator.

Δ-Σ modulation is a family of analog to digital converters (ADC) which gained a particular interest during the last two decades thanks to their high performance. However, stability analysis of such circuits remains a challenging problem.

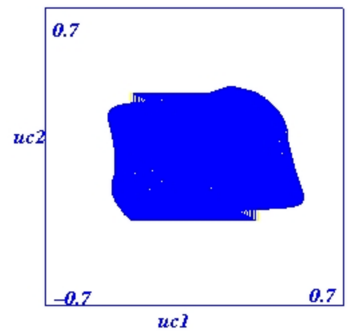

Figure 3: Reachable set projected on variables u_{C1} and u_{C2}

3.1 Δ-Σ modulation: principle

Basically, a Δ-Σ modulator processes an analog input through four steps [5]: (1) *dithering* in order to be sure that the signal bandwith lies within a given range $[-f_b, f_b]$; (2) *oversampling* or sampling at a frequency greater than the Nyquist rate $2 \times f_b$; (3) *noise shaping* so that the quantization error is 'pushed' toward high frequencies outside the bandwidth of frequencies of interest; (4) *quantization* typically on a few bits. In the following examples, quantization is done on one bit. We use an input-output plot of a simple model, shown

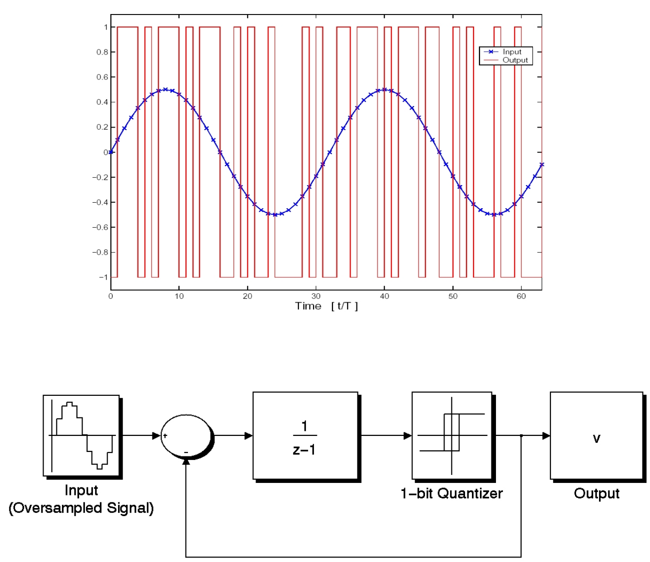

Figure 4: First order Δ-Σ modulator and example of inpout-output plot

in Figure 4 to explain intuitively how Δ-Σ modulation works. When the input sinusoid is positive, less than 1, the output takes the $+1$ value and the quantization error, which is the difference between the input and the output of the quantizer, is fed back with negative gain and 'accumulated' in the integrator $\frac{1}{z-1}$. This way, when the accumulated error reaches a certain threshold, it forces the modulator to output the value -1 for a certain time, thus reducing the mean of the quantization error. This model is called a first order Δ-Σ modulator since it uses a first order filter to process noise.

3.2 Hybrid modeling and analysis

We now study the model of a third-order Δ-Σ modulator (see Figure 5). This model is provided by the MATLAB Delta-Sigma toolbox [15].

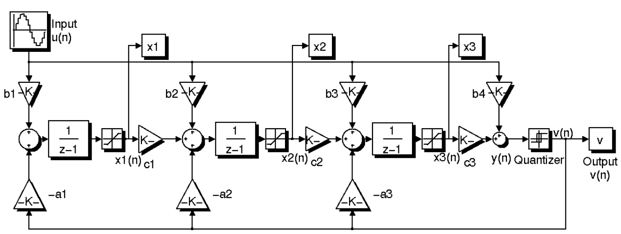

Figure 5: Model of a third-order modulator. Saturation blocks model saturation of the integrators

Higher order Δ-Σ modulators achieve better performance but induce stability issues [13]. A Δ-Σ modulator is said to be stable if under a bounded input, the states of its integrators remain bounded. This property is of great importance since integrators are naturally saturated blocks and saturation can deteriorate the circuit performance. Stability analysis for such systems is difficult due to the presence of two sources of nonlinearities: saturation and quantization. We show how these phenomena can be naturally modeled using hybrid automata.

We first notice that the modulator model under study works in discretized time (its hybrid nature lies in the fact that the input and the states of the integrators are in a continuous domain), and we shall therefore derive a discrete-time hybrid model. The continuous-time reachability techniques described in previous sections can be straightforwardly adapted to such hybrid models.

When the quantizer output remains constant and none of the integrators saturates, an affine state-space form of the system can be obtained:

$$\begin{aligned} x(n+1) &= Ax(n) + bu(n) - a \\ y(n) &= c_3 x_3(n) + b_4 \end{aligned}$$

Where $x(n) \in \mathbb{R}^3$ represents the integrator states, $u(n) \in \mathbb{R}$ is the input, $y(n) \in \mathbb{R}$ is the output. The matrix A, the vectors a and b are constant depending on the various gains of the model, and $u(n) \in \mathbb{R}$ is the input. Then saturation in the i^{th} integrator is modeled as:

$$sat(x_i(n)) = \begin{cases} x_i^{sat} & \text{if } x_i^n > x_i^{sat} \\ x_i(n) & \text{if } |x_i(n)| \leq x_i^{sat} \\ -x_i^{sat} & \text{if } x^i(n) < -x_i^{sat} \end{cases}$$

where $x_i^{sat} > 0$ is the saturation value. The output $v(n)$ is $v(n) = sign(y(n))$.

Thus, the dynamics of the third-order modulator of Figure 5 can be written as

$$\begin{aligned} x(n+1) &= sat(\tilde{x}(n+1)) \\ \tilde{x}(n+1) &= Ax(n) + bu(n) - sign(y(n))a \\ y(n) &= c_3 x_3(n) + b_4 \end{aligned}$$

This is a synthetic way to represent the dynamics of a hybrid system which is indeed piecewise affine, and in which each mode corresponds to a specific configuration of saturation of the integrators and a output value of the quantizer (for a modulator with 3 integrators and one bit, the number of modes is 54).

Figure 6 gives a graph representation of the hybrid automaton model of a modulator without saturation. This reset functions associated with transitions between modes in hybrid automata also allows to model resetting of the integrator states. Indeed, when an integrator quits its saturation zone, it can be advantageous to reset its state to 0 (see [6]).

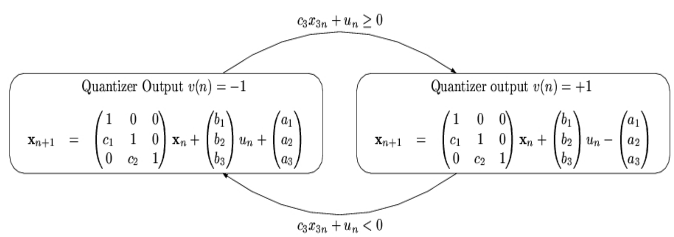

Figure 6: Hybrid automaton of a modulator without saturation

Using the resulting hybrid automaton model, we can study very interesting properties of the modulator. Saturation of the integrators (also refered to as 'clipping', see again [6]) can result in a non-desirable limit cycle behaviour. It is therefore important to know whether saturation can occur, which can indeed be checked by reachability analysis. One can also determine the set of initial states from which under a fixed bound on the input signals the system is guaranteed not to enter a saturation zone. However, it happens that the system can recover to normal operation from a saturation state. Hence, an interesting problem is to show that the system does not oscillate between saturation thresholds.

4 Conclusion

We have presented in this paper a framework for modeling and verification of analog and mixed-signal circuits using hybrid system techniques. This work opens a variety of future research directions. One challenging problem is the complexity of real-life circuits. The use of our verification method should be accompanied with various abstraction techniques exploiting the circuit structure. On the other hand, an automatic translation from a circuit description (such as in VHDL-AMS) to hybrid automata and a formal specification formalism for analog properties are necessary.

Acknowledgement. We would like to thank Lars Hedrich (IMS, University of Hannover), Bruce Krogh, Rob Rutenbar, Smriti Gupta (CMU) for providing the examples and fruitful exchanges.

References

[1] In M. Greenstreet and C. Tomlin, editors, *Hybrid Systems: Computation and Control*, LNCS 2289. Springer-Verlag, 2002.

[2] R. Alur, C. Courcoubetis, N. Halbwachs, T.A. Henzinger, P.-H. Ho, X. Nicollin, A. Olivero, J. Sifakis, and S. Yovine. The algorithmic analysis of hybrid systems. *Theoretical Computer Science*, 138:3–34, 1995.

[3] E. Asarin, T. Dang, and A. Girard. Reachability analysis of nonlinear systems using conservative approximation. In Oded Maler and Amir Pnueli, editors, *Hybrid Systems: Computation and Control*, LNCS 2623, pages 20–35. Springer-Verlag, 2003.

[4] E. Asarin, T. Dang, and O. Maler. The d/dt tool for verification of hybrid systems. In *Computer Aided Verification*, LNCS 2404, pages 365–370. Springer-Verlag, 2002.

[5] Pervez M. Aziz, Henrik V. Sorensen, and Jan Van Der Spiegel. An overview of sigma-delta converters. *IEEE Signal Processing Magazine*, pages 61–84, January 1996.

[6] Rex T. Baird and Terry S. Fiez. Stability analysis of high-order delta-sigma modulation for adc's. *IEEE Transactions on Circuits and Systems*, 41(1), January 1994.

[7] K. E. Brenan, S. L. Campell, and L. R. Petzold. *Numerical Solution of Initial Value Problems in Ordinary Differential-Algebraic Equations.* North Holland Publishing Co., 1989.

[8] U. Feldmann and M. Günther. The DAE-index in electric circuit simulation. In I. Troch and F. Breitenecker, editors, *Proc. IMACS, Symposium on Mathematical Modelling*, 4, pages 695–702, 1994.

[9] Henrik Floberg. *Symbolic Analysis in Analog Integrated Circuit Design.* Kluwer Academic Publishers, Boston, 1997.

[10] A. Ghosh and R. Verumi. Formal verification of synthesized analog circuits. In *Int. Conf on Computer Design*, 4, pages 40–45, 1999.

[11] Ernst Hairer, Christian Lubich, and Gerhard Wanner. *Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Differential Equations.*, volume 31 of *Series in Comput. Mathematics.* Springer, 2003.

[12] W. Hartong, L. Hedrich, and E. Barke. On discrete modelling and model checking for nonlinear analog systems. In *Computer Aided Verification.* Springer, 2002.

[13] Fernando Medeiro, Belen Pérez-Verdú, and Angel Rodríguez-Vázquez. *Top-Down Design of High-Performance Sigma-Delta Modulators*, chapter 2. Kluwer Academic Publishers, 2001.

[14] A. Salem. Semi-formal verification of vhdl-ams descriptors. *IEEE Int Symposium on Circuits and Systems*, 5:V–333–V–336, 2002.

[15] Richard Schreier. The delta-sigma toolbox version 6.0, January 2003.

[16] T. Wichmann. Computer aided generation of approximate DAE systems for symbolic analog circuit design. In *Proc. Annual Meeting GAMM*, 2000.

[17] T. Wichmann, R. Popp, W. Hartong, and L. Hedrich. On the simplification of nonlinear DAE systems in analog circuit design. In *Computer Algebra in Scientific Computing.* Springer, 1999.