Intertwined Global Optimization Based Reachability Analysis

Ibtissem Seghaier(✉) and Sofiène Tahar

Department of Electrical and Computer Engineering,
Concordia University, Montréal, QC, Canada
{seghaier,tahar}@ece.concordia.ca

Abstract. This paper proposes a semi-formal reachability analysis technique based on global optimization for hybrid systems. In order to model the hybrid system dynamics with parameter and noise disturbance, a system of stochastic recurrence equations formalism is proposed. Then, a reachability analysis approach is adopted to compute the reachable sets under an interval of initial conditions and in light of system parameters variability. The novelty of our approach is in approximating the reachable bounds in an intertwined forward/backward manner. The backward corrections refine the obtained reachable bounds in the forward scheme and so reduce the high reachability over-bounding due to the wrapping effect. Finally, a Monte Carlo hypothesis testing based technique is performed on the resultant reachable bounds to uncover the hybrid system failure with regard to a certain specification. These failures are quantified in terms of parametric yield rate which reflects the sensitivity of the hybrid system to variations in its parameters. We demonstrate the effectiveness of our proposed verification methodology by applying it on a mixed analog and digital electronics building block commonly used in communications systems.

Keywords: Hybrid systems · System of stochastic recurrence equations · Intertwined forward-backward reachability analysis

1 Introduction

Continuous and discrete systems behaviors have been extensively analyzed separetly by control theory and formal verification communities, respectively. However, the verification of their composition in the same system, termed as hybrid system, has gained a lot of attention lately [1]. Indeed, hybrid systems are basic blocks in embedded control systems that involve interaction between digital systems and the physical world via analog plants (e.g., sensors and actuators) [2]. The complex infinite possible behaviors that a hybrid system exhibits rend the verification of such systems both challenging and critical, especially for safety critical applications such as avionics, automotive engine, and medical systems [3]. Verification becomes particularly challenging with hybrid models that account

K. Barkaoui et al. (Eds.): VECoS 2017, LNCS 10466, pp. 139–154, 2017.
DOI: 10.1007/978-3-319-66176-6_10

for real system imperfections such as system parameter variations due to fabrication impurities along with input fluctuations. Monte Carlo simulation is a cornerstone and perhaps the most common practice in the verification of hybrid systems [4]. However, it is not sufficient to carry out multiple simulations when the system is actually required to match its specifications for all possible initial conditions and process parameters. Instead, reachability analysis techniques which refer to computing the set of all possible system behaviors emanating from an initial reachable set are adopted to prove that they satisfy a desired specification. Current reachability analysis techniques can be broadly classified into three main categories. Namely, SMT-solving [5], theorem proving [6] and flowpipe computation-based techniques [7]. Most of these reachability analysis techniques can only handle hybrid systems with linear continuous dynamics but a few are readily scalable to systems with nonlinear dynamics. In addition, because reachalbility analysis is in general undecidable, over-approximation is required to ensure the decidability of the reachability problem. This leads to verification errors in the computed reachable set that accumulates and even blows up with the reachable set evolution over time. This problem, known as *wrapping effect*, becomes a great concern for an accurate verification of hybrid systems. Hence, an efficient verification of these systems dictates two key requirements: (1) a uniform modeling formalism that fully reflects the relations as well as the interactions of the discrete and continuous parts of the system. With the uniformity, the model should also provide accuracy by realistically replicating noise, parameters and initial conditions variation; and (2) an accurate reachability analysis scheme that can handle nonlinear continuous hybrid systems and assess the effect of parameter variations while reducing the wrapping effect.

In this paper, we present a novel methodology for modeling and verification of continuous and hybrid systems under parameter and initial conditions uncertainties using a system of stochastic recurrence equations formalism. We propose an intertwined forward/backward reachability analysis technique based on global optimization that is capable of reducing the wrapping effect. The key insights is that for the purpose of nonlinear hybrid system verification, the reachable sets are tracked precisely by a backward reachability correction approach and the system failure rate is estimated using a hypothesis testing based approach.

The rest of this paper is organized as follows: in Sect. 2, we introduce some preliminary definitions of hybrid system modeling, including Latin Hypercube sampling and hypothesis testing techniques. Section 3 then discusses our proposed methodology for hybrid systems modeling and verification. In Sect. 4, we demonstrate the effectiveness of our methodology by applying it on a common analog and mixed signal design used in communication systems. Finally, conclusions and future work are given in Sect. 5.

2 Preliminaries

In this section, we define the terminology that will be used for hybrid systems modeling. We also present some background on the Latin Hypercube Sampling technique, and statistical hypothesis testing along with their definitions.

2.1 Hybrid System Modeling: System of Stochastic Recurrence Equations

Hybrid systems contain two different types of components: those with continuous dynamics and those with discrete dynamics. Despite their heterogeneous nature, a careful time domain discretization allows a unified description of all the hybrid systems components. Due to the statistical behavior that hybrid systems exhibit in the presence of uncertainties (such as noise and parameter variability), we are interested in modeling hybrid systems as a System of Stochastic Recurrence Equations (SSRE) [8], which is a formalism that allows to capture the statistical properties of the system in a unified discrete-time description. Moreover, the temporal properties of these hybrid components and their interactions can be expressed as SSRE. In what follows, we explain the SSRE notations and detail the conversion process of system equations and properties to SSREs. A system of recurrence equations is a set of relations between consecutive elements of a sequence. The notion of recurrence equations to describe discrete systems using the normal form: *generalized If-formula* was first proposed by Al-Sammane [9]. In addition, a stochastic recurrence equation can be generated for the case of continuous systems using the discrete version of their Stochastic Differential Equation (SDEs) [10]. In the following, we briefly present the SSRE theory. An SSRE is a set of SREs with stochastic processes. Let us consider the following Itô process $\{X_t, 0 \leq t \leq T\}$ SDE [11]:

$$dX_t(\omega) = f(X_t(\omega))dt + \sigma(X_t(\omega))dW_t(\omega) \tag{1}$$

where the stochastic variable W_t is a Brownian motion [12] (see Definition 1), the initial condition ($X_{t_0} = X_0$) and the diffusion coefficient σ are deterministic variables.

Definition 1. *(Brownian Motion) A scalar standard Brownian process, or standard Winer process over [0,T] is a random variable W_t that depends continuously on $t \in [0,T]$ and satisfies the following conditions:*

Condition 1. *$W(0) = 0$ with probability 1.*

Condition 2. *For $0 \leq s < t \leq T$ the random variable given by the increment $W_t - W_s$ is normally distributed with mean zero and variance $(t-s)$ ($W_t - W_s \sim \sqrt{t-s}\mathcal{N}(0,1)$).*

Condition 3. *For $0 \leq s < t < u < v \leq T$ the increments $W_t - W_s$ and $W_v - W_u$ are independent.*

By integrating Eq. (1) between s and $s + \Delta s$, we will have:

$$dX_{s+\Delta s}(\omega) = X_s(\omega) + \int_s^{s+\Delta s} f(X_{s+\Delta s}(\omega))dt + \int_s^{s+\Delta s} f\sigma(X_{s+\Delta s}(\omega))dW_{s+\Delta s}(\omega) \tag{2}$$

The Euler scheme [13] consists in approximating the integral Eq. (2) by the following iterative scheme:

$$\bar{X}_{s+\Delta s}(\omega) = \bar{X}_s(\omega) + f(\bar{X}_s(\omega))\Delta s + \sigma(W_{s+\Delta s}(\omega) - W_s(\omega)) \tag{3}$$

Definition 2. *(Generalized If-formula) The generalized* ***If-formula*** *is a class of symbolic expressions that extend recurrence equations to describe discrete systems. Let i and n be natural numbers. Let* $\mathbb{K}$ *be a numerical domain in (*$\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}$ *or* $\mathbb{B}$*), an* ***If-formula*** *is one of the following:*

- *A variable* $X_i(n)$ *or a constant C that takes values in* $\mathbb{K}$
- *Any arithmetic operation* $\diamond \in \{+, -, \times, \div\}$ *between variables* $X_i(n)$ *that take values in* $\mathbb{K}$
- *A logical formula: any expression constructed using a set of variables* $X_i(n) \in \mathbb{K}$ *and logical operators:* $not, and, or, xor, nor, \ldots,$ *etc.*
- *A comparison formula: any expression constructed using a set of variables* $X_i(n) \in \mathbb{K}$ *and comparison operators* $\triangle \in \{\neq, =, <, \leq, >, \geq\}$
- *An expression* $If(X, Y, Z)$*, where X is a logical formula or a comparison formula and* Y, Z *are any generalized* ***If-formula****.*

Here, $If(X, Y, Z) : \mathbb{B} \times K \times K \longrightarrow \mathbb{K}$ *satisfies the axioms:*

1. *If(true, X ,Y) = X*
2. *If(false, X ,Y) = Y*

Definition 3. *(SSRE) Consider a set of variables* $X_i(n) \in \mathbb{K}$*,* $i \in V = 1, \ldots, k, \omega \in, \mathbb{R}$*, an SSRE is a system of the form:*

$$X_i(\omega) = f_i(X_j(\omega)\gamma)), (j, \gamma) \in \varepsilon_i, \forall \omega \in \mathbb{R} \tag{4}$$

where $fi(X_j(\omega)\gamma))$ is a generalized ***If-formula*** of the recurrence stochastic differential equation given in Eq. (3). The set ε_i is a finite non empty subset of $1, \ldots, k \times \mathbb{N}$. The integer γ denotes the delay.

2.2 Latin Hypercube Sampling

To study parameter variation effects on the behavior of hybrid systems, an optimal exploration of the variation domain of the parameter values is very important in order to achieve a good accuracy and avoid non-informative verification runs. Traditional sampling techniques (e.g., Pseudo Random Sampling (PRS), Fractional Factorial, Central Composite, etc.) only arrange parameter values at some specific corners in the parameter space and can not handle multivariate stochastic parameters especially in terms of correlation. Consequently, when performing verification, it cannot mimic the system behavior in a global system parameter space. We first look at PRS as applied in the estimation of system failure in order to justify the use of Latin Hypercube Sampling (LHS). It has been demonstrated that the LHS technique gives samples that could reflect the integral distribution more effectively with a reduced samples variance [14]. Figure 1 illustrates the differences while using Monte Carlo PRS and Gaussian Monte Carlo LHS of a random normal parameter of transistor width for 1000 trials.

In the sequel, we explain the Latin Hypercube Sampling (LHS) main steps to generate a sample size N from n hybrid system parameter variables $\xi = [\xi_1, \xi_2, \ldots, \xi_n]$ with the probability distribution function $f_\xi(.)$.

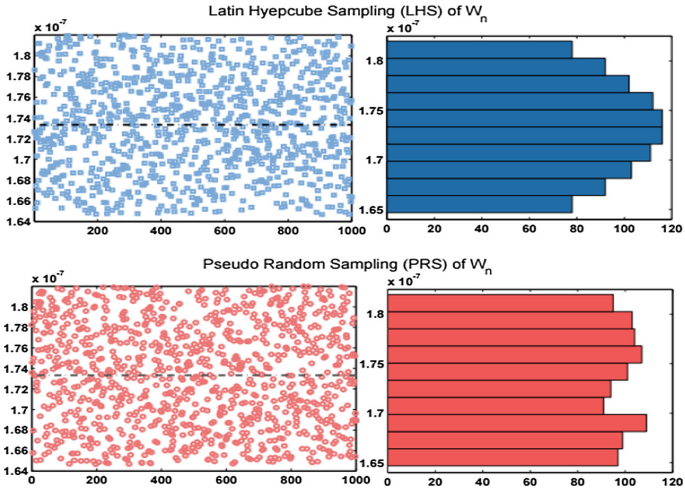

Fig. 1. Sampling differences between Monte Carlo PRS and LHS

First, the approach involves the partitioning of the range of each system parameter variable into N non overlapping intervals on the basis of equally probability size $\frac{1}{N}$. One value from each interval is randomly selected w.r.t. the conditional probability density in the variation interval defined by the technology library. The N values thus obtained for ξ_1, are paired in a random manner with the N values of ξ_2. These N pairs are combined in a random manner with the N values of ξ_3 to form N triplets, and so on, until a set of $N \times n$-tuples is formed. The choice of this sampling technique can be justified by its variance sampling reduction, which results in a better sampling coverage and consequently a better verification coverage [15].

2.3 Hypothesis Testing

Hypothesis testing [16] uses statistics to make decisions about the acceptance or the rejection of some statements based on the data from random samples. In this technique, the property of interest is formulated as a null hypothesis (H_0) which is tested against an alternative hypothesis (H_1). If we reject H_0, then the decision to accept H_1 is made.

Definition 4. *Given the property $\mathcal{P}$ within the ambit of a null hypothesis H_0, a significance level α, and a test statistic T, hypothesis testing is the process of verifying whether a system $\mathcal{S}$ satisfies H_0 with a probability greater than or equal to α (i.e., $\mathcal{S} \models Pr(T) \geq \alpha$).*

As depicted in Fig. 2, Hypothesis testing can be a one side test (upper test or lower tes) or two sided. In the case of a two sided test for example, we can verify if a variable X is within a bounded region $[x_1, x_2]$ as follows:

$$H_0 : P(x_1 < X < x_2) = P(X < x_2) - P(X < x_1) = 1 - \alpha \tag{5}$$

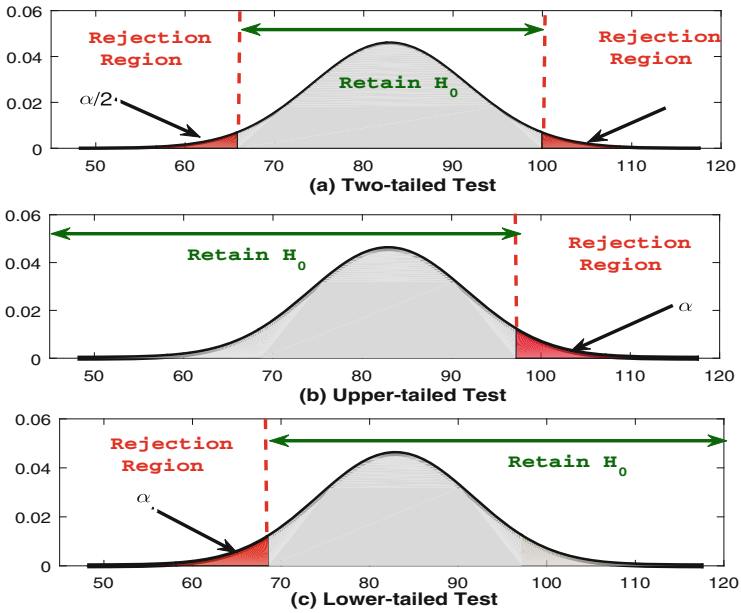

Fig. 2. Hypothesis testing concept

Following are the central steps to carry out hypothesis testing:

1. Elucidate the property to be verified and formulate it as H_0 and H_1.
2. Specify the appropriate level of significance α and determine the type of the test, namely, upper test, lower test or two sided test.
3. Select the appropriate test statistic.
4. Compute the critical region or p-value of the test statistic.
5. Compute the test statistic of the observed value for the original data.
6. Make the decision of accepting or rejecting the null hypothesis H_0. If the computed test statistic falls in the critical region, then the null hypothesis is rejected, otherwise H_0 is accepted.

The performance criteria of this approach is related to two types of errors as shown in Table 1:

Table 1. System verification classification

	Passed	Failed
Good System	✓	Type I error
Failed System	Type II error	✓

Type I error (α) or false positive, the null hypothesis H_0 is true but the decision based on the testing process erroneously rejected it. In other words, it represents the probability of accepting H_0 when H_1 holds.

Type II error (β) or false negative, the null hypothesis H_0 is false but the testing process concludes that it should be accepted. In other words, it corresponds to the probability of accepting H_1 when H_0 holds.

3 Proposed Methodology

An overview of the proposed methodology for intertwined forward/backward reachability analysis is shown in Fig. 3. Given a nonlinear hybrid system description, SSREs that express its stochastic behavior under noise perturbation are generated. The proposed SSRE formalism features a sound treatment of noise. It actually allows a consistent consideration of the noise effect to which the system is incurred during the reachability analysis process. More details about the system uncertainties modeling can be found in [17]. Then, parameter values from a certain distribution of the parameter space are derived using the efficient LHS technique. Next, reachability bounds of the hybrid system for a continuous set of initial conditions, and under the derived system parameters are generated using a novel intertwined forward/backward reachability analysis technique. The reachability computed using SSRE system model with parameters selected by the LHS procedure and for initial conditions that are defined within intervals (n-cubes) is based on the global optimization theory. The SSRE is not solved for every initial condition value but it employs the reachability analysis algorithm to optimize the search for the global extremum.

The output of this step is a refined reachability set generated from the backward reachability correction that includes all possible actual behaviors (trajectories) of the system. The main advantage of the proposed verification scheme is its generality and scalability. In fact, it does not make any assumption about the nature of the hybrid system dynamics: it works for any hybrid system with linear and nonlinear behavior. Next, appropriate null and alternative hypotheses are formulated from a certain SSRE specification of the hybrid system under verification. For each selected system parameters in the reachability iteration, Hypothesis Testing based Monte Carlo (MC) technique is conducted to estimate the system parametric failure which refers to failures caused by the deviation between manufactured system parameter values and intended parameter values. Each time the null hypothesis H_0, which represents the desired system property, is rejected, we draw a conclusion that the system fails to comply with its property and so we increment the number of system failures $N_{failure}$. Finally, the system yield rate is computed based on the probability of failure $P_{Failure}$ as follows:

$$P_{Failure} = \frac{Nb.\ of\ Rejected\ H_0}{Total\ Nb.\ of\ MC\ Trials}$$
$$Yield = 1 - P_{Failure}$$

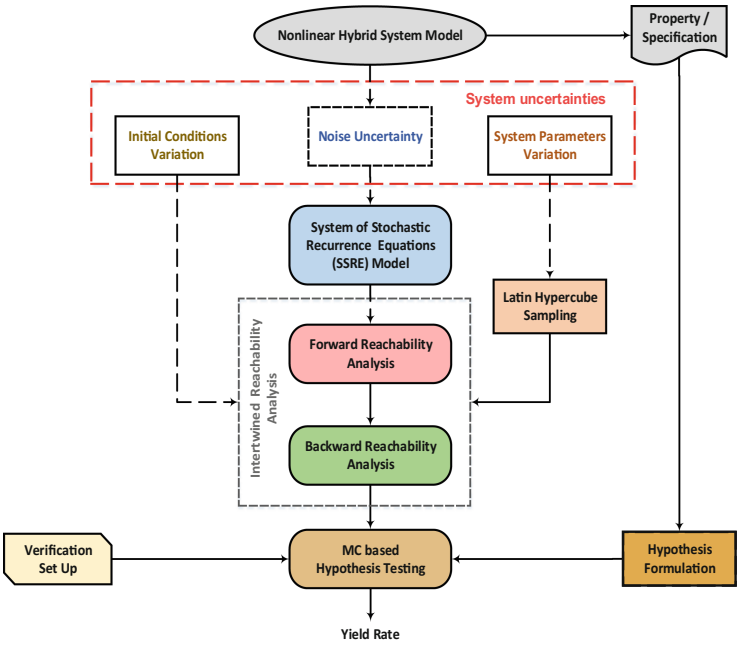

Fig. 3. Proposed verification methodology

3.1 Forward-Backward Reachability Analysis

Definition 5. *(Reachability Analysis) Reachable set (or bounds) is the collection of all possible trajectories or states of the hybrid system dynamical behavior originated from an interval of initial conditions. Mathematically, this can be defined as follows:*

$$X_{Reachable\_set} = \{x \in \mathbb{R}^{N_x} \mid \underline{X_L} \leq x \leq \overline{X_U}\} \tag{6}$$

where $\underline{X_L}$ *is the lower reachable bound of the reachable set (or region) and* $\overline{X_U}$ *is the upper bound of the reachable set.*

The proposed intertwined reachability analysis approach is shown in Fig. 4. The definition of reverse time dynamics of the SSRE model allows the forward/backward reachability exchange. The detailed implementation of the intertwined reachability analysis approach is summarized in Algorithm 1. Hybrid dynamical systems: An introduction to control and verification. Given an interval system of stochastic differential equations (an SSRE whose initial conditions are intervals), the algorithm defines the region of uncertainty of the system as an hypercube (n-cube) at time t_0 (Lines 3 and 18). Hence, the reachability analysis problem at a given simulation time point t^* for each system output (or state space) is equivalent to finding the maximum and minimum bounds of the SSRE model. In the proposed algorithm, the reachability analysis problem is so cast

Algorithm 1. Intertwined Forward/Backward Reachability Analysis

Require: $SSRE$: Hybrid System Model, X_0 : Interval of Initial Conditions, P : System parameters, N_x : Number of state variables, t_0 : Initial time, t_f : Final time

1: **for** $t_1^* \leftarrow t_0$ *to* t_f **do**
2: **for** $j \leftarrow 1$ *to* N_x **do**
3: $X_{ext}(t_1^*) = Generate(X_0)$ ▷ external surface of the uncertainty region
4: $X_{max}(t_1^*, j) = -\infty$
5: $X_{min}(t_1^*, j) = \infty$
6: **for** *each state variable* $X_{ext}(j) \in X_{ext}$ **do**
7: $Const = UpdateConstar(j, SSRE, P, X_{ext})$
8: $Grad = UpdateGrad(j, t_1^*, SSRE, P, X_{ext}))$
9: $[X_{max}(t_1^*), X_{min}(t_1^*)] = Global\_Opt(SSRE, j, t_0, t_1^*, P, X_{ext}), Grad, Constr)$
10: **end for**
11: $B_{L_{Forward}}(t_1^*) \leftarrow X_{min}(t_1^*)$
12: $B_{U_{Forward}}(t_1^*) \leftarrow X_{max}(t_1^*)$
13: $update\_forward(t_1^*, \Delta_t)$
14: **end for**
15: **end for**
16: **for** $t_2^* \leftarrow t_f$ *to* t_0 **do**
17: **for** $j \leftarrow 1$ *to* N_x **do**
18: $X_{ext}(t_2^*) = Generate(B_{L_{Forward}}(t_2^*), B_{U_{Forward}}(t^*))$ ▷ external surface of the approximate reachability bounds
19: $X_{max}(t_2^*, j) = B_{U_{Forward}}(t_2^*, j)$
20: $X_{min}(t_2^*, j) = B_{L_{Forward}}(t_2^*, j)$
21: **for** *each state variable* $X_{ext}(j) \in X_{ext}$ **do**
22: $Const = UpdateConstar_B(j, SSRE, P, X_{ext})$
23: $Grad = UpdateGrad_B(j, t_2^*, SSRE, P, X_{ext}))$
24: $[X_{max}(t_2^*), X_{min}(t_2^*)] = Global\_Opt_B(SSRE, j, t_f, t_2^*, P, X_{ext}), Grad, Constr)$
25: **end for**
26: $B_{L_{corrected}}(t_2^*) \leftarrow X_{min}(t_2^*)$
27: $B_{U_{corrected}}(t_2^*) \leftarrow X_{max}(t_2^*)$
28: $update\_backward(t_2^*, \Delta_t)$
29: **end for**
30: **end for**

into a constrained multivariable nonlinear global optimisation problem. It was proven that under continuity condition, it is sufficient to compute the evolution of the external surface of the uncertainty region [18]. This means that to calculate the reachable bounds, it is sufficient to compute the trajectories emanating from the external surface of the region of the uncertainty region.

The extreme functions (Max and Min) at a specific time t^* of the system equations $SSRE(t^*, j, X_{ext}), \forall j = 1, \ldots, N_x$, which bound the system behavior, are first computed using the forward reachability analysis. We used the MATLAB Optimization solver [19] based on trust regions (Lines 1 to 15) to get these extreme functions of $SSRE(t^*, j, X_{ext}), \forall j = 1, \ldots, N_x$ by fixing the time variable to t^* and constraining the system behavior to evolve over the external uncertainty region (Line 7). The computed optimization point is then passed to the SSRE model, which uses X_{ext} as initial conditions and generates a partial derivatives (gradient) values that are used to control the stability of the reachability analysis (Line 8). The algorithm terminates if the optimisation method considers $SSRE(t^*, j, X_{ext}), \forall j = 1, \ldots, N_x$ as an extremum;

Otherwise the gradient values are used to select new points from the external uncertainty region X_{ext} and the above described steps are repeated. Athough this step guarantees the completeness of the reachability set, the upper and lower

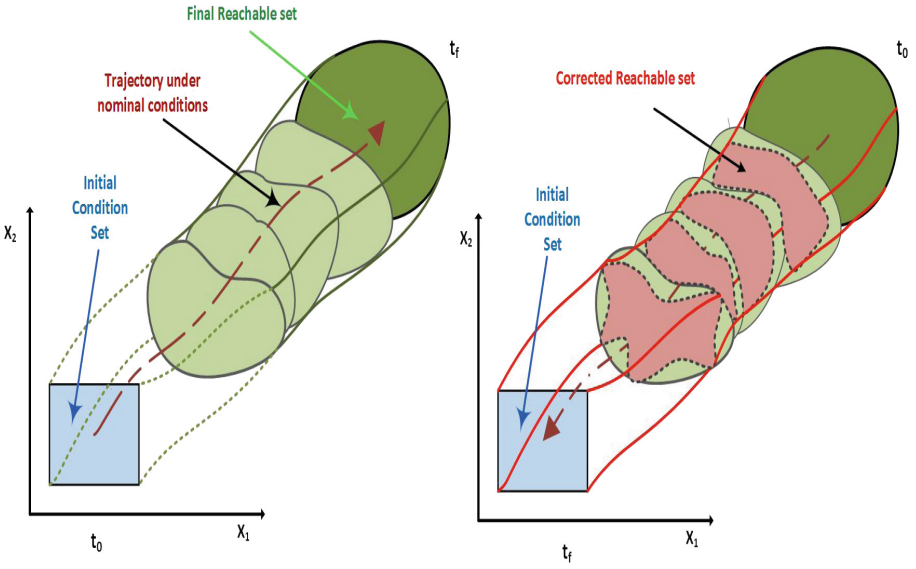

Fig. 4. Intertwined reachability analysis concept

obtained reachable sets are highly overbounded due to the *wrapping effect*. One way to tighten the reachability space is to conduct a backward reachability (Lines 16 to 30). Starting from the final computed set (Line 18), the backward optimization algorithm is now performed on the hybrid system SSRE reversed in time in order to compute backwards the reachability bounds and consequently correct the overbounded forward reachability set.

4 Application: PLL Frequency Synthesizer

In this section, we validate our proposed intertwined forward reachability analysis with backward correction methodology on a Phase Locked Loop (PLL) mixed signal design. More details about PLL case study as well as the results of another application are reported in [17]. All computation and hybrid system models were integrated in MATLAB environment and were run on a 64-bit Windows 7 machine with 2.8 GHz processor and 24 GB memory. The hypothesis testing is conducted for a level of significance $\alpha = 5\%$.

The PLL based frequency synthesizer is a basic and essential block of modern communication systems. It is basically a feedback circuit that tries to reduce the phase error between the input and the reference signals. In this case study, we consider a simple frequency synthesizer, that generates an output signal whose frequency is N times the frequency of the reference signal. We consider for this application a *Sine wave* reference signal with a frequency of ω_0, the PLL output is a *Cosine wave* signal with frequency $N \times \omega_0$.

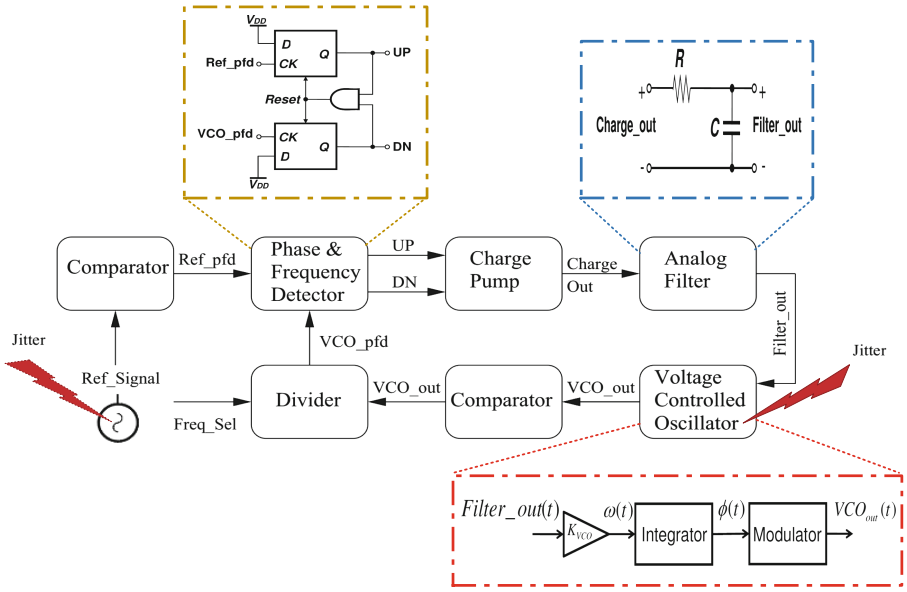

Fig. 5. PLL design block diagram

Figure 5 shows a block based description of a second order PLL based frequency synthesizer. It consists of a reference oscillator, a Charge Pump (CP), a Low Pass Filter (LPF), and a Voltage Controlled Oscillator (VCO). In order to model this PLL using SSREs notation, we need to model each block separately and then link them according to the PLL architecture in Fig. 5. The noise considered in this case study is the random temporal variation of the phase (a.k.a jitter) in the reference oscillator and the VCO block. It is well-known that jitter is the most dominant and critical noise metric in PLL because large jitter can modulate the oscillator signal both in frequency and amplitude. These modulation effects can cause a deviation in the phase from targeted locking range and hence results in a design failure. The efficient verification of PLL for a certain design specification has always been a challenge for circuit designers. We apply the proposed methodology to verify the locking property of a second order PLL design shown in Fig. 5. The lock time property is a safety property that expresses how fast the frequency synthesizer switches from one frequency to another. The verification of this property is achieved by checking that the PLL reaches the proper DC value within the lock time parameter range which is $\in [0.002, 0.0024]$ seconds.

This property is defined within the ambit of an SSRE model in Eq. (7), where the SSRE concatenation operator ($\wedge$) indicates that the two Boolean expressions hold simultanuously.

$$Property\_PLL = If(Filter\_out(Lock\_time_{min} + n) \in DC\_level\_range \wedge \quad (7)$$
$$Filter\_out(Lock\_time_{max} - n) \in DC\_level\_range, true, false)$$

The verification property is *For a given confidence level* α, *and* N *Monte Carlo trials, what is the probability that the PLL meets the lock-time requirement?*.

In this case, the PLL has been designed with a lock-time in the range of [0.002 , 0.0024] sec. Hence, the null hypothesis H_0 and the alternative hypothesis H_1 of the Property in Eq. (7) can be, respectively, expressed as:

$$H_0 : lock\_time \in [0.002, 0.0024]$$
$$H_1 : lock\_time \notin [0.002, 0.0024]$$

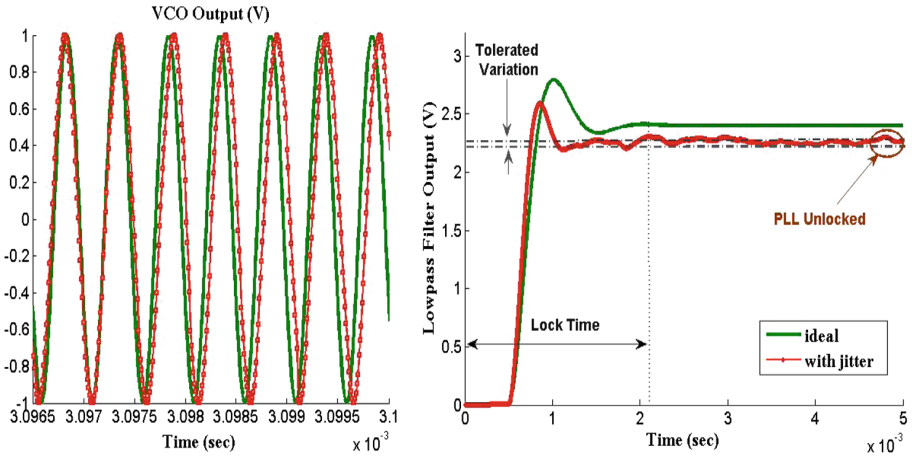

Fig. 6. PLL Output with and without phase noise (Color figure online)

Figure 6 depicts a comparison between the locking property of the PLL design whose parameter values are listed in Table 2 with and without jitter. A comparision of the same reachability algorithm without backward refinement [20] for the PLL design is given in [17]. It can be remarked that in the case of jittery PLL (red dotted line), the low pass filter outputs do not stabilize to the tolerated DC level and keep fluctuating outside the tolerated range. As a result, the PLL locking property is violated and the verification fails. Therefore, the verification of the PLL with consideration of jitter is very important when performing reachability analysis. Now, we validate our proposed intertwined forward/backward reachability technique on the jittery PLL design for an entire range of initial conditions and with consideration of parameter variations. The derived forward and backward reachable bounds are shown in Fig. 7, in which the forward reachability bound is painted in red and the backward reachability bound in green. In the forward iteration, the reachable set is highly over-approximating the PLL behavior. By performing the backward correction, we were able to tighten up this over-approximation and trace back the circuit dynamics down to the initial condition. The results of the PLL yield estimation using a variant of statistical Monte Carlo

Table 2. PLL frequency synthesizer parameters

Name	Value	Unit
RC	0.0001	s
α	$exp(\frac{-10^8}{0.0001})$	-
V_c	5	V
ω_0	$\pi \times 10^6$	rad.Hz
ω_{vco}	$2\pi \times 10^6$	rad.Hz
K_{vco}	$\frac{2\omega_{vco}}{V_c}$	rad.Hz
DC_{level}	2.5	V

technique [21] called Monte Carlo-Jackknife (MC-JK) and our proposed intertwined reachability technique are summarized in Table 3. It is worth mentioning that our technique converges in one iteration only while Monte Carlo technique requires thousands of runs. From Table 3, it can be noticed that our proposed method finds a lower yield percentage compared to the statistical Monte Carlo scheme in [21]. This can be explained by the fact that our verification approach can weed out PLL locking failures that were not covered in [21].

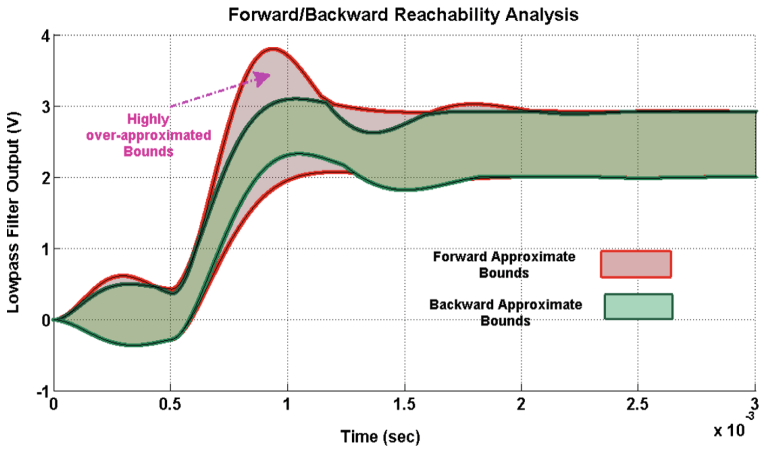

Fig. 7. Intertwined forward/backward reachability analysis of PLL under jitter (Color figure online)

In addition, the presence of combined jitter, initial conditions and process variations (Columns $8-10$) have substantially decreased the PLL yield, meaning the PLL presents more probability of lock failure.

The presence of jitter alone has shown a lower yield rate. This can be justified by the high sensitivity of the VCO block to jitter. The failure of the PLL is not

Table 3. Verification results for the PLL Lock-Time property

N=	Phase noise only			Parameter variation only			Phase noise & P.V		
	[21]	Our method	RE	[21]	Our method	RE	[21]	Our method	RE
	Yield (%)	Yield (%)	(%)	Yield (%)	Yield (%)	(%)	Yield (%)	Yield (%)	(%)
1000	82.4	74.1	8.3	84.7	79.2	5.5	80.6	71.5	9.1
	83.3	71.7	11.6	80.9	76.3	4.6	78.2	68.9	9.3
	81.7	69.8	11.9	79.2	72.7	6.5	77.5	67.3	10.2
5000	83.6	73.1	10.5	85.8	81.6	4.2	81.8	72.3	8.7
	80.2	72.3	7.9	81.9	77.8	4.1	78.2	70.1	8.9
	79.8	70.8	9	80.7	74.4	6.3	78.2	68.6	9.6
10000	81,7	69.9	11.8	83.6	79.7	3.9	80.2	66.1	14.1
	79.6	67.1	12.5	80.3	74.4	6.1	78.1	62.6	15.3
	78.1	65.9	12.2	81.9	71.8	10.1	76.8	60.1	16.7

due to lock up (non oscillation) of the VCO but, due to either an "ugly" (i.e., fluctuates outside the tolerated region) or delayed oscillation.

The Relative Error (RE) between our proposed approach and the MC technique (Columns 4, 7 and 10) becomes more pronounced when the number of Monte Carlo trials is increased due to the high MC sampling variance.

5 Conclusion

This paper presents a novel methodology for modeling and verification of nonlinear hybrid systems by computing reachable sets of possible state-space trajectories in the presence of uncertainties. In contrast to methods that use solely forward reachability, the refinement of the reachable state space is carried out in an intertwined forward/backward manner. The resulting set, which contains all periodic and aperiodic time bounded behaviors of the system under parameter variation and initial condition disturbance, can be used to verify critical properties such as bounds on voltages, currents, and cycle time (frequency) of embedded designs. Statistical verification based on hypothesis testing is then conducted on the resultant corrected reachable sets for an accurate parametric system failure estimation. Experimental results show that our intertwined forward/backward reachability analysis can succeed in accurately estimating the system failure rate (a.k.a yield) by reducing the highly over-approximation of the forward scheme in the presence of noise and process variations. Experimental results of a second order PLL application, our algorithm outperforms existing methods by providing up to 17% more reliable yield estimation of the locking time property. However, the computational cost of the proposed methodology highly increases with the number of process parameters and system properties to be verified. In our future research, we will further investigate the possibility of adopting efficient heuristics and parallelization techniques that may address the computational time issue. We plan to verify complex systems in presence of transient faults uncertainty [22] and that involve multiple performance metrics.

References

1. Lin, H., Antsaklis, P.J., et al.: Hybrid dynamical systems: an introduction to control and verification. Found. Trends Syst. Control **1**(1), 1–172 (2014)
2. Wolf, M.: High-performance embedded computing: applications in cyber-physical systems and mobile computing (2014)
3. da Silva Azevedo, L., Parker, D., Walker, M., Papadopoulos, Y., Araujo, R.E.: Assisted assignment of automotive safety requirements. IEEE Softw. J. **31**(1), 62–68 (2014)
4. Bouissou, M., Elmqvist, H., Otter, M., Benveniste, A.: Efficient Monte Carlo simulation of stochastic hybrid systems. (96), pp. 715–725 (2014)
5. Fränzle, M., Hermanns, H., Teige, T.: Stochastic satisfiability modulo theory: a novel technique for the analysis of probabilistic hybrid systems. In: International Workshop on Hybrid Systems: Computation and Control, pp. 172–186 (2008)
6. Fulton, N., Mitsch, S., Quesel, J.-D., Völp, M., Platzer, A., KeYmaera, X.: An axiomatic tactical theorem prover for hybrid systems. In: International Conference on Automated Deduction, pp. 527–538 (2015)
7. Adimoolam, A.S., Dang, T.: Template complex zonotopes: a new set representation for verification of hybrid systems. In: International Workshop on Symbolic and Numerical Methods for Reachability Analysis, pp. 1–2 (2016)
8. Milstein, G.N.: Numerical integration of stochastic differential equations, vol. 313 (1994)
9. Al-Sammane, G.: Simulation symbolique des circuits décrits au niveau algorithmique. Ph.D. thesis, Université Joseph-Fourier-Grenoble I, France (2005)
10. Kumar, P.R., Varaiya, P.: Stochastic systems: Estimation, identification, and adaptive control (2015)
11. Ikeda, N., Watanabe, S.: Stochastic differential equations and diffusion processes, vol. 24 (2014)
12. Revuz, D., Yor, M.: Continuous martingales and Brownian motion, vol. 293 (2013)
13. Milstein, G.N., Tretyakov, M.V.: Stochastic numerics for mathematical physics (2013)
14. Han, Y., Chung, C.Y., Wong, K.P., Lee, H.W., Zhang, J.H.: Probabilistic load flow evaluation with hybrid latin hypercube sampling and cholesky decomposition. IEEE Trans. Power Syst. **24**(2), 661–667 (2009)
15. Burrage, K., Burrage, P., Donovan, D., Thompson, B.: Populations of models, experimental designs and coverage of parameter space by Latin hypercube and orthogonal sampling. Procedia Comput. Sci. **51**, 1762–1771 (2015)
16. Martinez, W.L., Martinez, A.R.: Computational Statistics Handbook with MATLAB. CRC Press, Boca Raton (2007)
17. Seghaier, I., Tahar, S.: Intertwined global optimization based reachability analysis of analog and mixed signal designs; Technical report, Department of Electrical and Computer Engineering, Concordia University, June 2017. http://hvg.ece.concordia.ca/Publications/TECH_REP/IGO_TR17.pdf
18. Bonarini, A., Bontempi, G.: A qualitative simulation approach for fuzzy dynamical models. ACM Trans. Modeling Comput. Simul. **4**(4), 285–313 (1994)
19. Coleman, T., Branch, M.A., Grace, A.: Optimization toolbox. For Use with MATLAB. Users Guide for MATLAB 5, Version 2, Release II (1999)
20. Seghaier, I., Aridhi, H., Zaki, M.H., Tahar, S.: A qualitative simulation approach for verifying PLL locking property. In: Great Lakes Symposium on VLSI, pp. 317–322 (2014)

21. Seghaier, I., Zaki, M.H., Tahar, S.: Statistically validating the impact of process variations on analog and mixed signal designs, pp. 99–102 (2015)
22. Hamad, G.B., Kazma, G., Mohamed, O.A., Savaria, Y.: Efficient and accurate analysis of single event transients propagation using SMT-based techniques. In: International Conference on Computer-Aided Design, pp. 1–7 (2016)