Symbolic Analysis of Nonlinear Analog Circuits*

Alicia Manthe[†], Zhao Li[†], C.-J. Richard Shi[†] and Kartikeya Mayaram[‡]

[†]Department of Electrical Engineering, University of Washington, Seattle, WA 98195

[‡]Department of Electrical and Computer Engineering, Oregon State University, Corvallis, OR 97331

ABSTRACT

A new method is presented to model symbolically strongly nonlinear circuits, characterized by Piece-Wise Linear (PWL) functions. The method follows the idea of Bokhoven and Leenaerts, and formulates the problem as a linear complementarity problem (LCP). A new graph representation, called complementarity decision diagram, is introduced to represent the explicit LCP solutions. Orders of magnitude improvements in computational efficiency and memory usage have been obtained.

1. INTRODUCTION

Analysis of the effect of device nonlinearity on the system performance is critical to high-performance analog/RF systemson-chip design [2]. While a class of nonlinear circuits, known as *weakly nonlinear*, can be analyzed via linearized techniques such as small-signal analysis or techniques based on linearized analysis such as harmonic balance or Volterra series [2], many circuits ranging from switches, mixers, saturation-limited amplifiers to switched-capacitor filters and switching power converters, exhibit strong nonlinearities. Circuits exhibiting strong nonlinearities refer to sudden changes of device behavior, for example, switching of operating regions, sudden changes of device physics, and piecewise I-V characteristics.

Strong nonlinearities also arise in the following two scenarios. First, there is increasing interest in using digital logic signals to control the operations of analog/RF front-ends. As a consequence, more "novel" analog signal processing circuits may change their behaviors abruptly. Second, with the analog hardware description languages such as VHDL-AMS and Verilog-AMS gaining more momentum, behavioral models are being developed for systems-on-chip simulation and architecture evaluation. Many behavioral models are characterized as piecewise linear models consisting of sudden behavior changes.

Our work is inspired by the recent effort of Bokhoven and Leenaerts [1], which demonstrates that explicit formulae can be derived for a class of PWL circuits that can be formulated as socalled P-class linear complementarity problem (LCP). We propose a novel graph representation of this explicit LCP solving process. The proposed graph exploits the computational sharing inherently in the algorithm of Bokhoven and Leenaerts, and uses less than 1 percent of computations for relatively large circuits, in comparison to the straightforward implementation of the algorithm of Bokhoven and Leenaerts.

2. PRELIMINARY

Piece-Wise Linear (PWL) functions are used to model devices that exhibit strong nonlinearities.



Figure 1. An orthoator and its I-V curve.

To be able to represent each piece of the PWL function in a behavioral model van Bokhoven and Leenaerts in [1] makes use of an ideal diode. To be amenable to Modified Nodal Analysis (MNA), we will call this "new" basic two-terminal circuit element an *orthoator*, as illustrated in Figure 1. An orthoator describes the behavior of a circuit with "extremely hard" nonlinearities, and it is defined in terms of the current j through the orthoator and the voltage u across the orthoator as

$$u \ge 0, \ j \ge 0, \ u^T j = 0.$$
 (1)

The relationship between u and j is defined as the linear complementarity problem (LCP) [1]. The standard LCP resulting from circuit formulation can be re-written as follows:

$$\mathbf{u} = \mathbf{D}\mathbf{j} + \mathbf{q}$$
(2)
$$\mathbf{j}^{\mathsf{T}}\mathbf{u} = 0, \, \mathbf{j}, \, \mathbf{u} \ge 0$$

where **u** (voltage across *orthoator*), **j** (current through *orthoator*) and **q** (input sources) are column vectors of size $m \ge 1$ and **D** (linear components of the circuit) is a $m \ge m$ square matrix.

It has been shown that there exists a unique solution to (2) if and only if **D** is of class P. Then explicit solutions of **j** and **u** can be obtained explicitly using an operator called the modulus transform, which is stated here, as [1]:

$$y = \lfloor x \rfloor \rightarrow \begin{cases} y = x, & x \ge 0 \\ y = 0, & x < 0 \end{cases}$$
(3)

Consider the 1-dimensional (1-D) case (m = 1). The solution is $u = \lfloor q \rfloor$, j = 0 or $j = \lfloor -q/D \rfloor$, u = 0. This result is clearly seen by plugging in a zero for u to find j and vice versa.

The solution to the case m = 2 can be broken down to solve the problem of m = 1. Given the 2-D LCP:

$$\begin{bmatrix} u_{1} \\ u_{2} \end{bmatrix} = \begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix} \begin{bmatrix} j_{1} \\ j_{2} \end{bmatrix} + \begin{bmatrix} q_{1} \\ q_{2} \end{bmatrix}$$
(4)

Assume $j_1 = 0$, then the following is true $u_1 = \lfloor D_{12} * \hat{j}_2 + q_1 \rfloor$ and $\hat{u}_2 = D_{22} * \hat{j}_2 + q_2$. The formulation of \hat{u}_2 is equivalent to solving a 1-D case. Assume $\hat{u}_2 = 0$, then $\hat{j}_2 = \lfloor -q_2/D_{22} \rfloor$. Substitute \hat{j}_2 into u_1 yields the u_1 expression found in (5). To find j_1 then u_1 must be zero leading to: $0 = D_{11} * j_1 + D_{12} * j_2 + q_1$. Evaluating the function

^{*} This research was supported in part by U.S. Defense Advanced Research Project Agencies (DARPA) under Grant N66001-01-1-8919, and in part by the Semiconductor Research Corporation (SRC) under Contract 2000-NJ-827.

in terms of j_1 leads to the equation found in (5). The solutions for u_2 and j_2 are found the same way and are shown below.

(5)

$$u_{1} = \left[D_{12} * \left[\frac{-q_{2}}{D_{22}} \right] + q_{1} \right] \qquad j_{1} = \left[\frac{-D_{12}}{D_{11}} * \left[\frac{-q_{2}}{D_{22}} \right] - \frac{q_{1}}{D_{11}} \right] \\
u_{2} = \left[D_{21} * \left[\frac{-q_{1}}{D_{11}} \right] + q_{2} \right] \qquad j_{2} = \left[\frac{-D_{21}}{D_{22}} * \left[\frac{-q_{1}}{D_{11}} \right] - \frac{q_{2}}{D_{22}} \right]$$

In general, an *n*-dimensional (n-D) case can be found in the same way by breaking the problem down into smaller matrices. The *n*-D case leads to *n* levels of the modulus transform. Clearly this procedure takes an exponential amount of computation and space.

3. COMPLEMENTARITY DECISION DIAGRAM FOR EXPLICIT LCP SOLVING

As stated in Section 2, the LCP problem itself can be explicitly solved by using the methodology presented in [1]. However, the computation of these explicit functions can take an exponential amount of time. A new method is necessary to symbolically represent these expressions in a compact and efficient manner to be commendable to computer simulation.

We can determine any j_k , where $k = 1 \dots n$, by using the technique described in Section 2. Ideally we need to find a way to represent these exponentially increasing expressions. We introduce *complementarity decision diagrams* (CDDs) to embody these explicit LCP solutions symbolically. These CDDs are capable of exploiting the sharing of computation during LCP solving process.

The explicit equations for u and j have a distinct pattern, which allows the formula to be expressed by a CDD. A CDD is a signed, rooted, direct acyclic graph, similar in form to a determinant decision diagram (DDD), originally introduced in [3]. As illustrated in Figure 2, a CDD *vertex* V is associated with a subexpression (V.subexpr), a positive or negative *sign* (V.sign), and at most two edges. The edges can be one of the possible three; *1dot_edge* (solid line with solid dot), *1-edge* (solid line) and *0-edge* (dotted line). The 1-dot_edge and 1-edge point to a 1-child (V.1child) and 0-edge always points to a 0-child (V.0child). A CDD has one type of terminal and that is called *q-terminal*. A qterminal can be any one of the following {*q1, q2, ..., qm*}. Then each vertex V represents a *symbolic, explicit LCP expression* (V.expr) defined as follows:

V.expr = V.sign * V.subexpr * (V.child1).expr + (V.child0).expr. (9)

The *I-dot_edge* implies the modulus transform operator is executed, [*V.expr*].

In Figure 2a, the CDD of a 1-D case is shown inside the solid circle. To the right is the CDD of the 2-D case. Notice the 2-D consists of 1-D CDDs, shown inside dashed circles. For the 2-D case, sharing amongst the q's is exploited. For a 3-D CDD sharing is seen amongst the 1-D subgraph and q's, for a 4-D CDD sharing is in use amongst the 2-D and lower subgraphs, and for the n-D CDD sharing of subgraphs is utilized amongst the (n-2)-D and lower subgraphs.

A key rule we used in constructing the CDD is to enforce the order of expansion of the matrix rows and columns. For example, consider the two vertices inside the solid circle (D_{12} and D_{13}) in Figure 2b. The order of this chain is {2, 3}, where the order is

determined by the representation of j in D_{ij} . Since these two vertices in the chain are connected through the 0-edge representing addition, then due to the communitive property order $\{2, 3\}$ is equivalent to $\{3, 2\}$. Therefore, when any such chain is constructed, we enforce the same vertex order (from the smallest to the largest row/column index). This allows the maximal possible sharing of sub-graphs (sub-expressions) in the CDD.



Figure 2. (a) Instances of CDDs: 1-D case (left), 2-D case (right). (b) 3-D subgraph of a 5-D CDD.

Shown on the right is a comparison of the size of CDDs created dense LCP from matrices versus the number of terms representative in an explicit LCP expression. Clearly the CDD sizes increase much less dramatically than the size of the expression. This is due to the massive sharing of sub expressions in the CDD.



4. CONCLUSIONS

A graph-based method is presented for symbolic analysis of strongly nonlinear analog circuits. The method is based on the explicit procedure due originally to Bokhoven and Leenaerts. We proposed a compact graph representation, called the Complementarity Decision Diagram (CDD), and showed that the CDDs can be orders of magnitude more efficient than the original method for relatively large circuits.

5. **References**

- D. M. W. Leenaerts and W. M.G. van Bokhoven, *Piecewise Linear Modeling and Analysis*, Kluwer Academic Publishers, 1998.
- [2] K. Mayaram, "Distortion" in *The Electrical Engineering Handbook*, Editor, R. C. Dorf, Second Edition, CRC Press, pp. 147-157, 1997.
- [3] C.-J. Shi and X.-D. Tan, "Canonical symbolic analysis of large analog circuits with determinant decision diagrams", *IEEE Trans. Computer-Aided Design*, vol. 19, pp. 1-18, Jan. 2000.