

SRAM Dynamic Stability Verification by Reachability Analysis with Consideration of Threshold Voltage Variation

Yang Song¹, Hao Yu², Sai Manoj Pudukotai Dinakarrao², and Guoyong Shi¹

¹School of Microelectronics, Shanghai Jiao Tong University, Shanghai, 200240, China

²School of Electrical and Electronic Engineering, Nanyang Technological University, 639798, Singapore
haoyu@ntu.edu.sg *

ABSTRACT

Dynamic stability margin of SRAM is largely suppressed at nano-scale due to not only dynamic noise but also process variation. A novel dynamic stability verification is developed in this paper based on analog reachability analysis for checking SRAM failure. In the presence of mismatch such as threshold voltage variation of all transistors, zonotope-based reachability analysis is deployed to efficiently verify SRAM failure at transistor level. The threshold voltage variation is considered by the modified input range of SRAM. As such, the suppressed stability margin and further failure region can be verified by performing a time-evolved reachability analysis with formed zonotope to distinguish safe and failure regions. One can perform efficient verification of the SRAM dynamic stability without repeated yet time-consuming Monte-Carlo simulations considering variations from all transistors. As demonstrated by numerical experiment results, the developed reachability analysis can accurately verify the SRAM dynamic stability under threshold voltage variations from all transistors. Speedup of more than 400× in runtime can be achieved over the Monte Carlo approach of 500 samples with the similar accuracy.

Categories and Subject Descriptors

B.8.1 Hardware [Performance and Reliability]: Reliability, Testing, and Fault-Tolerance

Keywords

Verification, Reachability Analysis, SRAM Reliability

1. INTRODUCTION

Static noise margin (SNM)[1, 2] is traditionally deployed for SRAM failure characterization due to simple interpretation and measurement. As it may overestimate read failures and underestimate write failures, dynamic SRAM stability margin [3] is increasingly adopted by deploying critical

*This work is sponsored by MOE Tier-1 fund RG 26/10 from Singapore.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

ISPD'13, March 24–27, 2013, Stateline, Nevada, USA.

Copyright 2013 ACM 978-1-4503-1954-6/13/03 ...\$15.00.

word-line pulse width to produces better estimation of failure rates. The verification of SRAM stability margin becomes even harder when technology scales down into nano-scale, which has significantly suppressed the SRAM stability margin in presence of process variations. The operation of SRAM cell shows not only nonlinear but also undetermined behavior. For example, threshold voltage variation in any transistor of one SRAM cell can bring about malfunction. A thorough verification considering threshold variations in all transistors is necessary to provide designers a close scrutiny of potential hazards. However, such verification can be computationally expensive [4].

Based on DC characteristics of inverters, stability analysis has been performed in [5] by modeling failure with normal distribution even when failure occurs in tail of normal distribution. In [6], accurate estimation is achieved without the assumption of normal distribution of failure probability but is based on the most probable failure point searching. Moreover, stochastic orthogonal polynomials are used to derive failure/yield rate without the prohibitive Monte Carlo simulation in [7].

The analytical SRAM reliability verification is also developed. Euler-Newton curve tracing [8] is utilized to find the boundary between the success and failure regions without brute-force exploration in the parameter space. The work in [9] formulates dynamic noise margin with the use of stability boundary, namely the *separatrix* which separates two stability regions in the parameter space. Note that both approaches can reach to prominent efficiency when compared with brute-force Monte Carlo approach. But the searching of boundary for failure region is limited for dimensions of two parameters, and the computational cost may be high to determine the failure of 6T-SRAM with consideration for variations from all transistors.

Reachability analysis has been widely deployed in reliability verification of dynamic circuits and systems. By exploring potential trajectories of operating points in state space [10, 11], it can conveniently provide accurately predicted boundary of multiple trajectories under an uncertainty by one time computation. As such, there is no need to simulate different trajectories to distinguish safe and unsafe cases one by one. The reachability analysis has been deployed for a number of hard analog circuit verifications [12, 13, 14]. In this paper, we introduce a reachability based verification for SRAM dynamic stability, which can take into account variations from all transistors at the same time. The proposed reachability analysis is implemented into a SPICE-like simulator with consideration of nonlinear device model. It can

consider not only threshold voltage but also width variations from multiple transistors. The verification accuracy of the proposed approach is the same while computational cost is much smaller when compared with the Monte Carlo method. Experiments show that the proposed method can achieve speedups up to 481 \times over Monte Carlo approach with 500 samples under the similar accuracy.

The rest of this paper is organized as follows. Section 2 reviews the dynamic stability mechanisms for SRAM failures with consideration of process variation. Section 3 describes the reachability analysis for the verification of SRAM dynamic stability such as the linkage between the reachability analysis and the SRAM verification flow. The proposed verification methodology is validated by experiments in Section 4 for different SRAM malfunctions including write, read and hold failures. Conclusions are drawn in Section 5.

2. SRAM FAILURE ANALYSIS

A 6T-SRAM cell contains two cross-coupled inverters and two access transistors, which functions properly within dynamic noise margin (DNM) [9]. However, due to process variation, the mismatch among transistors may lead to functional failures. For the simplification of illustration, only variation of threshold voltage is considered in this paper, but the proposed approach is general to deal with other kinds of variations such as transistor width.

In this section, physical mechanisms of SRAM failures caused by threshold voltage variation are reviewed, including write, read and hold failures. In addition, the nonlinear SRAM dynamics is also presented as the basis for reachability based verification.

2.1 Failure Mechanisms

2.1.1 Write Failure

Write failure is defined as the inability to write data properly into the SRAM cell. During write operation, both access transistors should be strong enough to pull down or pull up the voltage level at internal nodes. As shown in Fig.1, write operation can be described on the variable plane as the process of pulling the operating point from initial state (bottom-right corner) to the target state (top-left corner). The crossing line named *separatrix* divides the variable plane into two convergent regions. Given enough time, operating point in any region will converge to the nearest stable equilibrium state either at top-left or bottom-right corner. Write operation is aimed at pulling operating point into targeted convergent region such that operating point can converge to the closest equilibrium state after operation finishes, which is shown by point B in Fig.1.

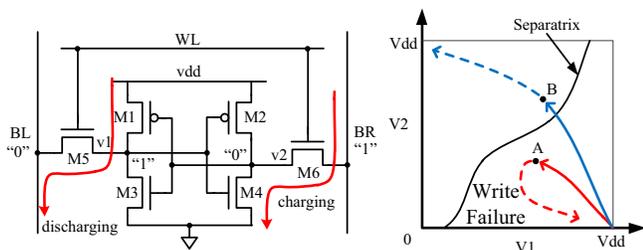


Figure 1: Illustration of SRAM write failure.

However, an increase in threshold voltage due to variation can reduce the transistor driving strength and vice versa for a decrease in threshold. The increase of V_{th} in M6 along with the decrease of V_{th} in M4 can result in difficulty to pull down v_2 . On the variable plane, it becomes more difficult for operating point to move towards the target state. If operating point cannot cross the *separatrix* before access transistors are turned off, it goes back to the initial state, which means a write failure.

2.1.2 Read Failure

Read failure refers to the loss of the previously stored data. Before read operation is performed, both BR and BL are pre-charged to v_{dd} . Suppose previous internal states in SRAM are $v_1 = v_{dd}$ and $v_2 = 0$, electric charge on BR is discharged through M6 and M4 while that on BL remains the same. As such, a small voltage difference between BR and BL is generated which will be detected and amplified. In this way, data stored in the SRAM can be read. Note that access transistors need careful sizing such that their pull-up strength is not strong enough to pull the stored "0" to "1" during read operation.

On the variable plane, operating point is inevitably perturbed and pulled towards the *separatrix*. If read operation does not last too long, access transistors can be shut down before operating point crosses the *separatrix*. As such, the operating point returns to the initial state in the end, as point A in Fig.2, which means a read failure.

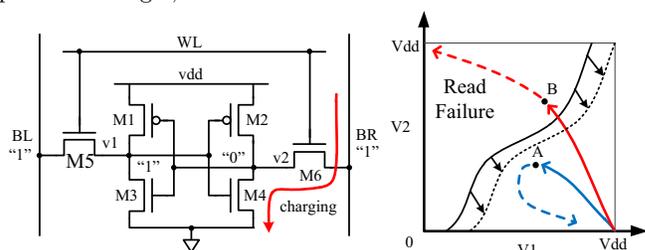


Figure 2: Illustration of SRAM read failure.

Even though all the sizing are carefully taken, threshold variations may still result in read failure. For example, variation caused by mismatch between M4 and M6 may result in unbalanced pulling strength, and v_2 can be pulled up more quickly. As a result, operating point crosses the *separatrix* before read operation ends, as point B in Fig.2.

2.1.3 Hold Failure

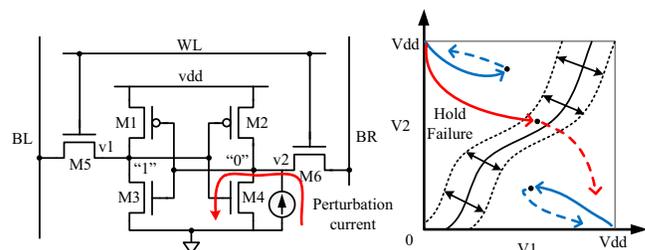


Figure 3: Illustration of SRAM hold failure.

Hold failure happens when the SRAM fails to retain the stored data. It can be caused by external noise or single event upset (SEU). The external perturbation can be modeled as noise current injected into SRAM. Similar to read operation, operating point is expected to converge back to

initial state after settling down from disturbance. Otherwise, it will cross to the other convergent region.

While access transistors have no impact on the retention of SRAM data, M1-4 together can determine the likelihood of hold failure by finding the position of the *separatrix* and thus threshold variation may cause failure by perturbing the *separatrix* as shown in Fig.3. As such, one needs to verify if the SRAM is still tolerable to the injected noise in the presence of threshold voltage variation.

2.2 SRAM Nonlinear Dynamics

The nonlinear dynamics of SRAMs can be described by a differential equation (1)

$$\dot{x} = f(z(t)) \quad (1)$$

in which $z^T(t) = [x^T, u^T]$ is a vector consisting of the state variable vector $x(t)$ and the input vector $u(t)$. $f(x, u)$ describes nonlinearity of SRAMs. As for Monte Carlo based method, one needs to repeatedly launch Newton-Raphson iterations to solve (1) for sample by sample.

Moreover, the nonlinear dynamic equation can be linearized at one solved operating point as $\frac{\partial f}{\partial z}$, and $f(x, u)$ at any neighboring point can be expressed by 1st-order Taylor expansion centering at the linearized operating point (2), called as *nominal point*,

$$\begin{aligned} \dot{x} = & f(z^*) + \frac{\partial f}{\partial z} \Big|_{z=z^*} (z - z^*) + \\ & \frac{1}{2} (z - z^*)^T \frac{\partial^2 f}{\partial z^2} \Big|_{z=\xi} (z - z^*), \quad (2) \\ \xi \in & \{z^* + \alpha(z - z^*) | 0 \leq \alpha \leq 1\} \end{aligned}$$

where z^* is a nominal point and z is a neighbor point around the nominal point. For notation, the 2nd-order remainder in (2) is represented by L , which is also called as *linearization error* in this paper.

As such, in the neighborhood of the nominal point, the SRAM nonlinear dynamics can be depicted by (5)

$$\dot{x}^* + \Delta \dot{x} = f(x^*, u^*) + A\Delta x + B\Delta u + L, \quad (3)$$

in which

$$\begin{aligned} A = & \frac{\partial f}{\partial x} \Big|_{x=x^*}, B = \frac{\partial f}{\partial u} \Big|_{u=u^*} \quad (4) \\ \Delta x = & x - x^*, \Delta u = u - u^*. \end{aligned}$$

After cancelling \dot{x}^* with $f(x^*, u^*)$, a linear differential equation for Δx can be obtained as

$$\dot{\Delta x} = A\Delta x + B\Delta u + L. \quad (5)$$

This equation lays the foundation for reachability analysis in the next section.

The threshold voltage variation can be implemented as additional noise current source added to the drain current of one transistor in SPICE as shown in Fig.4. As such, it becomes the 1st-order Taylor approximation of drain current by (6)

$$\begin{aligned} I_d + \Delta I_d = & \frac{1}{2} k \frac{W}{L} [V_{gs} - (V_{th} + \Delta V_{th})]^2 \quad (6) \\ \Delta I_d \approx & -k \frac{W}{L} (V_{gs} - V_{th}) \Delta V_{th}. \end{aligned}$$

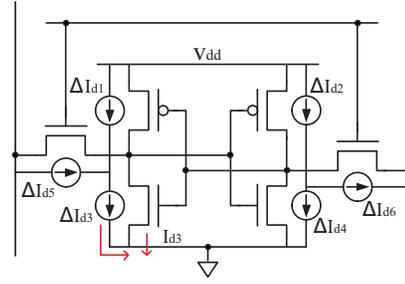


Figure 4: SRAM with threshold voltage variations modeled by additional current sources for all transistors.

The threshold voltage variation of each transistor is included in the input vector u as an independent current source. The other process variations can be conveniently considered in the similar way. Note that noise current in Fig.4 (from the internal transistor) is different from the noise current in Fig.3 (from the external environment).

3. SRAM VERIFICATION BY REACHABILITY ANALYSIS

In this section, we introduce reachability analysis for efficient SRAM dynamic stability verification considering threshold voltage variations. Reachability analysis [15] can efficiently determine a reachable region that one dynamic system can evolve with uncertain initial states or inputs. Applications of reachability analysis include formal verification of continuous, hybrid or discrete systems such as hard analog circuit verifications [12, 13, 14].

In the case of SRAM verification, variation of threshold voltage results in a number of SRAM trajectories. As such, if one models threshold voltage variations from all transistors as noise current sources added to the uncertain input, one can perform one-time reachability analysis to determine a set of reachable trajectories of SRAM. As such, one can easily determine if SRAM trajectories end up in the unsafe region with a failure. In the rest of this section, we discuss the

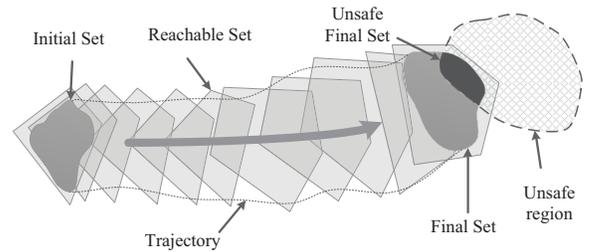


Figure 5: Reachable set and reachability analysis with a set of trajectories under uncertain input range.

reachability analysis for nonlinear circuits, and the according flow for SRAM verification.

3.1 Reachable Set

One important concept for reachability analysis is *reachable set*. A reachable set is the collection of all possible operating points or states in the variable space that a dynamic system may visit. Several approaches have been proposed to approximate reachable set by an enclosing hypercube. One

simple and symmetrical type of hypercube, *zonotope* [10] is defined as

$$Z = \{x \in \mathbb{R}^{n \times 1} : x = c + \sum_{i=1}^q [-1, 1]g^{(i)}\} \quad (7)$$

where $c \in \mathbb{R}^{n \times 1}$ is the zonotope center and $g^{(i)} \in \mathbb{R}^{n \times 1}$ is called as a zonotope generator.

For SRAM verification, let the zonotope center equal the nominal point in order to minimize linearization error [15]. Thus variation of state vector Δx in (5) is expressed as the combination of generators

$$\Delta Z = Z - c = \{x \in \mathbb{R}^{n \times 1} : x = \sum_{i=1}^q [-1, 1]g^{(i)}\}. \quad (8)$$

Note that variation of input vector Δu can be expressed in the same way. Each generator of Δu represents an independent noise current source in (6), i.e., the threshold voltage variation of one transistor.

Algorithm 1: Reachability Analysis for SRAM Dynamic Stability Verification

Input: Initial set Z_0 , system equation $\dot{x} = f(x, u)$, input sets $U_{1, \dots, N}$, simulation time interval h , maximum number of time steps N

Output: Final state P_N

```

1:  $k = 0$ 
2:  $P_N = \emptyset$ 
3: queue.push( $Z_0$ )
4: while !queue.isEmpty() do
5:    $Z_k = \text{queue.pop}()$ 
6:    $(c_k, G_k) = \text{pop}()$ 
7:   compute  $c_{k+1}$  and Jacobian matrices  $A, B$ 
8:   approximate linearization error  $L_{k+1}$ 
9:   if  $\mathbf{IH}(hL_{k+1}) \subseteq [-\varepsilon, \varepsilon]$  then
10:     $G_h = (I - hA)^{-1}G_k$ 
11:     $G_i = (I - hA)^{-1}hBU_{k+1}$ 
12:     $G_e = (I - hA)^{-1}hL_{k+1}$ 
13:     $G_{k+1} = G_h \oplus G_i \oplus G_e$ 
14:     $Z_{k+1} = (c_{k+1}, G_{k+1})$ 
15:    if  $k + 1 == N$  then
16:       $P_N = P_N \cup \text{projection}(Z_N)$ 
17:      continue
18:    end if
19:    queue.push( $Z_{k+1}$ )
20:     $k = k + 1$ 
21:  else
22:     $[Z'_k, Z''_k] = \text{split}(Z_k)$ 
23:    queue.push( $Z'_k$ )
24:    queue.push( $Z''_k$ )
25:  end if
26: end while

```

3.2 Reachability Analysis

The complete flow for reachability analysis for SRAM verification is shown in Algorithm 1. Here, Z_0 is the initial reachable set in zonotope form, and P_N is the projection of final reachable sets onto the variable plane for observation.

In each iteration cycle (Line 5-25), the calculation of a new reachable set goes through Line 5-14 with the final set

collected in Line 15-20. Otherwise, the current reachable set is split in Line 22-24 and iteration is performed again for two new sets. Detailed discussion of the verification flow follows in the remainder of this section.

3.2.1 Initial Trajectory Queue

According to Section 2, when the operating point gets close to the *separatrix*, it may move away from it later and return to the initial state; or it may cross the *separatrix* and go ahead for the opposite state. As such, more than one trajectories are needed to account for all potential situations, which are stored in a queue. The reachability analysis is then performed for all trajectories in the queue (Line 5-25).

3.2.2 Linear Multi-Step Integration

Within each cycle, the first step is to pick up a reachable set Z_k from the trajectory queue (Line 5-6). Newton-Raphson method is deployed to calculate the new zonotope center c_{k+1} based on c_k (Line 7). At the same time, the nonlinear dynamic function in (1) is linearized around c_{k+1} .

After c_{k+1} is obtained, the next step is the calculation of the unknown zonotope generators. Based on (8), Δx in (5) can be substituted with a series of generators. As such we further solve (5) by the Implicit Euler method with discretized time-step h (9).

$$\frac{G_{k+1} - G_k}{h} = AG_{k+1} \oplus BU_{k+1} \oplus L_{k+1} \quad (9)$$

with

$$G_k = [g_k^{(1)}, \dots, g_k^{(p)}], \quad U_k = [u_k^{(1)}, \dots, u_k^{(m)}] \quad (10)$$

where G_k and U_k are sets of generators for state variables and input current sources, respectively. The operator \oplus performs set addition for generator sets. Recall that A and B are the Jacobian matrices at the nominal point c_{k+1} (4).

3.2.3 New Reachable Set Formulation

The superposition principle allows to separate the solution of (9) into two parts: the homogeneous solution with respect to the initial state when there is no input (11a); and the inhomogeneous solution accounting for the system input when the initial state is the origin (11b)(11c). Note that the linearization error is treated as input. As a result, given an initial set for current time step, three sets of solutions are computed and added together by the so-called Minkowski sum (11d).

$$G_h = (I - hA)^{-1}G_k \quad (11a)$$

$$G_i = (I - hA)^{-1}hBU_{k+1} \quad (11b)$$

$$G_e = (I - hA)^{-1}hL_{k+1} \quad (11c)$$

$$G_{k+1} = G_h \oplus G_i \oplus G_e. \quad (11d)$$

This procedure is shown in Line 10-13, where a new reachable set Z_{k+1} is obtained by combining c_{k+1} with G_{k+1} .

3.2.4 Reachable Set Refinement

Approximation of linearization error L_k in Line 8 is a critical step in each iteration cycle. Linearization error (12) accounts for nonlinearity of SRAM dynamics. Here, nominal point x^* is the zonotope center for current iteration c_k and x varies within the zonotope Z_k . L_k cannot be exactly calculated but approximated for $x \in Z_k$. Detailed approximation of L_k can be found in [15].

$$L_k = \frac{1}{2}(x - x^*)^T \left. \frac{\partial^2 f}{\partial x^2} \right|_{x=x^*} (x - x^*), \quad (12)$$

$$\xi \in \{x^* + \alpha(x - x^*) | 0 \leq \alpha \leq 1\}.$$

As for SRAM, nonlinearity of the system is rather prominent in the transition area around the *separatrix* where linearization error expands rapidly. Over-expanded new reachable sets in (11d) may be too rough to be meaningful. Thus reachable set is split into smaller ones when linearization error exceeds user-defined limit so that linearization error of each new set is in an appropriate size.

A judgement condition for set splitting is shown in (13),

$$\mathbf{IH}(hL_k) \subseteq [-\varepsilon, \varepsilon] \quad (13)$$

in which $\mathbf{IH}()$ is the interval hull operation which converts a zonotope to a multi-dimensional interval and ε is user-defined limit vector. After the current reachable set is divided into two subsets, along with a new trajectory being created, the reachability analysis will be repeated at current time point for the two new subsets (Line 22-24).

3.2.5 Reachability Check

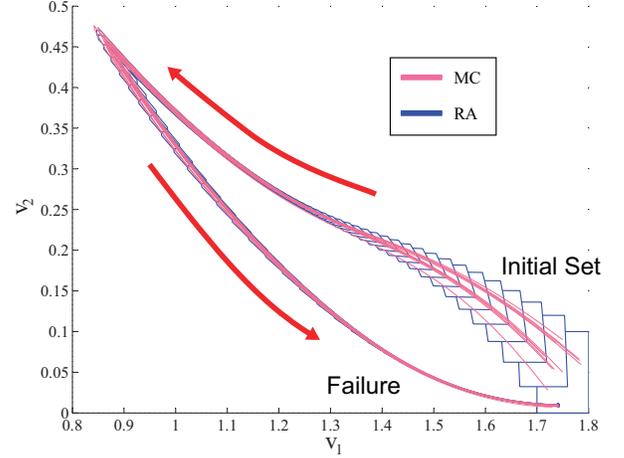
Finally, after reachable set at the maximum simulation step is generated, it will be projected onto the variable plane for observation. Projections of all possible final reachable sets are collected in P_N (Line 16). Dynamic stability is guaranteed if P_N ends up in the safe region.

As a summary, the developed reachability analysis by zonotope in Algorithm 1 can be efficiently deployed to check the SRAM dynamic stability in read, write and hold operations. As reachability analysis can consider mismatches from all transistors, modeled as input current sources, it is scalable to deal with large dimensional variation problem. Moreover, the impact of all possible mismatches is characterized in zonotope with the derived failure region by one time simulation, which avoids expensive multiple runs of Monte Carlo simulations.

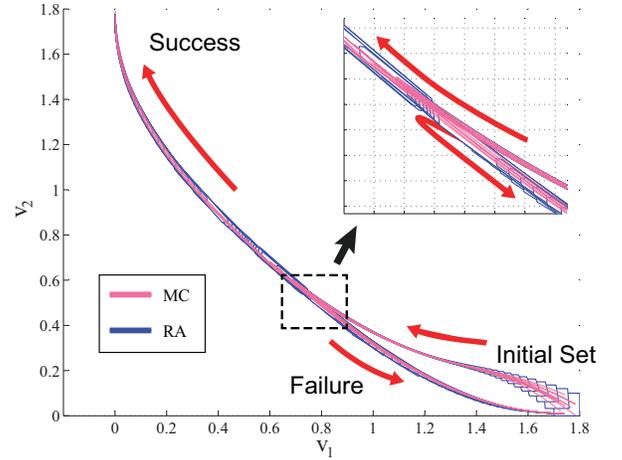
4. EXPERIMENTAL RESULTS

The verification of SRAM dynamic stability by reachability analysis is implemented inside one SPICE-like simulator by MATLAB. Application of zonotopes based reachability analysis is performed by MATLAB toolbox named Multi-Parametric Toolbox (MPT)[16]. Experiment data is collected on a desktop with Intel Core i5 3.2GHz processor and 8GB memory.

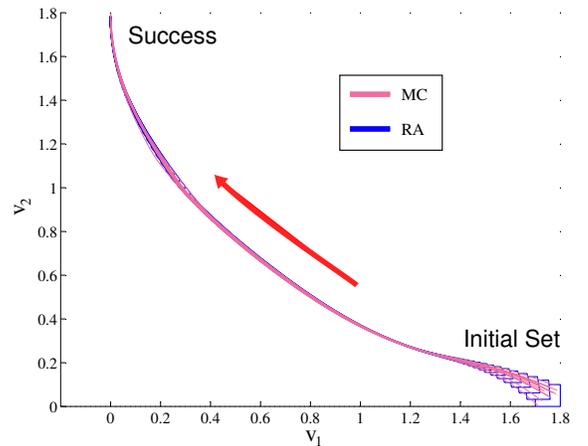
The setup of SRAM circuit is as follows. Supply voltage of SRAM v_{dd} is set to 1.8V. BSIM3v3 is used for the transistor model. Three different threshold variation ranges are tested in the experiment, including 1%, 5% and 10%. Larger variation range can be considered when high-order noise model is available. Threshold voltage variation in each transistor is introduced as a noise current source in (6), whose center value is 0 and variation is $|k \frac{W}{L} (V_{gs} - V_{th}) \delta V_{th}|$ where δ is the variation range. We start reachability analysis with an initial state set of $v_1 \in [1.7, 1.8]$ and $v_2 \in [0, 0.1]$. In this section, verification of different SRAM operations are performed, followed by a comparison with the Monte Carlo based verification of multiple samplings.



(a) Write operation fails with 50ns writing pulse.



(b) Write operation fails with 60ns writing pulse.



(c) Write operation succeeds with 70ns writing pulse.

Figure 6: Verification of write operation with threshold variation range of 5%.

4.1 Verification of Write Operation

First of all, the write operation is verified with consideration of threshold voltage variation. For comparison, Monte Carlo simulation is performed to verify the accuracy of reachability analysis. The duration of write signal is varied to exam SRAM behaviors under different conditions.

Verification results are shown in Fig.6 with threshold-voltage-variation range set to 5%. The trajectories simulated by Monte Carlo method are plotted in light purple and trajectories of reachability analysis are drawn in dark blue. Three different durations of write signal are tested, including 50ns, 60ns and 70ns.

In Fig.6(a) write signal lasts for 50ns. At the beginning, trajectories move towards the other corner of variable plane as data is being written into SRAM. Later, the turning point of trajectories is generated when the write signal flips to 0. Afterwards, trajectories return to initial states. As such, the data fails to be written into the SRAM, which means that write failure happens.

When the write pulse increases to 60ns in Fig.6(b), trajectories of reachable sets split around the center of the variable plane. This happens when the write signal shuts down. Some of the new trajectories move back to initial states, which means some states still fail the write operation.

Finally, when the duration increases to 70ns, all possible states finish write operation without failure. As shown in Fig.6, trajectories of Monte Carlo method remain within the reachable sets by reachability analysis with the similar accuracy. It indicates that reachability analysis can succeed in approximating the trajectory of SRAM.

4.2 Verification of Read Operation

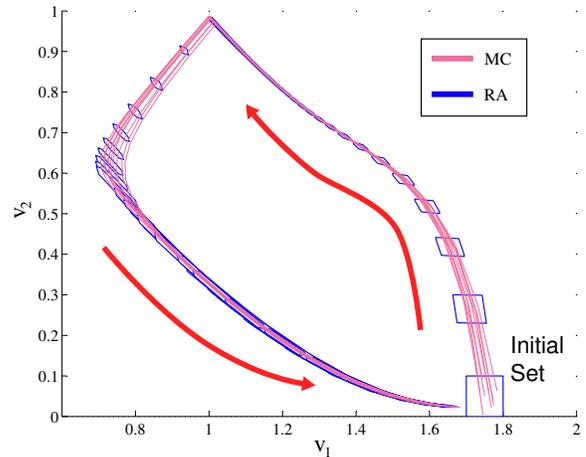
Moreover, the read operation can be also verified by reachability analysis. Instead of observing results by variations of signal duration, we compare the verification results with different threshold voltage variations within the same duration of input signal. Duration of read signal is set to 50ns. Two different threshold variation ranges are verified, including 5% and 1%.

As shown in Fig.7, when threshold voltage variation range is 1%, all reachable sets recover back to the initial state after read operation finishes (Fig.7(a)). SRAM can function properly in this situation. But after the variation range rises to 5%, some reachable sets do not return to the initial set but deviate from others after read operation ends (Fig.7(b)). In the end, these reachable sets rest in the opposite state, which means that read failure happens. Note that the turning points in Fig.7(a) and Fig.7(b) appear when the read signal shuts down. The reason is that the read signal controls the force that pulls the trajectory towards the *separatrix*. Without the pulling strength, trajectory will move towards an equilibrium state.

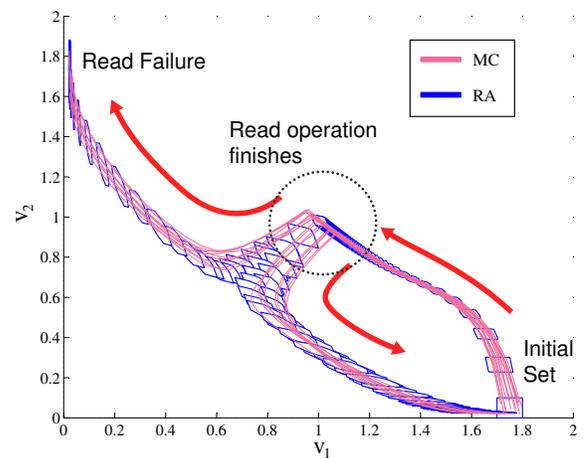
The Monte Carlo trajectories are plotted in light purple and the enclosing trajectories drawn by reachability analysis are in dark blue. They match each other perfectly in other parts of the variable plane where nonlinearity is weak. There is very small error after trajectories pass through the turning point due to the linearization error in (12), which can be resolved when refinement is applied on the reachable set.

4.3 Verification of Hold Operation

Next, the hold operation is verified when an injected environmental noise to node v_2 . Threshold voltage variation



(a) Read operation succeeds with 1% threshold variation range.



(b) Read operation fails with 5% threshold variation range.

Figure 7: Verification of read operation with different threshold variations.

range is set to 5% for each transistor. Duration of the injected noise is to last for 12.5ns. Two injected noises with different magnitudes are verified.

In Fig.8, the trajectories drawn by dash lines represent the case with 300uA injected noise; and other trajectories with solid lines have an injected noise of 315uA. Again, light purple trajectories are for Monte Carlo method and dark blue boxes are reachable sets by reachability analysis. When the injected noise is set to 300uA, trajectories (dash line) return to initial states after perturbation. As the noise magnitude increases to 315uA, trajectories deviate from the recovering route and converge to the opposite state. Hold failure then happens as a result. Results of Monte Carlo and reachability analysis can match with each other in the similar accuracy.

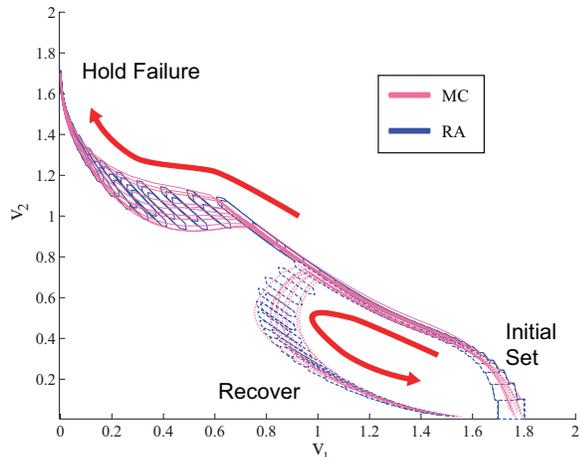


Figure 8: Verification of hold operation with different injected noise.

4.4 Comparison with Monte Carlo Method

At last, a detailed comparison between reachability analysis and Monte Carlo based verifications is made upon the write operation. For the concern of huge time consumption by Monte Carlo method, we use 500 samples which usually cost about 2 hours for a single round of verification according to our experiment. Different durations of write signal are considered as well as different threshold voltage variations in all transistors. Detailed experimental results are listed in Table 1 in which ‘Pulse’ refers to the duration of input signal and ‘Acceleration’ is the ratio of time consumption of Monte Carlo to that of reachability verification. As shown, compared with Monte Carlo, reachability analysis can achieve speedup up to more than 400 times. When write signal duration is set to 50ns (Fig.6(a)) or 70ns (Fig.6(c)), only one trajectory is generated by reachability analysis. Linearization is performed around one nominal trajectory which takes up most of the simulation time. As signal duration is set to 60ns, reachable sets are split into different parts (Fig.6(b)) and up to five trajectories are generated. The acceleration rate is roughly equivalent to the ratio of the number of Monte Carlo samples to the number of trajectories by reachability analysis. For all experiment cases listed in the table, the reachability analysis can achieve the similar accuracy as Monte Carlo method to report the failure region.

Table 1: Time consumption of SRAM verification.

Pulse (ns)	Threshold Variation	Reachability Analysis(s)	Monte Carlo(s)	Acceleration
50	1%	12.71	5635.57	443.38×
	5%	13.13	5817.71	443.01×
	10%	12.63	6078.68	481.24×
60	1%	52.70	6224.09	118.09×
	5%	52.58	6535.35	124.29×
	10%	52.68	6387.28	121.25×
70	1%	13.72	5931.76	432.32×
	5%	14.43	6245.45	432.73×
	10%	13.21	6348.54	480.45×

5. CONCLUSIONS

In this paper, we are the first to develop the reachability analysis for the verification of SRAM dynamic stability

in the presence of process variations from all transistors. By modeling variations as uncertain input currents added to the input range, the zonotope based reachability analysis is deployed to provide the system performance boundary for the estimation of SRAM dynamic stability region. As demonstrated by experimental results, when taking into account on threshold voltage variations from all transistors of 6T-SRAM memory cell, reachability analysis based verification can provide accurate trajectories with failure boundary as Monte Carlo method, but with speedup up to 481× achieved. With further implementation, stochastic reachability analysis can be used on the verification of statistical metrics including failure probability and yield rate.

6. REFERENCES

- [1] E. Seevinck, F. J. List, and J. Lohstroh. Static-noise margin analysis of MOS SRAM cells. *IEEE Journal of Solid-State Circuits (JSSC)*, Jan. 1987.
- [2] E. Grossar, M. Stucchi, K. Maex, and W. Dehaene. Read stability and write-ability analysis of SRAM cells for nanometer technologies. *IEEE Journal of Solid-State Circuits (JSSC)*, Nov. 2006.
- [3] S. O. Toh, Z. Guo, and B. Nikolić. Dynamic SRAM stability characterization in 45nm CMOS. In *IEEE Symposium on VLSI Circuits (VLSIC)*, Jun. 2010.
- [4] H. Yu and S. X.-D. Tan. Recent advance in computational prototyping for analysis of high-performance analog/rf ics. In *IEEE International Conf. on ASIC (ASICON)*, Oct. 2009.
- [5] K. Agarwal and S. Nassif. Statistical analysis of SRAM cell stability. In *ACM/IEEE Design Automation Conference (DAC)*, 2006.
- [6] D.E. Khalil, M. Khellah, N.S. Kim, Y. Ismail, T. Karnik, and V.K. De. Accurate estimation of SRAM dynamic stability. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Dec. 2008.
- [7] F. Gong, X. Liu, H. Yu, S. X.-D. Tan, J. Ren, and L. He. A fast non-monte-carlo yield analysis and optimization by stochastic orthogonal polynomials. *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, Jan. 2012.
- [8] S. Srivastava and J. Roychowdhury. Rapid estimation of the probability of SRAM failure due to mos threshold variations. In *IEEE Custom Integrated Circuits Conference (CICC)*, Sep. 2007.
- [9] W. Dong, P. Li, and G. M. Huang. SRAM dynamic stability: theory, variability and analysis. In *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, 2008.
- [10] A. Girard. Reachability of uncertain linear systems using zonotopes. In *Int. conf. on Hybrid Systems: computation and control (HSCC)*. Springer, 2005.
- [11] M. Althoff. Reachability analysis and its application to the safety assessment of autonomous cars. In *PhD Dissertation, TUM*, 2010.
- [12] M. Althoff and et.al. Formal verification of phase-locked loops using reachability analysis and continuation. In *IEEE Int. Conf. of Computer-aided-design (ICCAD)*, 2011.
- [13] G. Frehse, B.H. Krogh, and R.A. Rutenbar. Verifying analog oscillator circuits using forward/backward abstraction refinement. In *IEEE Design, Automation and Test in Europe (DATE)*, Mar. 2006.
- [14] Y. Song, H. Fu, H. Yu, and G. Shi. Stable backward reachability correction for pll verification with consideration of environmental noise induced jitter. In *IEEE/ACM Asia and South Pacific Design Automation Conference (ASPDAC)*, Jan. 2013.
- [15] M. Althoff, O. Stursberg, and M. Buss. Reachability analysis of nonlinear systems with uncertain parameters using conservative linearization. In *IEEE Conf. on Decision and Ctrl.*, 2008.
- [16] M. Kvasnica, P. Grieder, and M. Baotić. Multi-Parametric Toolbox (MPT). MPT 2.6.3 is available at <http://control.ee.ethz.ch/~mpt/>.