Formal Methods in System Design manuscript No.
(will be inserted by the editor)

Verifying Global Start-Up for a Möbius Ring-Oscillator

Chao Yan · Mark R. Greenstreet · Suwen Yang

June 13, 2013

Abstract This paper presents the formal verification of start-up for a differential ring-oscillator circuit used in industrial designs. We present an efficient algorithm for finding DC equilibria to establish a condition that ensure the oscillator is free from lock-up. Further, we present a formal verification solution for the problem. Using dynamical systems theory, we show that any oscillator must have a non-empty set of states from which it fails to start properly. However, it is possible to show that these failures only occur with zero probability. To do so, this paper generalizes the "cone argument" initially presented in [Mitchell and Greenstreet, 1996] and proves the soundness of this generalization. This paper also shows how concepts from analog design such as differential operation can be soundly incorporated into the verification to produce simpler models and reduce the complexity of the verification task.

1 Introduction

System-on-Chip (SoC) and analog-mixed-signal (AMS) designs have created new challenges for analog circuit designers. Typical analog design relies heavily upon simulation tools such as HSPICE and Spectre®. Long simulation times along with the continuous nature of device parameters, operating conditions and input waveforms mean that simulation tools can only provide partial verification of analog designs. In practice, designers typically focus their simulation efforts on parametric and small-signal sensitivity analysis when the circuit is in or near its intended operating mode. Such analysis can be used to determine the gain and bandwidth of an amplifier, the phase transfer function of a phase-locked loop, or to find transistor sizes to optimize a given circuit topology. However, simulations cannot show that the circuit will eventually reach its intended operating condition from all possible starting conditions.

This paper presents a rigorous, formal verification that a commonly used differential ring-oscillator circuit correctly starts oscillation with probability 1. As shown in Figure 1, the oscillator consists of n stages, where each stage has a pair of "forward" inverters (labeled fwd

This work was supported in part by grants from Intel, Oracle, and the Natural Sciences and Engineering Research Council of Canada.

Intel · University of British Columbia · Oracle

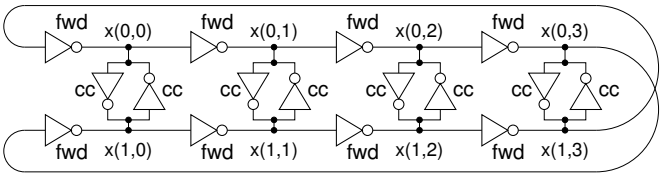


Fig. 1: The Möbius Ring-Oscillator

in the figure) and a pair of "cross-coupling" inverters (labeled cc). If the forward inverters are much larger than the cross-coupling inverters, then the circuit acts like a ring of four inverters and settles to one of two states:

$$\begin{aligned} &\text{State 1: } \mathsf{x(0,0)}, \mathsf{x(1,0)}, \mathsf{x(0,2)}, \mathsf{x(0,2)} \text{ are high, and} \\ &\qquad \mathsf{x(0,1)}, \mathsf{x(1,1)}, \mathsf{x(0,3)}, \mathsf{x(0,3)} \text{ are low.} \\ &\text{State 2: } \mathsf{x(0,1)}, \mathsf{x(1,1)}, \mathsf{x(0,3)}, \mathsf{x(0,3)} \text{ are high, and} \\ &\qquad \mathsf{x(0,0)}, \mathsf{x(1,0)}, \mathsf{x(0,2)}, \mathsf{x(0,2)} \text{ are low.} \end{aligned} \tag{1}$$

Conversely, if the cross-coupling inverters are much larger than the forward ones, then the circuit acts like four separate static latches and has sixteen stable states. If the forward and cross-coupling inverters have comparable strength, then the circuit should oscillate in a stable fashion.

The oscillator circuit is often referred to as a "Möbius" oscillator due to the "twist" in the connection from the last stage back to the first stage. It can be implemented with any *even* number of stages. If the oscillator is implemented with an *odd* number of stages, then the state where $x(i, j)$ is low for $i + j$ even and high for $i + j$ odd (or vice-versa) is stable regardless of the transistor sizes. A differential oscillator with an odd number of stages can be implemented by eliminating the twist in the connection from the last stage back to the first.

In 2008, researchers from Rambus posed the problem of showing that the oscillator circuit shown in Figure 1 starts from all initial conditions for a particular choice of transistor sizes [Jones et al., 2008]. They described this as a "real-world" problem noting that oscillators of this type have been used in real chips and have been observed to fail in the test-lab. They posed a further problem of determining the range of transistor sizes for which proper start-up is guaranteed. This paper presents solutions to these problems. The original problem as posed in [Jones et al., 2008] used two-stages as a simplest case. We have seen a published versions of the design with up to 128 stages [Xiao et al., 2001].

1.1 Contributions

This paper presents two approaches for verifying start-up of ring-oscillator circuits. First, we show how to locate all equilibrium points of the ring-oscillator circuit and test each point for stability. A correct oscillator should have no stable equilibrium points. In Section 3 we derive an efficient algorithm for finding DC equilibria to check if the oscillator is free from lock-up. This approach is fast, can readily handle oscillators with a large number of stages, and can be applied to a wide variety of ring-oscillator structures.

Showing that an oscillator has no stable equilibrium points does not prove that it will start correctly. The oscillator could have a stable harmonic mode or other behaviours. To

address this problem, we prove in Section 4 that no physically plausible oscillator can be guaranteed to start from *all* initial states. We show that the failure set can be arbitrarily small, and in Section 5 we present a new method to show that this failure set corresponds to a failure probability of zero. We also introduce a symmetry reduction method that allows us to exploit the differential operation of the oscillator in a formal verification context. Section 6 describes our implementation of the verification method using Matlab and Coho [Yan and Greenstreet, 2008]. Section 7 presents the results of verifying the oscillator circuit with these methods. To the best of our knowledge, this is the first formal solution to the problem as posed by the researchers at Rambus: "Will the oscillator start correctly from all initial conditions?" Our answer is: "No. But, if the circuit is designed properly, it will start with probability one."

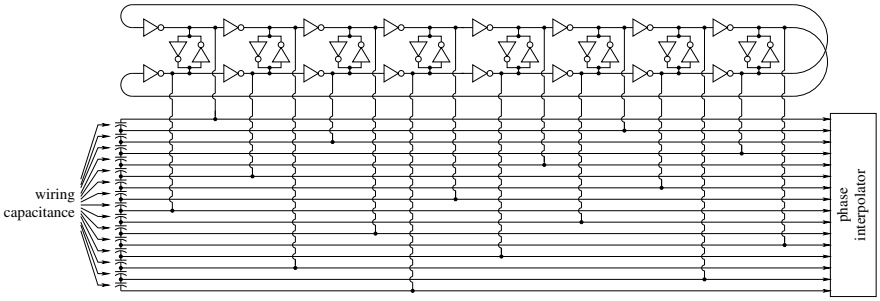
2 Related Work

Oscillator circuits have been a popular example for applying formal methods to analog circuit verification [Frehse et al., 2006; Gupta et al., 2004; Hartong et al., 2002; Little et al., 2006]. These early papers focused on simple oscillators, such as a tunnel-diode based design, that are not representative of the oscillator circuits used in real VLSI designs. More recently, several groups have reported results for the Möbius oscillator problem described above.

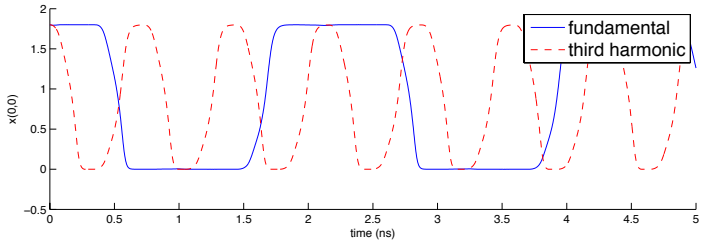
The earliest attempted verification of the oscillator that we have seen [Xiao et al., 2001] predates the formulation by [Jones et al., 2008] and considers the behaviours of a 128 stage oscillator for a pulse-width modulated voltage regulator. Their "proof" of correct operation assumes differential and periodic operation, and does not consider weak coupling between stages (e.g. due to power supply noise), that could stabilize undesired, higher harmonic modes of oscillation. Instead, they argue that random disturbances will cause higher harmonic modes to decay, leaving only the fundamental mode. However, unintended coupling between the oscillator stages can violate this assumption. As an example, Figure 2a shows an eight-stage oscillator whose outputs are connected through a bus to a phase interpolator. The bus wiring is chosen for this example so that each wire for the output of each forward inverter is adjacent to those for the inverters five earlier and five later in the cycle of forward inverters. By choosing different initial conditions, this oscillator can settle either into the intended, fundamental mode, or into a third-harmonic mode as shown in Figure 2b.

In [Greenstreet and Yang, 2008], we presented a partial response to the questions posed by the Rambus researchers. That work considered the problem of finding stable DC equilibrium points. An oscillator state is a *DC-equilibrium* if the time-derivatives for all node voltages at that point are zero. The equilibrium point is *stable* if small perturbations from that state produce trajectories that asymptotically return to the state. An absence of DC-equilibria is necessary condition for ensuring that an oscillator starts from all initial states. However, it is not a sufficient condition. As illustrated above, a circuit could also have stable harmonic modes or various non-periodic behaviours that would preclude proper start-up from all states.

Following [Greenstreet and Yang, 2008], several subsequent papers have also treated the verification problem as one of ruling out the existence of DC equilibria. For example, a SAT solve is used in [Tiwari et al., 2008, 2010] to identify DC equilibria. To find *stable* equilibria, they added constraints that at least one node of the circuit must be within 0.2 volts of power or ground. They did not state how they had arrived at these extra constraints or whether or not they can be shown to be sound.



(a) An eight-stage oscillator driving a bus



(b) Fundamental and harmonic modes for the oscillator

Steinhorst *et al.* [Steinhorst et al., 2009] presented a particle filtering approach to improve simulation coverage. They automatically run simulations from a large number of initial conditions from a uniform grid over the state space. The end points of these trajectories are plotted as arrows indicating the direction of the time-derivative vector at these points. These plots allow a simple visualization of the circuit's behaviour. The authors show how stable equilibria for a two-stage Möbius oscillator can be identified from such plots. While the technique provides an intuitively appealing visualization of the circuit behaviour, the coverage is exactly the same as that of the simulations performed, and failures could escape the verification process. Particle filtering was also proposed in [Kim et al., 2009] as a way of visualizing start-up anomalies of analog circuits such as oscillators.

More recently, Steinhorst and Hedrich presented a model-checking technique where the continuous state-space is discretized according to the flows of trajectories [Steinhorst and Hedrich, 2012]. They used the model checker to show two properties of a two-stage Möbius oscillator: (1) it has no stable DC-equilibria; and (2) that it does have a stable, periodic trajectory. These are essentially the DC-equilibrium properties that we verify in Section 3. They do not check for freedom from harmonic modes or other dynamic failure as we describe in Sections 5 through 7. They report run-times of 10-13 minutes for a two-stage oscillator. These are comparable to the run-times we report for a *complete* verification and much longer than the run-times of our method for verifying the absence of stable DC equilibria. They only considered at two-stage oscillator; it is not clear if their approach could scale to handle larger circuits.

Zaki *et al.* [Zaki et al., 2009] presented an approach where the "pencil-and-paper" analysis from [Greenstreet and Yang, 2008] was automated using HySAT [Fränzle, 2007] and Matlab toolboxes for interval arithmetic and matrix pseudo-spectrum calculations. This approach could find all DC equilibria for some simple circuits but failed for the Möbius os-

cillator. Recently, we have tried formulating the search for equilibria using the Z3 SMT solver Moura and Bjørner [2008]. This required approximating the circuit dynamics with polynomial functions, but even with those simplifications, the SMT solver did not complete its computation after over one day.

Little *et al.* [Little and Myers, 2008] showed that trajectories in a neighbourhood of the nominal periodic trajectory for the oscillator remain close to that nominal trajectory. This replaces the small-signal analysis of traditional analog design with linear hybrid Petri net (LHPN) model checking and confirms the stability of the desired oscillating behaviour. As the analysis only considers a portion on the state space near the desired trajectory, it does not verify proper start-up for all initial conditions.

3 DC Equilibrium Analysis

A necessary, but not sufficient condition for correct start up is that the oscillator should have no stable DC-equilibria (i.e. fix-point attractors). This does not guarantee correct operation; for example, the circuit could have a stable oscillation at or near a harmonic of the intended oscillation frequency, or it could have a chaotic attractor. However, a test for DC lock-up catches many of the start-up failures of real oscillator circuits. Furthermore, the approach presented here is computationally tractable, even for oscillators with a large number of stages, and can be easily extended to apply to other oscillator designs. Our analysis proceeds in two phases. First we describe a method for locating all DC-equilibria of a ring-oscillator circuit as depicted in Figure 1. We then analyze the stability of each equilibrium point. If all equilibria are unstable, then with probability 1, the circuit will not lock-up at a stationary state.

3.1 Finding DC Equilibria

An n-stage Möbius ring-oscillator has $2n$ nodes. We model MOSFETs as voltage controlled current sources with associated capacitances, and we model the interconnect in the oscillator as additional capacitances. For simplicity, we ignore wire resistance, noting that such resistances should be small compared with the transistor resistances for any reasonably compact layout of the circuit. Likewise, we ignore inductances. The state of the circuit is given by a vector of $2n$ elements, corresponding to the voltages on the $2n$ nodes of the circuit. A state is a DC equilibrium if the total current flowing into each node through the transistors is zero.

A brute-force analysis of the ring-oscillator would require searching a $2n$-dimensional space for DC equilibria. In addition to being time consuming, it would be difficult to show that such a search was exhaustive. Instead, we first show how we can simplify the problem to the search of a 2-dimensional space, and then further reduce it to a 1-dimensional search problem. This formulation makes the identification of DC equilibria straightforward.

Our analysis is based on simple observations of the monotonicity of the drain source current of a MOSFET. For simplicity, we assume that all N-channel devices have their bodies connected to a constant voltage, namely ground, and that all P-channel devices have their bodies connected to V_{dd}. We write $I_{ds}(v_s, v_g, v_d)$ to denote the drain source current of a transistor with source, gate and drain voltages of v_s, v_g and v_d respectively. The monotonicity property that we assume is:

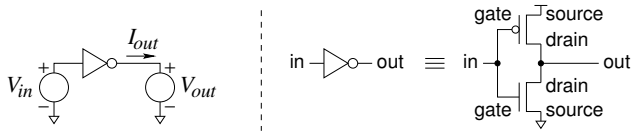


Fig. 2: An Inverter

Proposition 1 (Monotonic Drain Source Current) *The drain-to-source current of a MOSFET, $I_{ds}(v_s, v_g, v_d)$, is positive monotonic in the voltages v_g and v_d and negative monotonic in the voltage v_s.*

Proposition 1 holds for both N-channel and P-channel devices. Furthermore, it is based on basic physical properties of the transistors and should continue to hold in foreseeable CMOS fabrication processes.

Figure 2 shows a CMOS inverter. Let $I_{inv}(V_{in}, V_{out})$ denote the output current of such an inverter with input voltage V_{in} and output voltage V_{out}. For simplicity, we ignore gate leakage current; extending the analysis that we present below to include gate leakage is straightforward but would clutter the presentation. Thus,

$$I_{inv}(V_{in}, V_{out}) = -(I_{ds,n}(0, V_{in}, V_{out}) + I_{ds,p}(V_{dd}, V_{in}, V_{out})) \tag{2}$$

where $I_{ds,n}$ and $I_{ds,p}$ are the I_{ds} functions for the N-channel and P-channel transistors respectively. At a DC equilibrium, there is no current flowing into or out of the node capacitances, and we assume that the wires have no leakage, a very reasonable approximation. Thus, the current flowing out of the inverter is the negation of the sum of the drain-source currents for the two transistors that make up the inverter. We conclude:

Lemma 1 (Monotonic Inverter Current) *At a DC equilibrium, $I_{inv}(V_{in}, V_{out})$ is negative monotonic in both V_{in} and V_{out}.*

Proof The claim follows directly from Proposition 1 and the structure of the inverter.

Because the I_{inv} function is monotonic, its inverses are also functions. In particular, we define $in_{inv}(V_{out}, I_{out})$ to be the input voltage to the inverter such that if the output voltage is V_{out}, then the output current will be I_{out}. In other words:

$$I_{inv}(in_{inv}(V_{out}, I_{out}), V_{out}) = I_{out}. \tag{3}$$

To simplify the presentation, we will assume that all forward inverters have the same transfer function, and likewise that all cross-coupling inverters have the same transfer function. This assumption is only to avoid clutter in the presentation; at the end of this section, we show how these assumptions can be removed. Let $I_{inv,fwd}$ and $I_{inv,cc}$ denote these two transfer functions, and let $in_{inv,fwd}$ denote the in_{inv} function for a forward inverter. For $i \in \{0,1\}$ and $0 \leq j < n$, let $x(i,j)$ denote the voltage on node $\times(i,j)$ as indicated in Figure 1. An assignment of voltages, x, is a DC equilibrium iff

$$\begin{aligned} &\forall j \in \{0 \ldots n-1\}. \\ &\quad I_{inv,fwd}(x(0,j), x(0, j \oplus_n 1)) + I_{inv,cc}(x(0, j \oplus_n 1), x(()1, j \oplus_n 1) = 0 \\ &\wedge\ I_{inv,fwd}(x(1,j), x(1, j \oplus_n 1)) + I_{inv,cc}(x((0, j \oplus_n 1), x((1, j \oplus_n 1)) = 0 \end{aligned} \tag{4}$$

where $j\oplus_n 1$ denotes $(j+1)$ mod n. Let

$$back(x(0,j),x(1,j)) = \\ \left(in_{inv,fwd}(x(0,j),\ -I_{inv,cc}(x(1,j),x(0,j))),\ in_{inv,fwd}(x(1,j),\ -I_{inv,cc}(x(1,j),x(0,j)))\right). \tag{5}$$

In English, given the voltages on the outputs of an oscillator stage, *back* calculates the voltages that must be present on the inputs of the stage if the outputs are to be in DC equilibrium. We can use *back* repeatedly to work backwards all the way around the oscillator ring. In particular, if $back^n(x,y) = (y,x)$, then we've found a DC equilibrium. Otherwise, we can use monotonicity properties of the oscillator circuit to guide our search for the equilibrium. This produces an efficient search that finds all DC equilibria. We can now state two key lemmas:

Lemma 2 (DC Equilibria) *There is a DC equilibrium for an oscillator circuit with n nodes where nodes* $\times(0,n-1)$ *and* $\times(1,n-1)$ *have voltages* $x(0,n-1)$ *and* $x(1,n-1)$ *iff*

$$back^n(x(0,n-1),x(1,n-1)) = (x(1,n-1),x(0,n-1)).$$

Proof:

$\Rightarrow$

For $0 \le j < n-1$, let

$$(x(0,j),x(1,j)) = back^{n-1-j}(x(0,n-1),x(1,n-1)).$$

It is straightforward to show that this choice of node voltages satisfies Equation 4.

$\Leftarrow$

Assume otherwise. Then there must be some $j \in \{0 \ldots n\}$ such that

$$(x(0,j),x(1,j)) \neq back(x(0,j\oplus_n 1), x(1,j\oplus_n 1)).$$

Equation 4 is violated for nodes $x(0,j\oplus_n 1)$ and $x(1,j\oplus_n 1)$.

Lemma 3 (Monotonicity of back) *Let* $(x'(1,n-1),x'(0,n-1)) = back^n(x(0,n-1),x(1,n-1))$. *Then both* $x'(0,n-1)$ *and* $x'(1,n-1)$ *are positive monotonic in both* $x(0,n-1)$ *and* $x(1,n-1)$.

Proof The function *back* is negative monotonic in both of its arguments which follows immediately from Lemma 1 and Equation 5 (the definition of *back*). Accordingly, $back^2$ is positive monotonic in both of its arguments. For any Möbius ring oscillator, n is even. Let $n = 2m$. $back^n = back^{2m}$ and is positive monotonic in its arguments.

Lemma 2 reduces the space that must be searched to find DC equilibria from $2n$ dimensions down to two. Lemma 3 further reduces this to only one dimension. Thus, we can use these to implement an efficient search. Figure 3a gives pseudo-code for this search. The search at lines 2 and 3 is simple because of the monotonicity of $back^n$. We first try $x(1,n-1) = 0$ and then $x(1,n-1) = V_{dd}$ to determine whether or not there is a solution. If there is, then nearly any root finding algorithm can be used to find it. We use bisection on the value of $x(1,n-1)$; bisection is very robust, and it is fast enough for practical application.

Figure 4a illustrates this process. The plot is for a four-stage oscillator circuit in the TSMC 180nm, 1.8V CMOS process with the cross-coupling inverters 70% of the size of the forward inverters. The figure depicts the search for a value for $x(1,3)$ when $x(0,3)$ is assumed to have a value of 1.0 volts. The algorithm first checks $back^n$ with $x(1,3)$ equal to

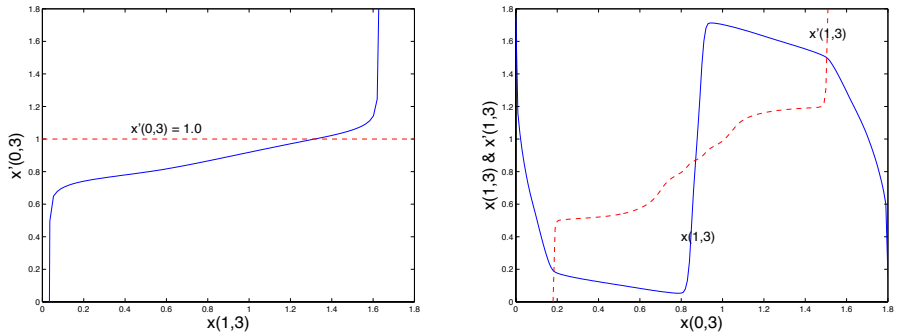
```
[ 1]  for x(0,n−1) := 0 to V_dd step δ_v do
[ 2]    find x(1,n−1) such that
[ 3]        second(back^n(x(0,n−1),x(1,n−1))) = x(0,n−1);
[ 4]    s = sign(x(1,n−1) − first(back^n(x(0,n−1),x(1,n−1))));
[ 5]    if((x(0,n−1) > 0) AND s ≠ sLast) then
[ 6]      search for back^n(u_0,v_0) = (v_0,u_0)
[ 7]          with u_0 ∈ [x(0,n−1) − δ_v, x(0,n−1)];
[ 8]      record equilibrium;
[ 9]    endif;
[10]    sLast = s;
[11]  od;
```

(a) DC Equilibrium Search Algorithm

```
[ 1]  Q := equilibrium points; /* see Fig. 3a */
[ 2]  for each q ∈ Q do
[ 3]    J := Jacobian operator at q;
[ 4]    Λ := eigenvalues(J);
[ 5]    μ := max_{λ∈Λ} Real(λ);
[ 6]    if(μ > 0) then
[ 7]      return(unstable);
[ 8]    endif;
[ 9]  od;
[10]  return(stable);
```

(b) Classifying DC Equilibria

Fig. 3: DC Equilibrium Analysis



(a) Searching for $x(1,3)$ so that $x'(0,3) = x(0,3)$. Shown with $x(0,3) = 1$.

(b) Searching for $x(0,3)$ to find DC equilibria

Fig. 4: Searching for DC-Equilibria

ground and $x(1,3)$ equal to V_{dd} to confirm the existence of a solution. Then, standard zero-finding methods (e.g. bisection and Newton's) are used to find the desired value for $x(1,3)$. For the example in the figure, a solution is found at $x(1,3) = 1.316$ with $back^4(1,1.316) = (0.9689,1)$. Thus, there is no equilibrium point with $x(0,3) = 1$.

Repeating the search described above for each candidate value of $x(0,n-1)$, we obtain a plot like the one shown in Figure 4b. Lines 5 and 6 of the algorithm (Figure 3a) identify intervals of $x(0,n-1)$ where the $x(1,n-1)$ and $x'(1,n-1)$ curves intersect. The search at lines 6 and 7 is constrained to these small intervals, and we use bisection to identify the values for $x(0,n-1)$ that give rise to DC equilibria. At the end of this computation, we have identified all DC equilibria for the oscillator circuit.

3.2 Testing Stability

To determine whether or not a DC equilibrium is stable, we construct the small-signal, transient model at each equilibrium point. With a slight abuse of notation, we will now write

$x(t) \in \mathbb{R}^{2n}$ to denote the state of the oscillator at time t (otherwise, we've been writing $x(i,j)$ to denote the voltage on node $\times(i,j)$). Likewise, we write x_j to denote the j^{th} component of x. This gives us a linear model for circuit behavior in a small neighborhood of the equilibrium:

$$\dot{x} \approx J_{eq}(x - x(eq)) \tag{6}$$

where $\dot{x}$ is the time derivative of x, $x(eq)$ is the voltage vector for the equilibrium point, and J_{eq} is the matrix representation of the linear approximation for $\dot{x}$ when x is near $x(eq)$. The solution to Equation 6 is

$$x(t_0 + t) = x(eq) + e^{tJ_{eq}}(x(t_0) - x(eq)). \tag{7}$$

If all of the eigenvalues of J_{eq} have negative real parts, then $e^{tJ_{eq}}$ goes to zero as t increases, and $V(t_0 + t)$ converges to $x(eq)$. Thus, such a DC equilibrium is stable (See [Hirsch and Smale, 1974] for an introduction to the dynamical systems theory). Conversely, if J_{eq} has any eigenvalues with positive real parts, then its DC equilibrium is unstable.

To model CMOS circuits, we apply Kirchoff's current law to get $I_m + I_c = 0$ where I_m is the vector of MOSFET currents flowing into each node and I_c is the vector of currents flowing into each node from the capacitances. Noting that $I_c = C\dot{x}$, where C is the matrix of internode capacitances, we get

$$\dot{x} = -C^{-1} I_m(x). \tag{8}$$

For simplicity, we assume that all capacitors are to ground and have constant values. The MOSFET currents are obtained by summing over the drain currents of all transistors with drains connected to the node (or, subtracting for a connection to a source).

To compute the Jacobian of $-C^{-1}I_m(x)$, we note that J is a matrix with

$$J(i,j) = \frac{\partial \dot{x}_i}{\partial x_j}. \tag{9}$$

Because C is constant, it suffices to compute the partial derivatives of $I_m(x)$ with respect to the components of x. In our implementation, we approximate I_m using bilinear interpolation over a table of values from HSPICE. Calculating the partial derivatives is straightforward.

Figure 3b shows how we combine these pieces to obtain an algorithm to verify that a Möbius ring-oscillator has no stable DC-equilibria.

3.3 Removing assumptions

In the analysis above, we made several assumptions to simplify the presentation. None of these are essential to the algorithm. This section summarizes the assumptions and describes how they can be removed.

Assumption all forward inverters have the same transfer function, and likewise for all cross-coupling inverters. Each transistor for each inverter could be chosen separately. In this case, each stage would have its own *back* function. Let $back_i$ be the *back* function for the stage with inputs $x(0,i)$ and $x(1,i)$. Define

$$\begin{aligned} Back_0(v_0, v_1) &= back_0(v_0, v_1) \\ Back_i(v_0, v_1) &= Back_{i-1}(back_i(v_0, v_1)),\ 1 \le i < n \end{aligned} \tag{10}$$

The analysis presented above can be readily extended using $Back_{n-1}$ instead of $back^n$. Because our approach does not rely on symmetry of the inverters, it can be used to examine

real-world designs with non-idealities in the layout or the impact of random, device-to-device parameter variation.

Assumption: all capacitors are to ground, and their values are independent of the node voltages. Capacitors do not affect the locations of the DC equilibrium points in the state space, but they could influence whether an equilibrium point is stable or unstable. Once a DC equilibrium is located, detailed circuit models could be used to extract the inter-node capacitances – simulators such as HSPICE provide such functionality. These capacitances could then be used in the eigenvalue calculations to test the stability (or not) of the equilibrium. Furthermore, pseudo-spectra techniques (e.g. [Rump, 2006]) can be used to determine if an unstable equilibrium is "close" to being stable, i.e. is the design susceptible to failure due to perturbations in the device parameters?

Assumption: gate-leakage is neglected. Given voltages for nodes $x(0,i)$ and $x(1,i)$ the gate-leakage currents can be computed along with the drain-source currents described above. These currents would change (very slightly, in practice) the current that must be provided by stage $i\ominus_n 1$ and could be incorporated into the *back* function, where $j\oplus_n 1$ denotes $(j-1) \bmod n$.

Assumption: interconnect resistance and inductance is ignored. The resistance of the interconnect that joins the drains of the N- and P-channel transistors in an inverter can be incorporated into the inverter model. The currents through the other wires are zero, if gate leakage is ignored, and very small otherwise. The voltage drops induced by these currents can be determined by solving the linear system arising by the application of Kirchoff's current laws. This system is positive definite; thus, it preserves the various monotonicity assumptions made above.

Inductors, like capacitors, do not affect the locations of the DC equilibrium points, but they could influence stability. As with capacitors, inductors can be included in the small-signal model used to check stability. By including inductors in this manner, we expect that the approach described here could be extended to LC oscillators in addition to the ring-oscillator structures considered here.

4 No Perfect Oscillator

The analysis in Section 3 can verify that an oscillator does not lock-up in a state where each node has a fixed voltage. This is a necessary condition for oscillator correctness, and from a designer's practical point of view, a very important condition. However, freedom from lock-up is not sufficient to prove that the oscillator will start properly. For example, there could be stable oscillations at a harmonic of the intended oscillation frequency or another frequency; or an oscillator circuit could exhibit chaotic behaviour from some initial conditions. This and the next two Sections show how proper start-up can be verified for a two-stage Möbius oscillator. First, we show that no physically plausible oscillator starts from all initial conditions. This was foreshadowed by the stability analysis in Section 3.2 where we showed that the circuit can have equilibrium points that are *unstable*: if the oscillator started exactly at one of those points, it would never leave. Furthemore, there can be initial conditions from which trajectories asymptotically approach such equilibria. This section shows that such initial conditions that lead to failure are inevitable for *any* physically plausible oscillator. Section 5.2 shows how such failures can be shown to be *negligible*: the set of points for which such failures occur have can zero volume in the state space of the oscillator, and thus, for reasonable probability distributions for the initial states, the probability of starting in

such as state is zero. Then, Section 6 describes how these mathematical results can be used to verify start-up of a Möbius oscillator.

4.1 Dynamical Systems and Oscillators

We assume that a circuit, such as an oscillator, is modeled by a system of ordinary differential equations. If the model has d variables, states of the circuit correspond to points in $\mathbb{R}^d$. The model includes a function, $f : \mathbb{R}^d \to \mathbb{R}^d$ that is the *time derivative* of the system: for state $\mathbf{x} \in \mathbb{R}^d$, $\dot{\mathbf{x}} = f(\mathbf{x})$ is the time-derivative of the system in state $\mathbf{x}$. For $\mathbf{x}_0 \in \mathbb{R}^d$, the *initial value problem* is to find a function $\mathbf{x} : \mathbb{R}^+ \to \mathbb{R}^d$ such that for all $t \geq 0$, $\frac{d}{dt}\mathbf{x}(t) = f(\mathbf{x}(t))$ and $\mathbf{x}(0) = \mathbf{x}_0$. Let $Q \subseteq \mathbb{R}^d$ be a closed set. Q is invariant with respect to f if all trajectories that start in Q remain in Q forever. To show that Q is invariant with respect to f, it is sufficient to show that for every $\mathbf{x} \in Q$ there is an $\varepsilon > 0$ such that $\mathbf{x} + \varepsilon f(\mathbf{x}) \in Q$. We impose two restrictions on f:

R1: There is a $Q \subseteq \mathbb{R}^d$ and some $K \in \mathbb{R}$ such that Q is invariant with respect to f and for every $\mathbf{x} \in Q$, $\|f(\mathbf{x})\| < K$.

R2: f is $\mathscr{C}^1$ in Q. This means that $f(\mathbf{x})$ is differentiable with respect to the components of $\mathbf{x}$, and these derivatives are continuous.

These two conditions guarantee the existence and uniqueness of solutions to the initial value problem for f and any $\mathbf{x}_0 \in Q$ (see [Hirsch and Smale, 1974, chap. 8.3]). We can define a function $\Phi_f(\mathbf{x}_0, t)$ such that if $\mathbf{x}$ is the solution to the initial value problem for f with $\mathbf{x}(0) = \mathbf{x}_0$, then $\mathbf{x}(t) = \Phi_f(\mathbf{x}_0, t)$. Given restrictions R1 and R2, $\Phi_f(\mathbf{x}_0, t)$ is a $\mathscr{C}^1$ function with respect to $\mathbf{x}_0$ and t for any $\mathbf{x}_0 \in Q$ and $t \geq 0$ (see [Hirsch and Smale, 1974, chap. 8.4]). We extend Φ_f to sets in the natural way: if $X \subseteq \mathbb{R}^d$, then $\Phi_f(X, t) = \{\mathbf{x}_2 | \exists \mathbf{x}_1 \in X.\ \mathbf{x}_2 = \Phi_f(\mathbf{x}_1, t)|\}$.

We assume that any physically plausible oscillator can be modeled by an ODE with f and Q satisfying restrictions R1 and R2. The requirement that f is $\mathscr{C}^1$ follows from the smoothness of the underlying physical models for electric fields, charge distributions, etc. The requirement of the existence of the set Q is satisfied because VLSI circuits generally have node voltages that are bounded by the voltages of ground and the power supply or that have limited excursions beyond these power supply voltages.

We now define "oscillation." If there is a $\mathbf{x}_0 \in Q$ and a $P > 0$ such that $\Phi_f(\mathbf{x}_0, P) = \mathbf{x}_0$, and for all $0 < t < P$, $\mathbf{x}(t) \neq \mathbf{x}_0$, then f has a solution with period P. In this case, we write $\Gamma_{f,\mathbf{x}_0} = \{\mathbf{x} | \exists t \in [0, P].\ \mathbf{x} = \Phi_f(\mathbf{x}_0, t)\}$ to denote the set of points in this periodic orbit. It is straightforward to show $\forall t > 0.\ \Phi_f(\Gamma, t) = \Gamma$. Let $J = Jac\, \Phi_f(\mathbf{x}_0, P)$, i.e., J is the matrix of partial derivatives of $\Phi_f(\mathbf{x}_0, P)$ with respect to $\mathbf{x}_0$. If J has $d-1$ eigenvalues with magnitude less than 1, then the periodic solution for $\mathbf{x}_0$ is a *periodic attractor* [Hirsch and Smale, 1974, Theorem 13.2]. Note that J has an eigenvalue of 1 whose corresponding eigenvector is $f(\mathbf{x}_0)$ – this corresponds to perturbing the trajectory along its orbit. We say that a system is an oscillator with period P if it has a periodic attractor with period P.

4.2 Oscillator Start-Up

First consider the set of possible initial states. Labeling one terminal of the power supply as "ground" and the other as "V_{dd}" is simply a designer convention. Depending on circuit details, the node voltages on power-up may be arbitrary values. Rather than trying to analyse

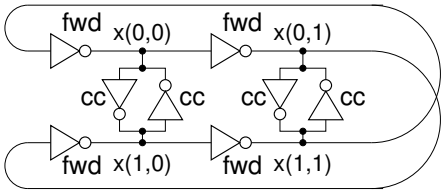


Fig. 5: A Two-Stage, Differential Ring-Oscillator Example

the circuit in detail, we simply assume that each node has an arbitrary initial voltage in $[V_{lo}, V_{hi}]$; typically V_{lo} is ground or close to ground; likewise, V_{hi} is close to V_{dd}. Let $X_0 = [V_{lo}, V_{hi}]^d$ denote the set of initial node voltages. Because X_0 contains all reachable states of the circuit, we assume $\gamma \subseteq X_0 \subset Q$, where γ is the desired periodic attractor of the oscillator. We can now describe an ideal oscillator.

A d-dimensional dynamical system with time-derivative function f is an *ideal oscillator* iff

The system is physically plausible: There is a set $Q \subseteq \mathbb{R}^d$ such that f and Q satisfy conditions R1 and R2.

Periodic behavior: The system has a periodic attractor. Let Γ be the orbit associated with this attractor.

Start up: There is a convex set $X_0 \subseteq \mathbb{R}^d$ of initial states such that $\Gamma \subseteq X_0 \subseteq Q$ and for every point $\mathbf{x}_0 \in X_0$ and every $\varepsilon > 0$, there is a $t > 0$ and a point $\mathbf{x}_1 \in \Gamma$ such that $\|\mathbf{x}_1 - \Phi(\mathbf{x}_0, t)\| < \varepsilon$.

The first two conditions were described in the previous section. The last condition states that the set of initial states must contain the periodic orbit as described above, and that for any initial state, the trajectories emanating from that state must eventually be arbitrarily close to the periodic orbit. The requirement that this initial set be convex reflects the topological properties of sets such as $[V_{lo}, V_{hi}]^d$ described above. We believe that this definition of an ideal oscillator captures the notion of the oscillator starting from all initial conditions requested in [Jones et al., 2008].

Theorem 1 *There is no ideal oscillator.*

Proof This follows directly from the property that solutions of ODEs that satisfy properties R1 and R2 are continuous in their initial conditions. Thus, the topology of the initial set, X_0, is preserved by $\Phi_f(X_0, t)$. However, any small neighborhood of a periodic attractor must have genus 1 (be "torus-like") whereas the set of initial states has genus 0 (i.e. it is "sphere-like"). Thus, it is not that case that all initial conditions lead to trajectories that are arbitrarily close to the desired attractor. This establishes the claim.

5 Formal Verification Outline

In this section and the next section, we present our formal verification solution for the question "Will the oscillator start up from all initial conditions?" We use the two stage ring oscillator as shown in Figure 5 as an example for our formal solution. Our verification proceeds in three main phases:

Differential Operation The oscillator shown in Figure 5 is a differential design: nodes $\times(0,0)$ and $\times(0,1)$ form a "differential pair" and likewise for nodes $\times(1,0)$ and $\times(1,1)$. The first phase of the verification shows that each of these differential pairs can be treated as a single signal.

Escape from the Failure Set As shown in Section 4, for any oscillator, there must be initial conditions from which it does not properly start. The second phase of the verification shows that this occurs with probability zero.

Proper Oscillation The first two phases show that all but a negligible subset of the initial conditions lead to a fairly small subset of the full state space. In the final phase, we use existing reachability methods to show that the oscillator starts up properly from the region.

This section describes the dynamical systems issues associated with each of these phases. Section 6 describes our verification method based on these observations.

We model the oscillator circuit from Figure 5 using non-linear ordinary differential equations (ODEs) obtained by standard, modified nodal-analysis. This gives us an equation of the form:

$$\dot{\mathbf{x}} = f(\mathbf{x}) \tag{11}$$

where $\mathbf{x}$ is a vector of node voltages, $\dot{\mathbf{x}}$ is the vector of time derivatives for these voltages, and f is the function modeling the non-linear dynamics of this circuit. Let d be the dimensionality of $\mathbf{x}$. We assume that f is C^1 which guarantees that Equation 11 has a unique solution for any initial state, $\mathbf{x}(0)$. For simplicity, we model the system as being autonomous (no inputs or outputs). Inputs (e.g. to model VCO control inputs, power supply noise), can be modeled by giving f additional parameters, i.e. $f(\mathbf{x},\mathbf{in})$.

5.1 Differential Behaviour

For $j \in \{0,n\}$, nodes $\times(0,j)$ and $\times(1,j)$ in the oscillator from Figure 5 form a "differential pair". The *differential component* of the differential pair is $x(0,j)-x(1,j)$, and $x(0,j)+x(1,j)$ is the *common mode* component. When the oscillator is operating properly, the common mode components are roughly constant, and the oscillation is manifested in the differential components. Let V_0^+ be the nominal value for the common mode components. We show that for a relatively small V_{err} if $|x(0,j)+x(1,j)-V_0^+| > V_{err}$, then $\frac{d}{dt}(x(0,j)+x(1,j))$ and $(x(0,j)+x(1,j)-V_0^+)$ have opposite signs. This shows that that the common mode component for nodes $\times(0,j)$ and $\times(1,j)$ converges to within V_{err} of the nominal value.

5.2 Escape from the Failure Set

Theorem 1 shows that there is no perfect oscillator. For the Möbius ring-oscillator, there is an equilibrium point, $\mathbf{x}_{fail}$, i.e. a point where $\dot{\mathbf{x}} = 0$, and there is a manifold, X_{fail} such that

$$\forall \mathbf{x} \in X_{fail}.\ \lim_{t\to\infty} \|\Phi_f(\mathbf{x},t) - \mathbf{x}_{fail}\| = 0\ .$$

Thus, direct application of continuous state-space model checkers (e.g. [Frehse, 2005; Hartong et al., 2002]) to the oscillator start-up problem will identify regions where trajectories might stay forever. Because we cannot show that the set of failure states is empty, we must settle for showing that it is *negligible* (i.e. occurs with probability zero). This is sufficient in practice, as designers are not worried about a design that fails with probability zero.

For intuition, consider an oscillator where all inverters are identical. We define V_{eq} as the voltage that can be applied to the input of the inverter such that the output settles to the same voltage. When all of the inverters are identical, $\mathbf{x}_{fail}$ is the point at which all node voltages are V_{eq}. Furthermore, any trajectory starting at a point where

$$\forall j \in [0, n-1].; x(0,j) = x(1,j) \tag{12}$$

converges to $\mathbf{x}_{fail}$; thus, such points are in X_{fail}.

Using existing reachability methods, we can find a small region, U_{fail}, that contains the point $\mathbf{x}_{fail}$. Furthermore, we can show that if an oscillator starts any point where each node has a voltage in the interval $[0, V_{dd}]$, then within bounded time, the oscillator state will either be in U_{fail}, or it will be in a region where we can show convergence to the desired periodic orbit.

We will show that the set of failing trajectories is sufficiently small as to ensure that the oscillator fails to start with a probability of zero. To do this, we need a notion of probability – for the present work, we will assume that there is some smooth distribution of initial states. As in the previous section, we write $\mathbb{R}^d$ to denote the phase space. We will avoid a detailed treatment of measure theory (see [Pollard, 2001]) by noting that when we say that $B \subseteq \mathbb{R}^d$ is measurable, we mean that it has a well-defined d-dimensional "volume" (i.e. it is Lesbesgue measurable), and we write $|B|$ to denote this volume (i.e. measure). We write $\mu(B)$ to denote the probability that the initial state of the oscillator is in B. Our assumption that μ is smooth (i.e. absolutely continuous) means that if $|B|$ is zero, then $\mu(B)$ is zero as well. For example, let

$$B = \{(x(0,0), x(0,1), x(1,0), x(1,1)) \mid (x(0,0) = x(1,0)) \wedge (x(0,1) = x(1,1))\}$$

i.e. the plane described by Equation 12. Because this plane has zero volume, $|B| = 0$, and by our smoothness assumption, $\mu(B) = 0$ as well.

Let U be a bounded, measurable subset of $\mathbb{R}^d$. We define

$$\begin{aligned} \mathsf{escape}_f(\mathbf{x}, U) &= \exists t \in \mathbb{R}^+.\ \Phi_f(\mathbf{x}, t) \notin U \\ \mathsf{trapped}_f(U) &= \{\mathbf{x} \in U \mid \neg \mathsf{escape}_f(\mathbf{x}, U)\} \end{aligned}$$

For any $U \subseteq \mathbb{R}^+$, and any $t \in \mathbb{R}$, $|U| = 0 \Leftrightarrow |\Phi_f(U,t)| = 0$. Thus, it suffices to show that $|\mathsf{trapped}_f(U_{fail})| = 0$.

Before stating the main theorem for this section, we provide some intuitive motivation. Let $H \in \mathbb{R}^{d \times d}$ be a symmetric matrix. If the eigenvalues of H are of mixed sign, then the set $\{x \mid x^T H x = 0\}$ is a cone. We can think of the quantity $\mathbf{x}^T H \mathbf{x}$ as the "distance" from $\mathbf{x}$ to the boundary of the cone – this distance is positive if $\mathbf{x}$ is inside the cone, negative if $\mathbf{x}$ is outside the cone, and zero if $\mathbf{x}$ is on the boundary of the cone. Figure 6 shows such a cone along with curves of constant "distance" from the boundary. If $\mathbf{x}$ is inside the cone and

$$\tfrac{d}{dt}(\mathbf{x}^T H \mathbf{x}) = 2\mathbf{x}^T H \tfrac{d}{dt}\mathbf{x} > 0 \tag{13}$$

then the trajectory at $\mathbf{x}$ is moving further to the inside of the cone.

In Theorem 2 below, we consider two trajectories; one starting at point $\mathbf{x}_1$ and the other at $\mathbf{x}_2$ where $\mathbf{x}_2$ starts inside a cone whose apex is at $\mathbf{x}_1$. As both trajectories evolve, we move the apex of the cone along the trajectory emanating from $\mathbf{x}_1$. The trajectory emanating from $\mathbf{x}_2$ stays within this cone and diverges from the trajectory that emanated from $\mathbf{x}_1$. This shows that only one of the trajectories can remain in U. From this, we conclude that $\mu(\mathsf{trapped}_f(U)) = 0$. This argument is formalized below.

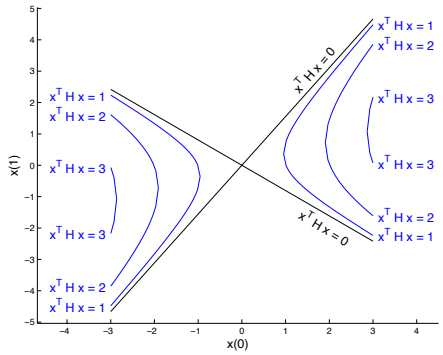


Fig. 6: A cone

Theorem 2 *Let μ be a smooth probability measure over $\mathbb{R}^d$. Let U be a bounded, measurable subset of $\mathbb{R}^d$, and $f : \mathbb{R}^d \to \mathbb{R}^d$ be bounded and C^1 in U. If there is a symmetric matrix $H \in \mathbb{R}^{d\times d}$ that has at least one positive eigenvalue, and $k > 0$ such that for all $\mathbf{x}_1, \mathbf{x}_2 \in U$:*

$$\begin{aligned}&(\mathbf{x}_2 - \mathbf{x}_1)^T H(\mathbf{x}_2 - \mathbf{x}_1) \;>\; 0\\ \Rightarrow\; &(\mathbf{x}_2 - \mathbf{x}_1)^T H(f(\mathbf{x}_2) - f(\mathbf{x}_1)) \;>\; k(\mathbf{x}_2 - \mathbf{x}_1)^T H(\mathbf{x}_2 - \mathbf{x}_1)\,,\end{aligned}$$

then $\mu(\mathsf{trapped}_f(U)) = 0$.

Proof Assume that $\mathsf{trapped}_f(U) \neq \emptyset$ as the other case is trivial. Because H is symmetric, all of its eigenvalues are real. Let $\lambda_{\max}$ be the maximum eigenvalue of H, and let $\mathbf{u}$ be an eigenvector for $\lambda_{\max}$. Let $\mathbf{x}_0$ be any point in $\mathsf{trapped}_f(U)$, and $\alpha \in \mathbb{R}$ such that $\alpha > 0$ and $\mathbf{x}_0 + \alpha\mathbf{u} \in U$. Let $\mathbf{x}_1 = \mathbf{x}_0 + \alpha\mathbf{u}$.

We now show $\mathbf{x}_1 \notin \mathsf{trapped}_f(U)$. Consider two trajectories,

$$\begin{aligned}\eta_0(t) &= \Phi_f(\mathbf{x}_0, t), \text{ the trajectory that starts at } \mathbf{x}_0\\ \eta_1(t) &= \Phi_f(\mathbf{x}_1, t), \text{ the trajectory that starts at } \mathbf{x}_1\end{aligned}$$

Note that both trajectories start in U. We'll show that these two trajectories diverge; accordingly, at most one of them can remain in U. Let

$$w(t) = (\eta_1(t) - \eta_0(t))^T H(\eta_1(t) - \eta_0(t))$$

We claim that for $t \geq 0$, $w(t) \geq \alpha^2\lambda_{\max}e^{kt} > 0$. First note that $w(0) = \alpha^2\lambda_{\max}$ which satisfies the claim (at $t = 0$). Both $w(t)$ and $\alpha^2\lambda_{\max}e^{kt}$ are continuous functions of t. Thus, if the claim were ever to be violated, there would have to be a value of t for which $w(t) = \alpha^2\lambda_{\max}e^{kt}$ and $\frac{d}{dt}w(t) < \frac{d}{dt}\alpha^2\lambda_{\max}e^{kt}$. For the sake of contradiction, let t be such a time. Then

$$\begin{aligned}\tfrac{d}{dt}w(t) &= (\eta_1(t) - \eta_0(t))^T H(f(\eta_1(t)) - f(\eta_0(t)))\\ &> k(\eta_1(t) - \eta_0(t))^T H(\eta_1(t) - \eta_0(t))\\ &= kw(t) = k\alpha^2\lambda_{\max}e^{kt} = \tfrac{d}{dt}\alpha^2\lambda_{\max}e^{kt}\end{aligned}$$

But this shows that $\frac{d}{dt}w(t) > \frac{d}{dt}\alpha^2\lambda_{\max}e^{kt}$, a contradiction. Thus, $w(t) \geq \alpha^2\lambda_{\max}e^{kt}$ as claimed.

Because, $w(t) \geq \alpha^2\lambda_{\max}e^{kt}$, $\|\eta_1(t) - \eta_0(t)\|$ must diverge as $t \to \infty$. By assumption, $\eta_0(t)$ stays in U, and U is bounded. Therefore, $\eta_1(t)$ must exit U.

We have shown that for any point $\mathbf{x}_0 \in \mathsf{trapped}_f(U)$, all points in the cone defined by H whose apex is at $\mathbf{x}_0$ must escape from U. This shows that $\mathsf{trapped}_f(U)$ must have lower dimension than the full space. Thus, $|\mathsf{trapped}_f(U)| = 0$, and therefore $\mu(\mathsf{trapped}_f(U)) = 0$ as claimed.

Note: Theorem 2 was based on the cone argument from [Mitchell and Greenstreet, 1996]. The present theorem generalizes the result from [Mitchell and Greenstreet, 1996] to systems of arbitrary dimensions and whose Jacobian matrices have complex eigenvalues. The conditions for Theorem 2 are slightly stronger than those from [Mitchell and Greenstreet, 1996] (for the systems where the latter applies) – this is mainly for simplicity.

5.3 Proper Oscillation

For the trajectories under consideration after the first two steps, the common-mode components of both differential signal pairs are within V_{err} of V_0^+. This allows the differential equation model from Equation 11 to be rewritten as a *differential inclusion* [Henzinger et al., 1998]:

$$\dot{\mathbf{u}} \in F(\mathbf{u}) \tag{14}$$

where $\mathbf{u}$ is the vector $\frac{\sqrt{2}}{2}[x(0,0) - x(1,0), x(0,1) - x(1,1)]$. By using an inclusion, F accounts for *all* values of the common mode components in $[V_0^+ - V_{err}, V_0^+ + V_{err}]$. Reducing the four-dimensional state space of the original problem to a two-dimensional space makes the exploration of trajectories from all remaining start conditions straightforward.

By showing that all such trajectories lead to an oscillation in the fundamental mode, we solve the first part of the challenge problem from [Jones et al., 2008]: we show that for a particular choice of transistor sizes, the circuit will start oscillation from all initial conditions except for a set of zero measure. Section 7 provides a brief description of how these methods can be extended to establish a range of transistor sizes for which the oscillator will start with probability one.

6 Verification Implementation

This section describes our implementation of the verification techniques described in the previous section. We construct an ODE model for the ring oscillator circuit using standard, modified nodal analysis. We obtain drain-to-source current data by tabulating HSPICE outputs and fitting piece-wise quadratic functions to this tabulated data. The resulting errors are less than 1%; thus, our transistor models closely match those used by practicing circuit designers in industry.

6.1 Differential Operation

This verification phase starts by changing the coordinate system to one based on the differential and common mode representation of signals.

Let $\mathbf{u}$ be the circuit state in "differential" coordinates:

$$\mathbf{u} = \frac{\sqrt{2}}{2} \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} x(0,0) \\ x(0,1) \\ x(1,0) \\ x(1,1) \end{bmatrix} \tag{15}$$

We assume each of nodes $\times(0,0)$, $\times(0,1)$, $\times(1,0)$ and $\times(1,1)$ can independently have initial voltages anywhere in $[0,1.8]$V. Thus, the differential components, u_1 and u_2, are initially in $[-0.9\sqrt{2}, +0.9\sqrt{2}]$, and the common mode components, u_3 and u_4, are initially in $[0,1.8\sqrt{2}]$.

To establish differential operation, we divide the range of each component u_i of $\mathbf{u}$ into m intervals, creating m^4 cubes. We construct a graph, $G = (V,E)$ to represent the reachability relationship between these cubes. Let $v_{i,j,k,\ell}$ be a vertex corresponding to the i^{th} interval for u_1, the j^{th} interval for u_2 and so on. There is an edge from v to w if f allows a flow out of the cube for v directly into the cube for w, and there is a self-loop for v if each component of f is zero somewhere in v. If a vertex of G has no incoming edges, then any trajectory that starts in the corresponding cube will eventually leave that cube, and no trajectories will ever enter the cube. Such a cube can be eliminated from further consideration. Thus, we only need to consider cubes whose vertices are members of cycles. These vertices can be identified in $O(V+E) = O(m^4)$ time. With a direct implementation of this computation, constructing G dominates the entire time for verifying the oscillator.

To obtain a more efficient computation, we first note that the goal is to establish differential operation. It is sufficient to project the vertices of V onto the common-mode components of the differential signals and show that most of this projection can be eliminated from further consideration. Let $G' = (V',E')$ where $v'_{k,\ell}$ corresponds to the k^{th} interval of u_3 and the ℓ^{th} interval of u_4. There is an edge in E' from v'_{k_1,ℓ_1} to v'_{k_2,ℓ_2} iff there exist i and j such that $(v_{i,j,k_1,\ell_1}, v_{i,j,k_2,\ell_2}) \in E$. Clearly, G' over approximates reachability. Thus, if a vertex of G' has no incoming edges, then all of the corresponding vertices in G must have no incoming edges as well. Computing the edges in E' requires examining all of the edges of E, but subsequent operations on the graph G' are much faster than those on G.

To reduce the time required to find edges of E, we start with a small value of m and thus a coarse grid. Many large blocks can be eliminated from G' even with a coarse grid. We then double m (i.e. divide each vertex of G' into four) and recompute reachability using the finer grid for finding edges in E as well. In practice this adaptive griding approach eliminates blocks quickly while achieving enough precision to allow the rest of the verification to proceed without difficulties.

6.2 Escape from the Failure Set

At the end of establishing differential operation, there are a few cubes with self-loops – there is more than one such cube because of the over approximations described above. These cubes contain the point $\mathbf{x}_{fail}$. We now construct a larger cube that contains all of these and make a change of variables so that this cube is centered at the origin. We'll write $\mathbf{x}$ for vectors in the original coordinate system and $\mathbf{u}$ for vectors in the coordinates where the center of a cube with a self-loop is at the origin. Let r be the maximum ℓ_2 distance of any point in this cube from the origin. Typically, we can make r arbitrarily small (but not zero) at a cost of increased compute time for further rounds of sub-dividing the blocks and finding cycles.

As described at the beginning of this section, we use piecewise quadratic models for transistor currents and model node capacitances as constants. Thus, the derivative function, f, is piecewise quadratic. Our repeated subdivision of cubes when establishing differential operation ensures that the cube containing $\mathbf{x}_{fail}$ is modeled by a simple quadratic (i.e. a single

"piece"). We can write this model as:

$$\dot{\mathbf{u}} = A_0 + A_1\mathbf{u} + \sum_{j=1}^{d}(\mathbf{u}^T A_{2,j}\mathbf{u})\mathbf{b}_j \tag{16}$$

where $\mathbf{b}_j$ is a unit vector corresponding to the j^{th} component of $\mathbf{u}$. We will assume without loss of generality that the $A_{2,j}$ matrices are symmetric.

To establish the hypotheses of Theorem 2, we again exploit the differential operation of the oscillator and choose $H = \mathrm{diag}([+1,+1,-1,-1])$. The two $+1$ elements of H anticipate a growing, differential component of the state, and the two -1 elements are for a diminishing common-mode component. To satisfy the condtions of Theorem 2, we need to find a $k > 0$ such that for any $\mathbf{u}_1$ and $\mathbf{u}_2$ in U with $(\mathbf{u}_2-\mathbf{u}_1)^T H(\mathbf{u}_2-\mathbf{u}_1) > 0$,

$$\frac{(\mathbf{u}_2-\mathbf{u}_1)^T H(f(\mathbf{u}_2)-f(\mathbf{u}_1))}{(\mathbf{u}_2-\mathbf{u}_1)^T H(\mathbf{u}_2-\mathbf{u}_1)} > k$$

Substituting for f from Equation 16 yields:

$$\frac{(\mathbf{u}_2-\mathbf{u}_1)^T H(f(\mathbf{u}_2)-f(\mathbf{u}_1))}{(\mathbf{u}_2-\mathbf{u}_1)^T H(\mathbf{u}_2-\mathbf{u}_1)} = \frac{(\mathbf{u}_2-\mathbf{u}_1)^T HA_1(\mathbf{u}_2-\mathbf{u}_1)}{(\mathbf{u}_2-\mathbf{u}_1)^T H(\mathbf{u}_2-\mathbf{u}_1)} + \frac{(\mathbf{u}_2-\mathbf{u}_1)^T H\sum_{j=1}^{d}((\mathbf{u}_2-\mathbf{u}_1)^T A_{2,j}(\mathbf{u}_2+\mathbf{u}_1))\mathbf{b}_j}{(\mathbf{u}_2-\mathbf{u}_1)^T H(\mathbf{u}_2-\mathbf{u}_1)} \tag{17}$$

We now derive a lower bound for the linear part of Equation 17,

$$\frac{(\mathbf{u}_2-\mathbf{u}_1)^T HA_1(\mathbf{u}_2-\mathbf{u}_1)}{(\mathbf{u}_2-\mathbf{u}_1)^T H(\mathbf{u}_2-\mathbf{u}_1)} \tag{18}$$

and an upper bound for for the magnitude of the quadratic part,

$$\left|\frac{(\mathbf{u}_2-\mathbf{u}_1)^T H\sum_{j=1}^{d}((\mathbf{u}_2-\mathbf{u}_1)^T A_{2,j}(\mathbf{u}_2+\mathbf{u}_1))\mathbf{b}_j}{(\mathbf{u}_2-\mathbf{u}_1)^T H(\mathbf{u}_2-\mathbf{u}_1)}\right| \tag{19}$$

when $(\mathbf{u}_2-\mathbf{u}_1)^T H(\mathbf{u}_2-\mathbf{u}_1) > 0$.

Equation 18 is a convex conic program and can be solved by standard techniques [Boyd and Vandenberghe, 2004, chap. 4.4]; let $\mathsf{lin}_{\min}$ be the minimum value for Equation 18.

To bound the magnitude of the quadratic term, let $\sigma_{\max}$ denote the largest singular value of any of the $A_{2,j}$ matrices. Then, for all $j \in 1\ldots d$,

$$(\mathbf{u}_2-\mathbf{u}_1)^T A_{2,j}(\mathbf{u}_2+\mathbf{u}_1) \leq \sigma_{\max}(\mathbf{u}_2-\mathbf{u}_1)^T(\mathbf{u}_2+\mathbf{u}_1)$$

Therefore,

$$\begin{aligned}&\left\|\textstyle\sum_{j=1}^{d}((\mathbf{u}_2-\mathbf{u}_1)^T A_{2,j}(\mathbf{u}_2+\mathbf{u}_1))\mathbf{b}_j\right\| \\ &\leq \sigma_{\max}\sqrt{d}(\mathbf{u}_2-\mathbf{u}_1)^T(\mathbf{u}_2+\mathbf{u}_1)\end{aligned}$$

Noting that the largest singular value of H is 1, and $\|\mathbf{u}_2+\mathbf{u}_1\| \leq 2r$, we get:

$$\begin{aligned}&(\mathbf{u}_2-\mathbf{u}_1)^T H\textstyle\sum_{j=1}^{d}((\mathbf{u}_2-\mathbf{u}_1)^T A_{2,j}(\mathbf{u}_2+\mathbf{u}_1))\mathbf{b}_j \\ &\leq 2r\sigma_{\max}\sqrt{d}\|\mathbf{u}_2-\mathbf{u}_1\|^2\end{aligned} \tag{20}$$

By our choice of H,

$$(\mathbf{u}_2-\mathbf{u}_1)^T H(\mathbf{u}_2-\mathbf{u}_1) \leq \|\mathbf{u}_2-\mathbf{u}_1\|^2 \tag{21}$$

Thus,

$$\left|\frac{(\mathbf{u}_2-\mathbf{u}_1)^T H\sum_{j=1}^{d}((\mathbf{u}_2-\mathbf{u}_1)^T A_{2,j}(\mathbf{u}_2+\mathbf{u}_1))\mathbf{b}_j}{(\mathbf{u}_2-\mathbf{u}_1)^T H(\mathbf{u}_2-\mathbf{u}_1)}\right| \leq 2r\sqrt{d}\sigma_{\max} \tag{22}$$

Let $k = \text{lin}_{\min} - 2r\sigma_{\max}\sqrt{d}$. Combining the results from Equations 17 through 22, yields

$$(\mathbf{u}_2 - \mathbf{u}_1)^T H(f(\mathbf{u}_2) - f(\mathbf{u}_1)) \geq k(\mathbf{u}_2 - \mathbf{u}_1)^T H(\mathbf{u}_2 - \mathbf{u}_1)$$

If $k > 0$, then we can satisfy the conditions of Theorem 2. In practice, the conditions of Theorem 2 can be satisfied by obtaining a sufficiently small value for r as described at the beginning of this section.

6.3 Proper Oscillation

As described in Section 5.3, we reduce the state space from four dimensions to two by replacing the differential equation model for the circuit with a differential inclusion. The space to be considered forms a ring: the outer boundary is determined by the assumption that all signals have voltages between ground and V_{dd}, and the inner boundary is established by eliminating trajectories in a neighborhood near x_{fail}. Figure 9b shows the remaining region. We use a collection of "spokes" as shown in Figure 10, and show that all trajectories in these wedges converge to a unique, periodic attractor. The computation has three parts:

1. Starting from each "spoke", show that all trajectories starting at that spoke eventually cross the next spoke.
2. Show that all trajectories starting from the inner or outer boundary eventually cross the next spoke.
3. Starting from one spoke, compute the reachable set until it converges to a limit set.

7 Results

We generated transistor models using HSPICE to determine drain-to-source currents for 0.18μ long and 1μ wide nMOS and pMOS devices with the gate and drain voltages swept from 0 to 1.8V in 0.01V steps. For the nMOS transistors, we assume that the source and body are at 0V, and for the pMOS devices, we assume that they are at 1.8V. We assume that all transistors have a length of 0.18μ, and obtain current for other widths by linear scaling from the 1μ data. For all inverters, we use pMOS devices that are twice as wide as the nMOS devices. All forward inverters have transistors of the same size, and likewise for the cross-coupled inverters. In the following, s denotes ratio of the cross-coupled inverter size to the forward inverter size. This section first present the DC equilibrium analysis result: the circuit is free from lock-up for $0.638 < s < 2.243$. Then the circuit is verified for $0.673 \leq s \leq 2.0$.

The verification routines were implemented using Matlab with Coho used for the final reachability computation. All times were obtained running on a dual Xeon E5520 (quad core) 2.27GHz machine with 32GB of memory. The computations described here are all performed using a single core.

7.1 DC Equilibria

We implemented the analysis described in Section 3 as Matlab scripts and compared our results with HSPICE simulations. We varied the transistor width to determine the inverter sizings required to guarantee oscillation. Figure 7 displays the results. Let

$$\textit{instability}(s) = \min_{q \in Q(s)} \max_{e \in \Lambda(q)} \textit{real}(e) \tag{23}$$

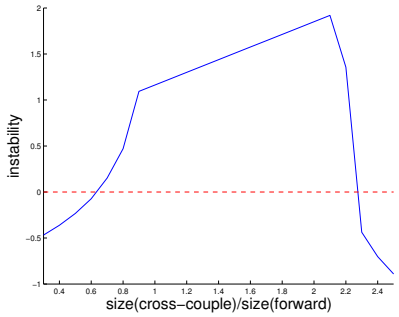


Fig. 7: Stability Test for a 4-Stage Oscillator

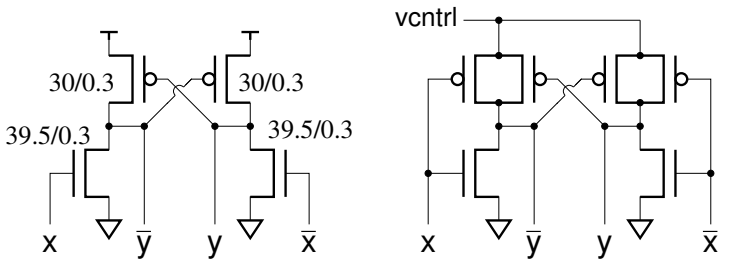
where $Q(s)$ is the set of all DC equilibria for an oscillator with an inverter size ratio of s, and $\Lambda(q)$ is the set of eigenvalues of the Jacobian matrix for $\dot{V}$ at q. Thus, if $instability(s) > 0$, all DC equilibria are unstable and the oscillator will not lock up. Conversely if $instability(s) < 0$, the circuit can lock-up for some initial conditions. We determined that lock-up is excluded for $0.638 < s < 2.243$. The runtime for a four-stage oscillator is only 16 seconds, more efficient compared with 10-13 minutes for a two-stage oscillator in [Steinhorst and Hedrich, 2012].

We then simulated the oscillator circuit using both HSPICE and Matlab's numerical integrator function (ode113) to explore the behavior near the critical values. As expected, we found lock-up behaviors for values of $s < 0.638$ and $s > 2.243$. To our surprise, we found that the four-stage oscillator can support stable oscillations for values of s very close to zero.

For $\varepsilon < s < 0.638$ (with a very small, positive ε), the four-stage ring has three stable behaviors: it can lock-up at either of the two stable DC equilibria described in Equation 1, or it can oscillate. The actual behavior depends on the initial conditions. Because the analysis algorithm from Section 3 provides the values of the equilibrium points, it was a simple matter to specify initial conditions for HSPICE that elicited each of the possible steady-state behaviours.

When we simulated the oscillator in HSPICE without specifying initial conditions, it found a DC operating point where all node voltages are roughly the same (a metastable point in the dynamics). This state is clearly metastable, which a designer would typically test by perturbing the circuit slightly. For example, injecting a 0.05V pulse onto one node of the circuit drove it away from this metastable point and into its stable oscillatory behavior. This is a potentially treacherous situation for an analog circuit designer. It shows that there are designs for which many initial conditions (including the default in HSPICE) lead to the desired behavior, but some conditions can lead to lock-up. Thus, this is a failure mode that could easily go undetected using standard simulation methods and only be discovered in the test lab, or after the chip has been shipped in products. We believe that examples such as this demonstrate the value of our formal approach to analyzing analog circuits.

We repeated these experiments with a two-stage ring-oscillator. The upper critical values for s are unchanged. For small values of s, the oscillator can either lock-up or exhibit stable oscillations for $0.31 < s < 0.638$. Thus, a design that has only been validated by simulation has the same potential failures as the four-stage version. For $s < 0.31$, the circuit fails to oscillate for all initial conditions.



(a) From [Nissinen et al., 2003] (b) From [Tatschl-Unterberger et al., 2005]

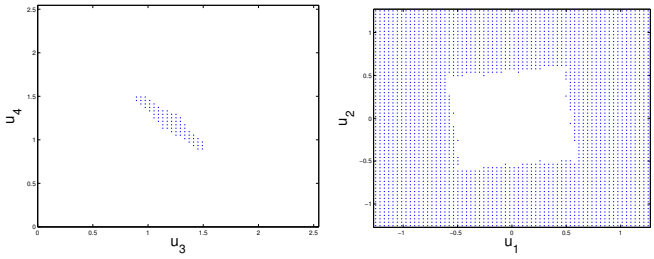
Fig. 8: Other Ring-Oscillator Circuits

To examine the robustness of the oscillator to process parameter variations, we repeated the analysis with inverters where the p-channel transistors were 2.5 times the width of their n-channel counterparts. This is roughly the ratio needed to achieve equal rise and fall times in this process. With this sizing, the critical values for s become 0.64 and 2.20. When the p-channel transistors were 1.5 times the width of their n-channel counterpart, we obtained critical values for s of 0.578 and 2.496. The range of s for which lock-up free operation is guaranteed grows slightly with smaller p-channel devices, but the sensitivity to parameter changes appears to be fairly small.

Variants of the oscillator from Figure 5 are common in the design literature (e.g. [Djahanshahi and Salama, 2000; Heydari and Pedram, 2001; Nissinen et al., 2003; Park and Kim, 1999; Tatschl-Unterberger et al., 2005; Xiao et al., 2001]). Figure 8 shows two typical examples; in each circuit, signals x and $\overline{\mathsf{x}}$ are the differential inputs, and y and $\overline{\mathsf{y}}$ are the differential outputs. Most of the oscillators are designed for high-frequency operation and have two to eight stages. An exception is the design from [Xiao et al., 2001] which is used for an on-chip voltage regulator and has 128 stages.

In spite of the diversity of these oscillator circuits, we note that all of them preserve the monotonicity requirements upon which our analysis is based. In particular, Lemma 3 can be adapted to apply to each of these circuits. We illustrate this by briefly examining the start-up conditions for the circuit in Figure 8b as many common oscillator designs are minor variations of this circuit. The difference between the oscillator stage from Figure 8b and that from Figure 5 is that the stage from Figure 8b omits the pull-down N-channel transistors in the cross-coupled inverters. Applying our analysis to this design, we found that the circuit has stable DC equilibria for any reasonable choice of transistor sizes!

The lock-up behavior in the design from [Tatschl-Unterberger et al., 2005] occurs at the DC-equilibria where the forward inverters dominate the cross-coupling ones. At these equilibria, even numbered stages will have both outputs within a few hundredths of a volt of V_{dd}, while odd numbered stages have both outputs a few tenths of a volt about ground (or vice-versa). At such a state, the output conductances of the forward inverters are very large and their transconductances are very small. Thus, each stage has a small-signal gain less than one, and disturbances die out as they propagate around the ring. If the cross-coupled P-channel devices are extremely large, then the forward inverters never drive their outputs much below $V_{dd}/2$; they remain saturated; and this provides enough transconductance to ensure oscillation. However, such a design is highly impractical as the large pull-ups drastically lower the frequency of oscillation and increase the power consumption.



(a) Common-mode convergence to $V_{dd}\sqrt{2}/2$

(b) Eliminating the unstable equilibrium

Fig. 9: Results of the first two verification steps

We demonstrated these behaviors with HSPICE simulations. We considered forward inverters with no pull-up (Figure 8a), and ones with the pull-up the same width or twice the width of the pull-down. We varied the size of the cross-coupled pull-ups from half the width of the forward inverter pull-down to four times the width of the forward inverter pull-down. In all cases, either a stable DC equilibrium was reached or stable oscillations occurred depending on the initial conditions. These circuits may work most of the time in practice because the initial conditions that lead to a stable DC equilibrium may occur infrequently. However, we are not aware of CAD tools or techniques that could verify the reliability of such designs.

7.2 Complete Verification of Start-Up for a Two-Stage Oscillator

The first phase of the verification establishes differential operation. Initially, the computation partitions the space for each of the u_i variables into 8 regions, creating a total of $8^4 = 4096$ cubes to explore. After eliminating cubes that have no incoming or self-circulating flows, the remaining cubes are subdivided and rechecked until there are 64 intervals for each variable. Figure 9a shows the remaining cubes projected onto the common-mode variables, u_3 and u_4 at the end of this phase.

Table 1 summarize the process of estabilishing differential operation. At the end of the first round with 8 intervals per variable, there are 700 cubes under consideration (18% of the total space). With each subdivision, the number of cubes remaining increases by a factor of roughly 4.6, and thus the volume of the space under consideration drops by about a factor of roughly 0.29. With 64 intervals per region, 74500 cubes remain (0.45% of the total space). The decrease in the volume is steady, suggesting that further reductions would be possible with more iterations. However, the time per iteration increases with the number of cubes under consideration, and the time for this phase dominates the total verification time. Thus, for verifying this circuit, there is no incentive to further refine the region bounding the common-mode signal.

The second phase of the verification eliminates the unstable equilibrium. The equilibrium is near the point where all node voltages are 0.867V. We chose U to be the hyper-rectangle with sides of length 0.1V whose center is at this point. The region U contains all cubes that correspond to graph-vertices with self-loops from phase 1. There is more than one

ΔV	#cubes (start)	#cubes (end)	time (seconds)
$V_{dd}/8$	4,096	700	5
$V_{dd}/16$	11,200	3,422	20
$V_{dd}/32$	54,752	16,432	101
$V_{dd}/64$	262,912	74,500	474

Table 1: Computation summary for showing differential operation

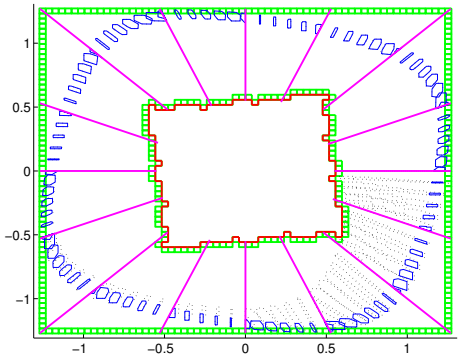


Fig. 10: Computing the invariant set

such cube due to the use of interval arithmetic in computing the adjacency graph to ensure soundness. Using the least-squares best-fit quadratic model for points in U yields:

$$\begin{aligned} lin_{\min} &> 5 \times 10^{10}\text{sec}^{-1}, \\ \sigma_{\max} &< 2 \times 10^{9}\text{sec}^{-1}\text{V}^{-1}, \text{ and} \\ r &= 0.1\text{V} \end{aligned}$$

from which we get that the conditions of Theorem 2 are satisfied for any k with $0 < k < 4.92 \times 10^{10}\text{sec}^{-1}$. Thus, we can safely remove the cubes in U.

We can now repeat the procedure from phase 1 to remove all cubes that transitively have no incoming flows. This phase eliminates roughly half of the remaining cubes, leaving 38384 cubes for analysis by the final phase. This is the fastest of the three phases – most of the time for this phase is spent removing cubes that have no incoming flows after the metastable cubes have been eliminated.

The final phase starts with the 38384 cubes from the second phase. As described in Section 6.3, we divide these cubes into 16 wedges divided by "spokes" in the $u_1 \times u_2$ projection. As described in Section 6.3, it is sufficient to show trajectories starting on the boundary of the wedge lead to points inside the next wedge in the clockwise direction. With 16 wedges, we perform 48 reachability computation runs. At this point, the oscillator is verified.

We also ran a longer reachability computation starting from a spoke and completing two complete cycles of the oscillation. The second cycle starts from a smaller region that the first and establishes tighter bounds on the limit cycle. The blue polygons in Figure 10 indicate this limit cycle. The remaining width of the limit cycle is mainly due to approximating the four-dimensional differential equation with a differential inclusion.

The computation is very efficient. The run-time of the first phase is about 720 seconds, and the run-time of reachability computation is less than 470 seconds. Reducing state space

Table 2: Verification Times (seconds)

s	phase 1	phase 2	phase 3	Total	verified?
[0.5,0.7]	2053	749	—	2802	NO
[0.7,0.9]	1122	464	—	1586	NO
[0.85,0.9]	705	237	—	942	NO
[0.875,0.9]	652	209	659	1520	YES
[0.9,1.1]	724	257	468	1449	YES
[1.1,1.3]	533	171	382	1086	YES
[1.3,1.5]	429	132	402	963	YES
[1.5,1.7]	378	112	512	1002	YES
[1.7,1.9]	335	99	624	1058	YES
[1.9,2.0]	308	91	688	1087	YES
[2.0,2.1]	308	91	1150	1549	NO
[1.9,2.1]	308	91	1850	2249	NO
[2.1,2.3]	293	80	3879	4252	NO
[2.3,2.5]	268	74	3861	4203	NO

helps to improve performance significantly. It takes several hours to complete the reachability computation in full-dimensional space from a single cube. In contrast, the run-time in reduced-space is less than 10 minutes as shown above. Reducing the space also introduces over-approximations to the reachable regions. However, this did not lead to false-negative results, as the circuit converges to the oscillation orbit rapidly.

7.3 Verification for a range of sizes

Phases 1 and 3 of our verification method use conservative over-approximations to guarantee soundness of the results. These approximations make it straightforward to model s as being in an interval rather than having a precise value. We have verified escape from the failure set for values of s from 0.673 to 2.0 by testing values of s in steps of 0.01 for $0.6 \leq s \leq 2$ and in steps of 0.001 for $0.67 \leq s \leq 0.7$. The lower-bound for s is slightly higher than the one (0.638) reported by the DC analysis. For $s > 2$, the third phase of the verification fails to show that trajectories leave the "corners" of the $u_1 \times u_2$ space. These correspond to lock-up of the cross-coupled inverters. As described above, the DC analysis method shows that these lock-up states become stable for $s > 2.243$. The gap between the reachability computation and the DC analysis is presumably due to conservative over-approximations used in the reachability method.

Table 2 shows the run times for each phase of the verification. In general, the time for the first phase decreases with larger values of s because the stronger cross-coupled inverters eliminate the common-mode component of the signals faster. For the same reason, the number of cubes at the end of phase 1 decreases with larger values of s resulting less time for phase 2. If the second phase shows a failure, we don't run phase 3. Generally, the run-time for phase three increases with larger s because the oscillator period increases, and it takes the reachability computation more steps to show that trajectories exit regions where the cross-coupled inverters are close to locking up. For the smallest values of s, the oscillator converges to its limit set more slowly, and we ran the reachability computation longer to establish a small limit set.

8 Conclusions

This paper has presented two approaches to verifiying that the oscillator circuit presented in [Jones et al., 2008] properly starts from almost all initial conditions. In particular:

- Section 3 presented an algorithm to find all DC equilibria of an oscillator circuit and classify their stability properties. This method is fast, can handle oscillators with a large number of stages, and can be applied to a wide range of ring-oscillator topologies.
- Section 4 showed that no "physically plausible" oscillator starts from all initial conditions (Theorem 1, Section 4). Our proof is based on the observation that smooth dynamical systems preserve the topological properties of sets.
- We presented a generalization of the "cone-argument" from [Mitchell and Greenstreet, 1996] to show that the failures occur with probability zero and thus the oscillator starts with probability one (Theorem 2, Section 5).
- Our approach shows how reachability analysis can be combined effectively with dynamical systems analysis.
- We showed how differential-operation, a common feature of analog designs, can be exploited for model reduction.

First, metastable behaviors are unavoidable for most mode-switching circuits. While metastability is most often associated with synchronizer circuits [Chaney and Molnar, 1973; Kinniment et al., 2007], it arises anytime the state of a continuous system can evolve to two or more distinct states. For example, when a phase-locked loop (PLL) locks, the VCO phase may advance to match the phase of the reference, or the VCO may drop back depending on the initial conditions. Thus, there are conditions where any physically realizable PLL takes an arbitrarily long time to lock. On the other hand, there are are published verifications of bounded lock time for phase-locked loops (e.g. [Althoff et al., 2011]). The discrepancy is resolved by noting that [Althoff et al., 2011] uses an abstract model for the phase-comparator that makes a discontinuous step as the phase difference passes through 180°. For many designs, this is a reasonable abstraction; yet, we note that a PLL can fail to lock if there is a dead-spot in the response of the phase-comparator at the wrap-around point. We see our work as complementary to that of [Althoff et al., 2011] – they provide powerful abstractions that enable the verification of larger designs, and we provide methods of ensuring that those abstractions are sound.

Second, our verification combined analytical methods from dynamical systems theory with reachability methods that are more typical of the formal methods community. Neither alone is sufficient to verify the oscillator. Reachability techniques are inadequate because they cannot show escape from a failure set of zero measure. Such "failures" are not of concern to practical designers as they are unobservable in the physical system. On the other hand, the dynamical systems methods that allow us to establish probability-one results are arguments about local dynamics. The reachability computations are needed to go from these local results to proving global properties.

The notion of probability that we used, a smooth distribution over initial states, was simplistic. A more physical model would use stochastic integration techniques to determine the evolution of this distribution under the circuit dynamics as perturbed by noise processes such as thermal noise. While this might be more satisfying, it would mainly serve to make the mathematics more complicated, and quantitative results would be hard to obtain due to the highly non-linear dynamics of the circuits. However, the basic topological observations on which we base our results would be preserved. Thus, we believe that our probability one results would continue to hold in a more detailed, stochastic model.

Proving that something happens "eventually" can be unsatisfying, as such proofs often don't give an indication of how long one needs to wait. Our proof for Theorem 2 shows that the divergence is at least as fast as an exponential with time-constant k. For the oscillator considered, $k \approx 1/(20\text{ps})$. Thus, we can make a quantitative conclusion that in a few nanoseconds, the probability that the oscillator has not started is extremely small. This should satisfy practicing designers.

Of course, there are many areas of future work. Most immediately, we claimed escape from the failure set for a wide-range of inverter sizes by verifying the property for a large number of closely spaced choices of the sizes. We would like to use interval-arithmetic methods to show that these intervals are completely covered. To do so, we are making a few extensions to the intlab package [Rump, 1999]. Likewise, we plan to show that the method can be applied to a design in a more state-of-the-art process (e.g. using PTM models [Cao, 2008]).

We would like to verify larger circuits. As described in Section refsec:relwk, a ring oscillator with six or more stages can have stable higher harmonic modes if small inter-stage couplings are included in the model. We would like to verify (and refute) such designs. We note that many real-world analog designs consist of tens or hundreds of transistors, and these circuits typically consist of many replicas of a few simple blocks. This motivates developing parameterized approaches to verification of analog and mixed signal designs. The multi-stage Möbius oscillator circuit is a good example for detailed analysis of how reachability computation complexity scales with circuit size.

Acknowledgments

Throughout this work we have benefited from interactions with many excellent colleagues. We would like to express our particular gratitude to Michael Friedlander, Kevin Jones, Victor Konrad, Ian Mitchell, John Poulton, and Mohamed Zaki for helpful conversations and inspiration in the course of this research. Mitchell

References

Althoff, Mattiah, Akshay Rajhans, et al.. 2011. Formal verification of phase-locked loops using reachability analysis and continuization. In *Proceedings of the 2011 international conference on computer aided design*, 659–666.

Boyd, Stephen, and Lieven Vandenberghe. 2004. *Convex optimization*. Cambridge University Press.

Cao, Yu. 2008. PTM: predictive technology model. http://ptm.asu.edu.

Chaney, T. J., and C. E. Molnar. 1973. Anomalous behavior of synchronizer and arbiter circuits. *IEEE Transactions on Computers* C-22 (4): 421–422.

Djahanshahi, H., and C. A. T. Salama. 2000. A two-stage differential CCO implementation in submicron CMOS. In *Proceedings of the* 43^{rd} *IEEE midwest symposium on circuits and systems*, 294–297.

Fränzle, Martin. 2007. HySAT: An efficient proof engine for bounded model checking of hybrid systems. *Formal Methods in System Design* 30 (3): 179–198. doi:10.1007/s10703-006-0031-0.

Frehse, Goran. 2005. PHAVer: Algorithmic verification of hybrid systems past HyTech. In *Proceedings of the fifth international workshop on hybrid systems: Computation and control*, 258–273. Springer. LNCS 3414.

Frehse, Goran, Bruce H. Krogh, and Rob A. Rutenbar. 2006. Verifying analog oscillator circuits using forward/backward abstraction refinement. In *Proceedings of design automation and test europe*, 257–262.

Greenstreet, Mark R., and Suwen Yang. 2008. Verifying start-up conditions for a ring oscillator. In *Proceedings of the* 18^{th} *great lakes symposium on VLSI (GLSVLSI'08)*, 201–206.

Gupta, Smriti, Bruce H. Krogh, and Rob A. Rutenbar. 2004. Towards formal verification of analog designs. In *Proceedings of 2004 IEEE/ACM international conference on computer aided design*, 210–217.

Hartong, Walter, Lars Heidrich, and Erich Barke. 2002. Model checking algorithms for analog verification. In *Proceedings of the 39th ACM/IEEE design automation conference*, 542–547.

Henzinger, Thomas A., Pei-Hsin Ho, and Howard Wong-Toi. 1998. Algorithmic analysis of nonlinear hybrid systems. *IEEE Transactions on Automatic Control* 43 (4): 540–554.

Heydari, P., and M. Pedram. 2001. Jitter-induced power/ground noise in CMOS PLLs: a design perspective. In *Proceedings international conference on computer design (ICCD)*, 209–213.

Hirsch, Morris W., and Stephen Smale. 1974. *Differential equations, dynamical systems, and linear algebra*. San Diego, CA: Academic Press.

Jones, Kevin D., Jaeha Kim, and Victor Konrad. 2008. Some "real world" problems in the analog and mixed-signal domains. In *Proc. workshop on designing correct circuits*.

Kim, Jaeha, Metha Jeeradit, Byongchan Lim, and Mark A. Horowitz. 2009. Leveraging designer's intent: a path toward simpler analog CAD tools. In *Proceedings of the custom integrated circuits conference (CICC'2009)*, 613–620. doi:10.1109/CICC.2009.5280741.

Kinniment, David J., Charles Dike, et al.. 2007. Measuring deep metastability and its effect on synchronizer performance. *IEEE Transactions on VLSI Systems* 15: 1028–1039.

Little, Scott, and Chris Myers. 2008. Abstract Modeling and Simulation Aided Verification of Analog/Mixed-Signal Circuits, Princeton, NJ. presented at the 2008 Workshop on Formal Verification for Analog Circuits (FAC'08).

Little, Scott, Nicholas Seegmiller, David Walter, Chris Myers, and Tomohiro Yoneda. 2006. Verification of analog/mixed-signal circuits using labeled hybrid petri nets. In *Proceedings of the international conference on computer aided design*, 275–282.

Mitchell, Ian, and Mark Greenstreet. 1996. Proving Newtonian arbiters correct, almost surely. In *Proceedings of the third workshop on designing correct circuits*. Båstad, Sweden.

Moura, Leonardo, and Nikolaj Bjørner. 2008. Z3: An efficient SMT solver. In *Tools and algorithms for the construction and analysis of systems*, eds. C. R. Ramakrishnan and Jakob Rehof. Vol. 4963 of *Lecture notes in computer science*, 337–340. Springer. ISBN 978-3-540-78799-0.

Nissinen, I., A. Mantyniemi, and J. Kostamovaara. 2003. A CMOS time-to-digital converter based on a ring oscillator for a laser radar. In *Proceedings of the* 29^{th} *european solid-state circuits conference (ESSCIRC'03)*, 469–472.

Park, Chan-Hong, and Beomsup Kim. 1999. A low-noise, 900MHz VCO in 0.6μ CMOS. *IEEE Journal of Solid-State Circuits* 34 (5): 586–591.

Pollard, David. 2001. *A user's guide to measure theoretic probability*. Cambridge University Press.

Rump, S. M. . 1999. INTLAB - INTerval LABoratory. In *Developments in Reliable Com-*

puting, ed. Tibor Csendes, 77–104. Dordrecht: Kluwer Academic Publishers. `http://www.ti3.tu-harburg.de/rump/`.

Rump, S. M. . 2006. Eigenvalues, pseudospectrum and structured perturbations. *Linear Algebra and its Applications* 413 (1-2): 567–593. doi:10.1016/j.laa.2005.06.009.

Steinhorst, S., and L. Hedrich. 2012. Trajectory-directed discrete state space modeling for formal verification of nonlinear analog circuits. In *2012 ieee/acm international conference on computer-aided design (iccad)*, 202–209.

Steinhorst, Sebastian, Markus Peter, and Lars Hedrich. 2009. State space exploration of analog circuits by visualized multi-parallel particle simulation. In *International conference on signal processing systems (ICSPS'09)*, 858–862.

Tatschl-Unterberger, Eva, Sasan Cyrusian, and Michael Ruegg. 2005. A 2.5GHz phase-switching PLL using a supply controlled 2-delay-stage 10GHz ring oscillator for improved jitter/mismatch. In *IEEE international symposium on circuits and systems (ISCAS)*, 5453–5456.

Tiwari, Saurabh K., Anubhav Gupta, et al. 2008. fSpice: A Boolean Satisfiability Based Approach for Formally Verifying Analog Circuits, Princeton, NJ. presented at the 2008 Workshop on Formal Verification for Analog Circuits (FAC'08).

Tiwari, S. K., A. Gupta, et al.. 2010. First steps towards SAT-based formal analog verification. In *Proceedings of the 2010 international conference on computer aided design*.

Xiao, Jinwen, Angel V. Peterchev, and Seth R. Sanders. 2001. Architecture and IC implementation of a digital VRM controller. In *IEEE* 32^{nd} *annual power electronics specialists conference (PESC'01)*, Vol. 1, 38–47.

Yan, Chao, and Mark R. Greenstreet. 2008. Faster projection based methods for circuit-level verification. In *Proceedings of the 2008 Asia and South Pacific design automation conference (ASPDAC'08)*, 410–415.

Zaki, Mohamed H., Ian Mitchell, and Mark R. Greenstreet. 2009. Towards a Formal Analysis of DC Equilibria of Analog Designs, Grenoble, France. Presented at the 2009 Workshop on Formal Verification for Analog Circuits (FAC'09).