Establishing Reachset Conformance for the Formal Analysis of Analog Circuits

Niklas Kochdumper*, Ahmad Tarraf**, Malgorzata Rechmal***, Markus Olbrich***, Lars Hedrich**, Matthias Althoff*
*Technical University of Munich, **Goethe University Frankfurt, ***Leibniz University Hannover
*{niklas.kochdumper@, althoff@tum.de, **{tarraf@, hedrich@em.cs.uni-frankfurt.de, ***{malgorzata.rechmal@, markus.olbrich@ims.uni-hannover.de

Abstract— We present the first work on the automated generation of reachset conformant models for analog circuits. Our approach applies reachset conformant synthesis to add non-determinism to piecewise-linear circuit models so that they then enclose all possible behaviors of the real system. Since piecewise-linear models are hybrid, we introduce the first reachset conformant synthesis algorithm for hybrid system models. Furthermore we present a novel technique to compute the required non-determinism. The effectiveness of our approach is demonstrated on a real analog circuit. Since the resulting models enclose all measurements, they can be used for formal verification.

I. INTRODUCTION

Since many safety-critical systems like autonomous vehicles, robots collaborating with humans, and automated medical systems, are controlled by circuits, there is an increasing demand for their formal verification. In general, formal analysis tools for dynamical systems (e.g., CORA [2], Flow* [5], HyLAA [4], and SpaceEx [7]) require simple, yet conformant models of the real system. Approximate circuit models are often not conformant to the real system [19].

We present reachset conformant synthesis to add non-determinism to an approximate circuit model so that the adapted model contains the set of all possible system behaviors. Since the required non-determinism is automatically computed from measurements of the real circuit, the resulting reachset conformant model does not only enclose all differing system behaviors due to approximation errors in the model, but also due to disturbances, sensor noise, and inaccuracies in manufacturing. In this work, we present the first approach for the automated generation of reachset conformant models for analog circuits.

State of the Art Reachset conformance testing is a recently developed approach based on reachability analysis that can be applied to check if a system model is conformant with measurements of the real system [3, 12]. As visualized in Fig. 1, the behavior of the real system (red line) is enclosed by the reachable set of the model (gray area). It is shown in [12] that reachset conformance is sufficient for the formal verification of safety properties. Other works successfully applied reachset conformance testing for robot manipulators [9], human arms [14], and pedestrians [10]. An extension to reachset conformance testing is reachset conformant synthesis as introduced in [9], where the required non-determinism is determined automatically. The authors of [9] presented a reachset conformant synthesis algorithm for linear continuous system models. Since [9] is the only work on reachset conformant synthesis so far, there does not yet exists a reachset conformant synthesis algorithm for hybrid systems.

One method related to reachset conformance is equivalence checking, which checks if two analog circuits exhibit the same behavior under the same input excitation [15]. Equivalence checking is often applied to verify the equivalence between a behavioral and an accurate circuit model [13,15].

One major advantage of the approach presented in this work is that it can be applied to arbitrary dynamic piecewise linear (PWL) approximative circuit models. PWL models partition the state space into regions, where the system behavior for each region is described by a linear ordinary differential equation, see e.g., [6, 11]. More recent approaches try to use these models for formal verification. [20] directly generates these models but concentrates on oscillator circuits with no inputs. The approach in [18] does not exhibit a PWL model but reformulates the verification problem as a satisfiability problem.

The nominal models used in this work are generated from [16] based on eigenvalue clustering and the local
linearization-based approach from [8].

**Notation** Sets are denoted by calligraphic letters, matrices by uppercase letters, vectors by lowercase letters, and lists by bold uppercase letters. Given two matrices $C$ and $D$, $[C \mid D]$ denotes the concatenation of the matrices. Given a list $L$, the operations $\text{remove}(L, l)$ and $\text{add}(L, l)$ remove and add the element $l$, respectively. The left multiplication of a matrix $M \in \mathbb{R}^{m \times n}$ with a set $S \subset \mathbb{R}^n$ is defined as $MS = \{Ms \mid s \in S\}$, and the Minkowski addition of two sets $S_1 \subset \mathbb{R}^n$ and $S_2 \subset \mathbb{R}^n$ is defined as $S_1 \oplus S_2 = \{s_1 + s_2 \mid s_1 \in S_1, s_2 \in S_2\}$.

II. Reachset Conformance

The goal of reachset conformant synthesis is to add non-determinism to a nominal system model such that the resulting model includes all possible behaviors of the real system. The nominal system model is given as

$$\dot{x}(t) = f(x(t), u(t)), \quad x(t) \in \mathbb{R}^n, \quad u(t) \in \mathbb{R}^m,$$

where $x$ is the state vector and $u$ is the input vector. To preserve time information during reachability analysis, we introduce the extended state vector $z(t) = [x(t) \, u(t)]^T$, so that

$$\dot{z}(t) = \begin{bmatrix} f(x(t), u(t)) \\ 1 \end{bmatrix} = \hat{f}(z(t), u(t)). \tag{1}$$

We add non-determinism through uncertain additive inputs $V \subset \mathbb{R}^n$ and uncertain measurement errors $W \subset \mathbb{R}^n$; we use the shorthand $U = \langle V, W \rangle_U$ for a concise notation. Adding $V$ to (1) results in the differential inclusion

$$\dot{z}(t) \in \left\{ \hat{f}(z(t), u(t)) + \begin{bmatrix} v \\ 0 \end{bmatrix} \mid v \in V \right\}. \tag{2}$$

We model PWL models by hybrid automata:

**Definition 1** (Hybrid Automaton) A hybrid automaton $H$ with $p$ discrete modes consists of (a) a list $F = (f_1(\cdot), \ldots, f_p(\cdot))$ storing the differential equations $\dot{z}(t) = f_i(\cdot)$ describing the dynamic in each mode $i = 1, \ldots, p$; (b) a list $S = (S_1, \ldots, S_p)$ storing the invariant set $S_i \subset \mathbb{R}^{n_i}$ for each mode $i = 1, \ldots, p$; and (c) a list $T = (T_1, \ldots, T_q)$ storing the transitions $T_i = \langle G_i, r_i(\cdot), s_i, g_i \rangle_T$, $i = 1 \ldots q$ between discrete modes, where $G_i \subset \mathbb{R}^{n_i+1}$ is the guard set, $r_i : \mathbb{R}^{n_i+1} \rightarrow \mathbb{R}^{n_i+1}$ is the reset function, and $s_i, g_i \in \{1, \ldots, p\}$ are the indices of the source and target modes, respectively.

For a concise notation, we use the shorthand $H = (F, S, T)_{HA}$ for a hybrid automaton. The state of a hybrid automaton is defined as $\sigma(t) = (z(t), m(t))$, where $z(t) \in \mathbb{R}^{n+1}$ is the continuous state, and $m(t) \in \{1, \ldots, p\}$ is the discrete state. The evolution of a hybrid automaton is described informally as follows: Given an initial state $\sigma_0 = \sigma(0) = \langle z_0, m_0 \rangle_S$ with $z_0 \in S_{m_0}$, the continuous state $z(t)$ evolves according to the flow function $\hat{f}_{m_0}(\cdot)$ of the mode $m_0$. If $z(t)$ is within the guard set $G_i$ of a transition $T_i = \langle G_i, r_i(\cdot), s_i, g_i \rangle_T \in T$ with $s_i = m_0$, the transition to the mode $g_i$ is taken and the continuous state $z(t)$ is updated according to the reset function $r_i(\cdot)$. Afterward, the evolution of the continuous state continues according to the flow function $\hat{f}_{g_i}(\cdot)$ of mode $g_i$ until the next transition is taken. We denote the trajectory of the continuous state for the evolution of the hybrid automaton described above by $\xi(t, u(t), \sigma_0, \nu)$, where $\nu$ is the model uncertainty.

**Definition 2** (Reachable Set) The reachable set at time $t$ for a hybrid automaton $H$, a nominal system input $u_n(\cdot)$, a set of initial continuous states $Z_0 \subset \mathbb{R}^{n+1}$, and the initial mode $m_0$ is

$$R_H(t, u_n(\cdot), Z_0, m_0, U) \equiv \{\xi(t, u_n(\cdot), (m_0, z_0)_S, \nu) \mid z_0 \in Z_0, \nu \in U\},$$

and the bloated reachable set is

$$B_H(t, u_n(\cdot), Z_0, m_0, U) \equiv \bigcup_{i=1}^{p} R_{H,i}(t, u_n(\cdot), \Sigma_0, U) \oplus W_i,$$

where $R_{H,i}(t, u_n(\cdot), Z_0, m_0, U)$ is the part of the reachable set belonging to the $i$-th mode. The list $U = (U_1, \ldots, U_p)$ stores the model uncertainty $U_i = \langle V_i, W_i \rangle_U$ for each mode $i = 1, \ldots, p$.

The reachable set for a differential inclusion as defined by (2) is denoted by $R_f(t, u_n(\cdot), Z_0, U)$.

We denote by $\mu(t, \sigma(0), u_n(\cdot))$ the measured trajectory of the system state $\sigma(t)$ for the system input $u_n(\cdot)$. For conformance checking, a test suite is generated by measuring $h$ trajectories $\mu(t, \sigma_0, u_{n,i}(\cdot))$ for different input signals $u_{n,i}(\cdot)$, $i = 1, \ldots, h$ at sampled times $t_0 = 0, \ldots, t_h = t_e$.

$$Y_i = (\mu(t_1, \sigma_0, u_{n,i}(\cdot)), \ldots, \mu(t_h, \sigma_0, u_{n,i}(\cdot))),$$

where $t_e$ is the time horizon. For a concise notation we use the shorthand $M_i = \langle Y_i, u_{n,i}(\cdot) \rangle_M$ for a measurement $M_i$ consisting of the measured trajectory $Y_i$ and the corresponding input signal $u_{n,i}(\cdot)$.

The goal of reachset conformant synthesis is to choose the model uncertainty $U$ such that the volume of the final bloated reachable set is minimized while all measurements are enclosed by the bloated reachable set:

$$\min_U \sum_{i=1}^{h} \text{volume}(B_H(t, u_{n,i}(\cdot), z_{0,i}, m_{0,i}, U))$$

s.t. $\forall i = 1 \ldots h, \forall j = 0 \ldots k$

$$z_{j,i} \in B_H(t_{j}, u_{n,i}(\cdot), z_{0,i}, m_{0,i}, U),$$

where $(z_{j,i}, m_{j,i}) = \mu(t_j, \sigma_0, u_{n,i}(\cdot))$ is the measured state and the operation volume returns the volume of a set.
A. Reachset Conformance for Hybrid Automata

We first present Alg. 1 for reachset conformant synthesis of hybrid automata. Instead of solving (3), Alg. 1 heuristically computes a feasible and close to optimal solution in a computationally efficient way by obtaining the required model uncertainty for each discrete mode independently. The inputs to Alg. 1 are the nominal hybrid automaton $H$, the initial model uncertainties $U$ for each mode, and the measurements $M$. Alg. 1 uses a queue $M$ that stores all measurements for which conformance is not yet guaranteed. The while-loop in line 4 of Alg. 1 iterates until $M$ is empty, in which case all measurements are reachset conformant.

We initialize the starting point for the reachable set computation $\hat{z}_{0,j}$ with the continuous part of the first measured state $\sigma_0$ in line 2 of Alg. 1. In each iteration of the while-loop in line 4, the for-loop in line 5 iterates over the $p$ modes of the hybrid automaton $H$. For each mode $i$, we select from the queue $M$ all measurements $M_i$ for which the first hybrid state $\sigma_0$ belongs to mode $i$ (see line 8 of Alg. 1). These measurements are split in lines 13, 14, and 18 of Alg. 1 into one part $M_j$ with measurements belonging to mode $i$, and one remainder $\overline{M}_j$.

Using the extracted measurement parts $M_i$, the uncertainty $U_i$ for the current mode $i$ is updated in line 23 of Alg. 1 with the operation \texttt{reachConfMode}, whose implementation is described later in Sec. C. The lists $X$ and $T$ store, for each measurement in $M_i$, the corresponding initial continuous state and initial time for the reachable set computation, respectively. We perform \textit{reachset conformance synthesis} for a mode in the original rather than the extended state space.

The new starting point $\hat{z}_{0,j}$ for the reachable set computation for all remainders of the measurements $\overline{M}_j \in M$ must also be calculated. This is illustrated by the example shown in Fig. 2. First, the reachable set $P$ for the current measurement $\overline{M}_j$ is computed according to line 25 of Alg. 1 (see Fig. 2 (a)). Using the calculated reachable set $P$, we select the transition $T_i \in T$ that is taken by the trajectory of measurement $\overline{M}_j$ with the for-loop in line 26 of Alg. 1. For the example shown in Fig. 2 there exist six transitions $T_1, \ldots, T_6$ with guard sets $G_1, \ldots, G_6$ whose source is the current mode $i = 1$. From these transitions only $T_2$ satisfies $s_l = i \land g_l = m_0 \land P \cap G_l \neq \emptyset$ with $i = 1$ and $m_0 = 2$ (see Fig. 2 (b)). The new starting point for the next mode $\hat{z}_{0,j}$ is therefore computed as $\hat{z}_{0,j} = r_2(\text{center}(P \cap G_2))$ (see Fig. 2 (c)) according to line 28 of Alg. 1, where operation \texttt{center} returns the center of a set. Finally, we add the remainders of the measurements $\overline{M}_j$ to the queue $M$ in line 32 of Alg. 1.

---

**Algorithm 1 reachConfHA($H$, $U$, $M$)**

**Require:** Hybrid automaton $H = (F, S, T)_{H,A}$ with $p$ modes, where $F = (f_1(\cdot), \ldots, f_p(\cdot))$, list of initial model uncertainties $U = (U_1, \ldots, U_p)$, list of measurements $M = (M_1, \ldots, M_k)$.

**Ensure:** Reachset conformant model defined by the nominal system model $H$ and the list of model uncertainties $U = (U_1, \ldots, U_p)$.

1: for $M_j := \langle (\sigma_0, \ldots, \sigma_k), u_n(\cdot) \rangle \in M$ do
2: \hspace{1em} $\hat{z}_{0,j} \leftarrow z_0$
3: end for
4: while $M \neq \emptyset$ do
5: \hspace{1em} for $i \leftarrow 1$ to $p$ do
6: \hspace{2em} $M \leftarrow \emptyset$, $\overline{M} \leftarrow \emptyset$, $X \leftarrow \emptyset$, $T \leftarrow \emptyset$
7: \hspace{2em} for $M_j := \langle (\sigma_0, \ldots, \sigma_k), u_n(\cdot) \rangle \in M$ do
8: \hspace{3em} if $m_0 = i$ then
9: \hspace{4em} $M \leftarrow \text{remove}(M, M_j)$
10: \hspace{4em} $X \leftarrow \text{add}(X, \hat{x}_{0,j})$, $T \leftarrow \text{add}(T, \hat{t}_{0,j})$
11: \hspace{4em} if $\exists l \in \{0, \ldots, k\}$ $m_l \neq i$ then
12: \hspace{5em} $l^* \leftarrow \min_{l \in \{0, \ldots, k\}} m_l \neq i$
13: \hspace{5em} $\overline{M}_j \leftarrow \langle (\sigma_0, \ldots, \sigma_{l^*}), u_n(\cdot) \rangle$
14: \hspace{5em} $\overline{M}_j \leftarrow \langle (\sigma_{l^*+1}, \ldots, \sigma_k), u_n(\cdot) \rangle$
15: \hspace{5em} $\overline{M} \leftarrow \text{add}(\overline{M}, \overline{M}_j)$
16: \hspace{5em} $\overline{M} \leftarrow \text{add}(\overline{M}, \overline{M}_j)$
17: \hspace{4em} else
18: \hspace{5em} $\overline{M}_j \leftarrow \langle (\sigma_0, \ldots, \sigma_k), u_n(\cdot) \rangle$
19: \hspace{5em} $\overline{M} \leftarrow \text{add}(\overline{M}, \overline{M}_j)$
20: \hspace{4em} end if
21: \hspace{3em} end if
22: \hspace{2em} end for
23: \hspace{1em} $U_i \leftarrow \text{reachConfMode}(f_i(\cdot), M_i, X, T, U_i)$
24: \hspace{1em} for $\overline{M}_j := \langle (\sigma_0, \ldots, \sigma_k), u_n(\cdot) \rangle \in \overline{M}$ do
25: \hspace{2em} $P \leftarrow \mathcal{R}_{f_i}([0, t_0 - \hat{t}_{0,j}], [t_0 - \hat{t}_{0,j}], \hat{z}_{0,j}, U_i)$
26: \hspace{2em} for $T_i := (G_i, r_i(x), s_i, g_i) \in T$ do
27: \hspace{3em} if $s_i = i \land g_i = m_0 \land P \cap G_i \neq \emptyset$ then
28: \hspace{4em} $\hat{z}_{0,j} \leftarrow r_i(\text{center}(P \cap G_i))$
29: \hspace{4em} break
30: \hspace{3em} end if
31: \hspace{2em} end for
32: \hspace{2em} $M_j \leftarrow \overline{M}_j$, $\overline{M} \leftarrow \text{add}(\overline{M}, M_j)$
33: \hspace{2em} end for
34: \hspace{1em} end for
35: \hspace{1em} end while

---
B. Reachset Conformance for Analog Circuit Models

There are two main types of errors that contribute to differing behaviors between the PWL model and the real system: 1) the abstraction error made by using an approximative PWL model, and 2) real world errors resulting from disturbances, sensor noise, and inaccuracies in the manufacturing. Our uncertainty model is constructed so that the abstraction error is mainly captured by the uncertain additive inputs $\mathcal{V}$ and the real-world errors are mainly captured by measurement uncertainties $\mathcal{W}$. The overall approach for analog circuits therefore first performs reachset conformant synthesis using simulations of the accurate system model to compute feasible sets $\mathcal{V}_1, \ldots, \mathcal{V}_p$, so that the reachable set encloses the simulations (see Fig. 3 (a)). Then, reachset conformant synthesis uses measurements of the real system to compute feasible sets $\mathcal{W}_1, \ldots, \mathcal{W}_p$, so that the bloated reachable set encloses the measurements (see Fig. 3 (b)).

![Image](image_url)

**Fig. 3:** Simulation of an accurate system model (blue), simulation of the approximate system model (red), measurements of the real system (green), reachable set (gray), and bloated reachable set (orange).

C. Reachset Conformance for Linear Continuous Systems

As described in Sec. II.A, we use the original state $x(t)$ instead of the extended state $z(t)$ for reachset conformant synthesis of a single mode. For PWL models, the flow function $f(\cdot)$ is linear for each mode of the hybrid automaton:

$$\dot{x}(t) = f(x(t), u(t)) = Ax(t) + Bu(t) + c,$$

where $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, and $c \in \mathbb{R}^n$. With the linear dynamics from (4), the reachable set and the bloated reachable set as defined in Def. 2 are according to [1, Sec. 3.2] computed as

$$\mathcal{R}_f(t, u_n(\cdot), X_0, U) \equiv e^{At} X_0 + \int_0^t(\int_0^{\tau} e^{A(t-\tau)}(Bu_n(\tau) + c)d\tau) d\tau + \int_0^t e^{A(t-\tau)}d\tau \mathcal{V} \equiv \mathcal{H} \oplus EV \oplus W,$$

and

$$\mathcal{B}_f(t, u_n(\cdot), X_0, U) \equiv \mathcal{H} \oplus EV \oplus W,$$

where $X_0 \in \mathbb{R}^n$ is the initial set and $U = \langle \mathcal{V}, \mathcal{W} \rangle_U$ are the model uncertainties. To solve (3), we present the computationally efficient implementation of reachConfMode described in Alg. 2. The quality of the obtained solution is demonstrated with numerical examples in Sec. III.

Within Alg. 2, vertices returns the vertices of a set, and interval returns an enclosing interval for a given point cloud.

**Algorithm 2 reachConfMode(f(\cdot), M, X, T, U)**

**Require:** Nominal system model $f(x, u) = Ax + Bu + c$, list of measurements $M = \{M_1, \ldots, M_h\}$, list of initial sets $X = \{X_{0,1}, \ldots, X_{0,h}\}$ and list of initial times $T = \{t_{0,1}, \ldots, t_{0,h}\}$ for each measurement, initial model uncertainty $U = \langle \mathcal{V}, \mathcal{W} \rangle_U$.

**Ensure:** Reachset conformant model defined by the nominal system model $f(\cdot)$ and the updated model uncertainty $U^*$.

1: for $M_i := \langle \mathcal{Y}, u_n(\cdot) \rangle_M \in M$ do
2: for $z_j \in \mathcal{Y}$ do
3: $\mathcal{H} \oplus EV \leftarrow \mathcal{R}_f(t_j - t_{0,i}, u_n(t + t_{0,i}), X_{0,i}, U)$ using (5)
4: if type($M_i$) == "simulation" then
5: $b \leftarrow \text{diffFromSet}(\mathcal{H}, z_j)$
6: $\hat{v} \leftarrow \text{solution of } EV = b$
7: $v_1, \ldots, v_q \leftarrow \text{vertices(}\mathcal{V}\text{)}$
8: $\mathcal{V}^* \leftarrow \text{interval}(\hat{v}, v_1, \ldots, v_q)$
9: $\mathcal{V} \leftarrow \mathcal{V}^*$
10: else if type($M_i$) == "measurement" then
11: $\hat{w} \leftarrow \text{diffFromSet}(\mathcal{H} \oplus EV, z_j)$
12: $w_1, \ldots, w_q \leftarrow \text{vertices}(\mathcal{W})$
13: $\mathcal{W}^* \leftarrow \text{interval}(\hat{w}, w_1, \ldots, w_q)$
14: $\mathcal{W} \leftarrow \mathcal{W}^*$
15: end if
16: end for
17: end for
18: $U^* \leftarrow \langle \mathcal{V}, \mathcal{W} \rangle_U$

The for-loop in line 1 of Alg. 2 iterates over all measurements $M_i \in M$, and the for-loop in line 2 of Alg. 2 iterates over all points $z_j \in \mathbb{R}^{n+1}$ of measurement $M_i$. In line 3 of Alg. 2, we compute the reachable set at the time of the measurement $t_j$ for each measured point.

As described in Sec. II.B, the additive uncertainty $\mathcal{V}$ and the measurement uncertainty $\mathcal{W}$ are determined separately. In lines 4 - 15 of Alg. 2, we therefore update either $\mathcal{V}$ or $\mathcal{W}$, depending on the type of the measurement $M_i$ returned by the operation type.

To prove that Alg. 2 is correct, we have to show that the updated set of additive uncertainties $\mathcal{V}^*$ satisfies $x_j \in H \oplus EV^*$, and the updated set of measurement uncertainties $\mathcal{W}^*$ satisfies $x_j \in H \oplus EV \oplus W^*$ for all measurement points $x_j$. Since the proofs for both cases are very similar,
we only consider the updated set of additive uncertainties \( V^* \) due to space limitations. From the reachable set \( H \oplus EV^* \), only the summand \( EV^* \) can be influenced by the set of uncertainties \( V^* \). To determine a suitable set \( V^* \), we therefore first compute the difference \( b \) between the current measurement point \( x_j \) and the set \( H \) in line 5 and of Alg. 2 using

\[
diffFromSet(S, p) = p - \arg\min_{s \in S} \| p - s \|_2, \quad (7)
\]

where \( p \in \mathbb{R}^n \) is a point and \( S \subset \mathbb{R}^n \) is a set.

The feasible uncertainty \( V^* \) is obtained by the following theorem:

**Theorem 1** Given two sets \( S_1 \subset \mathbb{R}^n \) and \( S_2 \subset \mathbb{R}^n \), as well as a point \( p \in \mathbb{R}^n \), it holds that \( p \in S_1 \oplus S_2 \) if \( \diffFromSet(S_1, p) \in S_2 \).

**Proof** According to (7), it holds that

\[
a = \diffFromSet(S_1, p) = p - \arg\min_{s \in S_1} \| p - s \|_2.
\]

If \( a = \diffFromSet(S_1, p) = p - s_1 \in S_2 \) it further holds that

\[
p = p - s_1 + s_1 = p - s_1 + s_1 | s_1 \in S_1 \}
\]

which proves that \( p \in S_1 \oplus S_2 \). \( \square \)

With \( b = \diffFromSet(H, x_j) \) as computed in line 5 of Alg. 2, it holds according to Theorem 1 that \( x_j \in H \oplus EV^* \) if \( b \in EV^* \). Since \( \text{interval} \) computes an enclosing interval for a given point cloud, the updated uncertainty set \( V^* \) computed in line 8 of Alg. 2 satisfies

\[
V \subseteq V^* \quad \text{and} \quad \hat{v} \in V^*.
\]

Since the point \( \hat{v} \) computed in line 6 of Alg. 2 satisfies \( E\hat{v} = b \), it therefore holds that

\[
b = E\hat{v} \in EV^*.
\]

which proves that \( x_j \in H \oplus EV^* \) according to Theorem 1.

The implementation of the operation \( \text{reachConfMode} \) presented in this subsection concludes our \( \text{reachset con-}

**III. Numerical Example**

In order to demonstrate the effectiveness of our approach we consider the example of a second-order low-pass filter (see Fig. 4). For the accurate system model the operation amplifier is modeled in SPICE with a LMC6484 described at transistor level. The resistors R1, R2, and R3 are chosen as 4.7kΩ, 10kΩ, and 4.5kΩ, while the capacitors C1 and C2 are set to 0.1µF and 1µF, respectively. We apply two approaches for the PWL model generation of analog transistor level circuits: eigenvalue clustering and local linearizations of the non-linear circuit.

To generate the behavior model using eigenvalue clustering, the approach presented in [17] and [16] is used. The number of different modes for the hybrid automaton is identified using the group and region identification described in [16]. One major advantage of this approach is the model order reduction: given a circuit with \( q \) states, a behavior model with \( n \ll q \) states is generated. For the considered low-pass filter, eigenvalue clustering results in a hybrid automaton with \( p = 3 \) modes.

In contrast to eigenvalue clustering, local linearization [8] utilizes a white box model. Based on the original circuit topology, the behavioral model is composed of static non-linear PWL device models and linear dynamic device models. For the considered low-pass filter, local linearization results in a hybrid automaton with \( p = 6 \) modes.

Our test suite consists of simulation results from an accurate system model as well as measurement from the real circuit for the input signals \( u_{n,1}(t) = 3V, u_{n,2}(t) = 4V, u_{n,3}(t) = 3V \sin(\omega_1 t), u_{n,4}(t) = 3V \sin(\omega_2 t), u_{n,5}(t) = 4V \sin(\omega_1 t), \) and \( u_{n,6}(t) = 4V \sin(\omega_2 t) \), where \( \omega_1 = 2\pi \) and \( \omega_2 = 200\pi \). Using this test suite and the generated nominal system models, we apply \( \text{reachset con-} \)

As shown in Fig. 5, the reachable sets for both generated reachset conformant models enclose all measurements for the input signals \( u_n(t) = 4.5V \) and \( u_n(t) = 4V \sin(20\pi t) \), even though these input signals are not included in the test suite considered for \( \text{reachset con-} \).
IV. Conclusion

We introduced the first approach for the fully automated generation of reachset conformant models for analog circuits. The resulting conformant model is well suited for formal verification since it is based on a simplified PWL abstraction of the circuit dynamics. For reachset conformant synthesis, we presented the first algorithm to calculate the required model uncertainty for hybrid systems. Furthermore, we introduced a reachset conformance concept and uncertainty model that is well suited for analog circuits, and in addition proposed a novel reachset conformant synthesis algorithm for linear continuous systems. Finally, we demonstrated the effectiveness of our overall approach on a real analog circuit, where we used two recently developed algorithms for the automated generation of PWL circuit models.

Fig. 5.: Reachable sets of the reachset conformant models generated with eigenvalue clustering and local linearization for the input signals \( u_n(t) = 4.5V \) (top) and \( u_n(t) = 4V \sin(20\pi t) \) (bottom). The corresponding measurements on the real circuit are depicted in red.

REFERENCES