Automatic Abstraction of Transistor Level Circuits to Hybrid Automata

***Abstract*—Formal verification of analog transistor level circuits is still an open problem. In this presentation we want to present an methodology which automatically abstracts a very accurate sampled analog circuit block using a reachability tool with SPICE-accuracy. The resulting hybrid automata (HA) models the – in most analog circuits desired – linear behavior, but also the technology dependent nonlinear and limiting behavior. We evaluate the accuracy of the model using some sample simulations of the SPICE-netlist and the HA. Finally, we show that it can be used for formal verification on high level with reachability tools.**

I. Introduction

The demand for formal verification has increased rapidly with the development of safety-critical systems, such as autonomous driving, medical surgeries and in human-robot environments. In the analog circuit domain formal verification is a problem tackled with different methods on different abstraction levels. For high level continuous systems methods modeling the analog circuit or its environment as a hybrid system is widely used [1], [2], [3], [4]. The methods are able to handle up to 20 state variables , if the underlying locations use linear ordinary differential equations (ODEs). Mostly they use reachability analysis to prove safety properties. However, monitor based approaches can be used to prove more LTL or CTL like properties [5].

If we are interested to close the chain of proof to the transistor level, we cannot continue with the hybrid automata (HA) as the ODEs become nonlinear differential algebraic equations (DAEs). In [6] a method for modeling the underlying DAE-system of electrical networks using piecewise linear regions for each nonlinear element is presented. It suffers from using an abstract transistor-model and the limited number of transistors to be verified. An advantage is the on-the-fly generation of locations of the HA automata preventing a state explosion problem at initialization and during evaluation of a given input stimulus.

In order to have results with technology accuracy e.g. an BSIM3 or Hicum accurate verification we could use sample based formal verification methods [7] loosing the range based proof. However, these methods are able to handle much larger circuits up to 80 transistors.

In this presentation we want to describe a method which combines the latter approaches automatically building a very abstract hybrid automata from full accurate sampled data of the state space of the underlying circuit. We believe in reaching these abstract model due to the fact, that many analog circuits have a large desired linear operating range. The resulting HA should consist of some linear states and some more adjacent states modeling the non-linearities in the limiting regions.

Fig. 1 illustrates the concept of our methodology. We start with an analog transistor-level netlist, use Vera [7] to discretize the reachable state space, cluster states in few large nearly linear regions and construct out of these regions the desired HA. Finally we are able to use the HA in the formal reachability tool Cora [4].

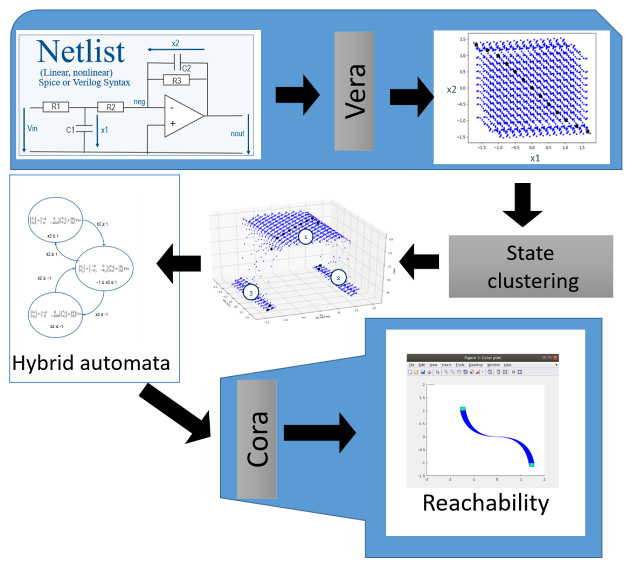


Fig. 1. Overview of the presented methodology

The total method is fully automatic and will result in a large speed up in formal circuit verification due to accurate proved abstract hybrid automata.

II. SPICE-accurate Sampling of reachable Region

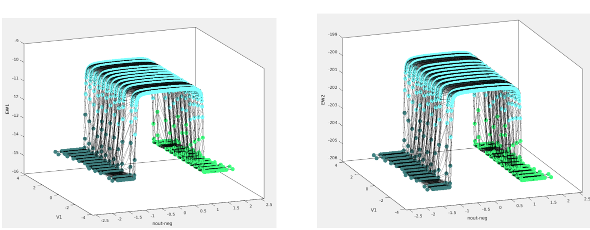
The starting point for the modeling process is a SPICE-netlist on transistor level or behavioral level (Verilog-A) or a mixed netlist. We are able to read netlists for Gnucap [8] or Titan, an in-house simulator of a big microelectronic company. These simulators solve the underlying circuit with full accurate. They are coupled with the equivalence checking tool Vera [9] using special communication and controlling modules enabling the directed solving of the nonlinear DAE system on arbitrary points in the state space. We use here only the reachability and sampling algorithms from Vera to construct a sampling of the reachable state space with the following properties:

- nonlinear order reduction of the states up to a given frequency,
- graph connection structure of the states consisting of predecessors, successors, timing information on edges between states and slew rate limited input connections,
- information of the nonlinear and locally linearised system at each sample point:
  - SPICE-accurate large signal circuit solution $\vec{X}_{state}$,
  - reduced Kronecker canonical linearized system $\dot{\vec{z}} = \underline{A} \cdot \vec{z} + \vec{b} \cdot u$ with system matrix $\underline{A}$, input vector b and transformed variable vector $\vec{z}$,
  - transformation matrices to Kronecker form $\underline{E}, \underline{F}$.

III. Eigenvalue based State Clustering

The file produced by Vera contains for every point a set of information including the eigenvalues, the operating point in the X-space and the numeric values for the transformation matrices $\underline{E}$ and $\underline{F}$. The objective now is to identify regions in the linear transformed z state space based on eigenvalues in order to find the states of a hybrid automata. To do so, first the eigenvalues are clustered by the k-means method. We extend the basic algorithm by an silhouette coefficient to identify an appropriate k – the number of clusters.

In addition to the eigenvalues we have to consider the spatial distribution in the state space. We use a breadth-first search on the graph Vera created resulting in connected regions with similar eigenvalues. Finally for each cluster, the information from the surrounding points are unified and the mean of the eigenvalues and right eigenvectors $\underline{\text{F}}$ and left eigenvectors $\underline{\text{E}}$ are generated. Fig. 2 illustrates the proposed clustering method for second order active low-pass filter with limitation at $\pm 1V$. Where EW, $V1$, $nout$ and neg stand for eigenvalue, input voltage, output voltage and the voltage at the negative terminal of the OP respectively. The ca. 4000 sampled states are clustered in 3 nearly linear regions.



(a) first eigenvalue (b) second eigenvalue

Fig. 2. State clustering into 3 regions illustrated for both eigenvalues in state space. Points belonging to a region have a unique color (dark green, light green and turquois).

IV. Generation of Hybrid Automata

From the obtained average values of the previous part, a hybrid automata in the z-space is created. The eigenvalue clustering resulted in N distinct locations which are the states of the HA. At each state, the behavior of the system is modeled using the eigenvalues and the transformation of the input vector to the z-space. The guards and invariants for each location are also obtained from the clustering process using halfplanes approximating the borders of the regions.

The hybrid automata that is created can be passed to Cora for reachability analysis. After the analysis is finished, the obtained values (zonotopes) are transformed back into the X-space using the transformation $\vec{x} = \underline{F}\vec{z} + x_{ap}$. An additional term must be considered in the locations of the hybrid automaton that do not contain the origin. This term corrects $\vec{z}$, and compensates for the vectorial shifting from the origin of the z-space.

V. Results and Conclusion

We applied the method to an low-pass filter on circuit level with an operational amplifier resulting in a limiting behavior. The results shown in Fig. 2 give a clear indication of the large linear region and the two limiting regions. The comparison of a simulation of the abstracted HA with the original netlist results in low errors. The limiting behavior have been accurately captured. In the presentation we will show results of some more analog circuits.

The proposed methodology builds an abstract hybrid automaton based on sample data of the state space. This method is fully automated and will increase scalebility of formal verification. The technique presented is very promising as there are different levels where significant fine tuning can occur, due to the insight of the system behavior, e.g. using linear parametric equations to describe a sequence of states.

References

[1] T. Dang, A. Donzé, and O. Maler, "Verification of analog and mixed-signal circuits using hybrid system techniques," in *Formal Methods in Computer-Aided Design*, pp. 21–36, Springer, 2004.

[2] G. Frehse, C. Le Guernic, A. Donzé, S. Cotton, R. Ray, O. Lebeltel, R. Ripado, A. Girard, T. Dang, and O. Maler, "SpaceEx: Scalable verification of hybrid systems," in *Computer Aided Verification*, pp. 379–395, Springer, 2011.

[3] M. Fränzle, H. Hungar, C. Schmitt, and B. Wirtz, "HLang: Compositional Representation of Hybrid Systems via Predicates," Reports of SFB/TR 14 AVACS 20, SFB/TR 14 AVACS, July 2007. ISSN: 1860-9821, http://www.avacs.org.

[4] M. Althoff, A. Rajhans, B. H. Krogh, S. Yaldiz, X. Li, and L. Pileggi, "Formal verification of phase-locked loops using reachability analysis and continuization," *Communications of the ACM*, vol. 56, no. 10, pp. 97–104, 2013.

[5] O. Maler and D. Nikovic, "Monitoring properties of analog and mixed-signal circuits," *International Journal on Software Tools for Technology Transfer*, vol. 15, no. 3, pp. 247–268, 2013.

[6] H. S. L. Lee, M. Althoff, S. Hoelldampf, M. Olbrich, and E. Barke, "Automated generation of hybrid system models for reachability analysis of nonlinear analog circuits," in *Design Automation Conference (ASP-DAC), 2015 20th Asia and South Pacific*, pp. 725–730, Jan 2015.

[7] S. Steinhorst and L. Hedrich, "Trajectory-directed discrete state space modeling for formal verification of nonlinear analog circuits," in *Proceedings of the International Conference on Computer-Aided Design*, pp. 202–209, ACM, 2012.

[8] A. Davis, "An overview of algorithms in Gnucap," in *University/Government/Industry Microelectronics Symp.*, pp. 360–361, 2003.

[9] S. Steinhorst and L. Hedrich, "Equivalence checking of nonlinear analog circuits for hierarchical AMS System Verification," in *VLSI and System-on-Chip (VLSI-SoC), 2012 IEEE/IFIP 20th International Conference on*, pp. 135–140, IEEE, 2012.