A Fast and Provably Bounded Failure Analysis of Memory Circuits in High Dimensions

Wei Wu¹, Fang Gong¹, Gengsheng Chen², Lei He¹,²

¹ Electrical Engineering Department, UCLA
http://eda.ee.ucla.edu

² School of Microelectronics, Fudan University
Outline

- Preliminary of High Sigma Analysis and Existing Approaches
- The Proposed Approach
- Experiment Results
- Conclusions and Future Work
Why Stochastic Modeling and Analysis?

- Ongoing scaling trends
  - Shrinking devices ➞ larger process variations
  - More duplicated circuits: memory, IO, multi-core ➞ higher robustness over variations

- Stochastic modeling and analysis helps to debug circuits in the pre-silicon phase, and enhances yield rate

90nm  45nm  14nm

Shrinking Feature Sizes
High Sigma Analysis

- High sigma for analog and custom circuits (IO, memory control, PLL)

*source: normal distribution on Wikipedia*
Existing Methods and Limitations

- MC simulation:
  - *time-consuming*

- Traditional Importance Sampling methods
  - *inaccurate* and *unreliable* at high dimension

- Statistical Blockade\(^1\):
  - Existing classifier is *not robust*

- Other approaches: probability collectives\(^2\), quick yield\(^3\) only work on low dimension problem.

---


Basic Idea in Importance Sampling

- Importance Sampling
  - Shift sampling distribution towards the failure region.

- Indicator Function

- Probability of rare failure events
  - variable $x$ and its PDF $h(x)$

$$prob(failure) = \int I(x) \cdot h(x) dx = \int I(x) \cdot \frac{h(x)}{g(x)} \cdot g(x) dx$$

- Likelihood ratio or weights for each sample of $x$ is $h(x)/g(x)$, which is unbounded when $g(x)$ becomes very small under high dimension.
Outline

- Preliminary of High Sigma Analysis and Existing Approaches
- The Proposed Approach
- Experiment Results
- Conclusions and Future Work
Overview of the Proposed Algorithm

- Three stage algorithm:
  - Build a region $R$ (eg. 0.99 quantile), $\{Y|Y\geq t\}$, which is not so rare, and evaluate the probability of this region, $P(Y\geq t)$ with MC
Overview of the Proposed Algorithm

- Three stage algorithm:
  - Build a region $R$ (e.g. 0.99 quantile), $\{Y|Y \geq t\}$, which is not so rare, and evaluate the probability of this region, $P(Y \geq t)$ with MC.
Overview of the Proposed Algorithm

- Three stage algorithm:
  - Build a region $R$ (e.g. 0.99 quantile), $\{Y | Y \geq t\}$, which is not so rare, and evaluate the probability of this region, $P(Y \geq t)$ with MC
  - Generate a new distribution $Y_t$ covers $R$ and estimate the conditional failure probability: $P(Y \geq t_c | Y \geq t)$. 

![Diagram showing a distribution with a quantile t and region R]
Overview of the Proposed Algorithm

- Three stage algorithm:
  - Build a region $\mathbf{R}$ (e.g. 0.99 quantile), $\{Y|Y \geq t\}$, which is not so rare, and evaluate the probability of this region, $P(Y \geq t)$ with MC.
  
  - Generate a new distribution $Y_t$ covers $\mathbf{R}$ and estimate the conditional failure probability: $P(Y \geq t_c|Y \geq t)$. 

$t$ (eg: 0.99-quantile)
Overview of the Proposed Algorithm

- Three stage algorithm:
  - Build a region \( R \) (eg. 0.99 quantile), \( \{Y|Y\geq t\} \), which is not so rare, and evaluate the probability of this region, \( P(Y\geq t) \) with MC
  - Generate a new distribution \( Y_t \) covers \( R \) and estimate the conditional failure probability: \( P(Y \geq t_c|Y \geq t) \).
  - Failure Probability:
    \[ P(Y \geq t_c) = P(Y \geq t) \times P(Y \geq t_c|Y \geq t) \]
Stage 2: Generate a new distribution $Y_t$ covers $R$ and estimate the conditional failure probability: $P(Y \geq t_c | Y \geq t)$.

- **mean-shift**: move towards the region with more potential failure. E.g. we move the mean to the centroid of $R$ in this work.

- **sigma-change**: reshape to dominate the “rare-event” region.

$$\sigma = \max(d, \sigma(Y_t))$$

to make sure the entire failure region can be properly covered.
Stage 3: Evaluation of Conditional Probability

- **Failure Probability:** \( P(Y \geq t_c) = P(Y \geq t) \cdot P(Y \geq t_c | Y \geq t) \)

- **Conditional Probability** is calculated as:

\[
P(Y \geq t_c | Y \geq t) = \frac{P(Y \geq t_c, Y \geq t)}{P(Y \geq t)} = \frac{P(Y \geq t_c)}{P(Y \geq t)} = \frac{\sum_{i=1}^{N} w(x_i) \cdot I_{\{Y \geq t \}}(x_i)}{\sum_{i=1}^{N} w(x_i) \cdot I_{\{Y \geq t \}}(x_i)}
\]

\[
w(x_i) = \frac{h(x_i)}{g(x_i)}; \quad I_{\{Y \geq t \}}(x_i) = \begin{cases} 0 & \text{if } Y(x_i) \notin \{Y \mid Y \geq t\} \\ 1 & \text{if } Y(x_i) \in \{Y \mid Y \geq t\} \end{cases}
\]

- **Boundedness analysis:**
  - Upper bound of estimations from classic importance sampling approaches \( \infty \)!
  - The estimations of the proposed algorithm are always bounded.
Outline

- Preliminary of High Sigma Analysis and Existing Approaches
- The Proposed Approach
- Experiment Results
- Conclusions and Future Work
High-Sigma Analysis on a SRAM circuit

- Functional Diagram on an SRAM circuit
High-Sigma Analysis on a SRAM circuit

- Functional Diagram on an SRAM circuit

  ![Functional Diagram](image)

- Test Cases

  - Bit-cell (54 variables, effectively 36 variables as $M_{p5}$ and $M_{p6}$ are OFF)
    - Consider timing failure as the “rare event” of interest.
High-Sigma Analysis on a SRAM circuit

- **Functional Diagram on an SRAM circuit**

![Functional Diagram of SRAM Circuit](image)

- **Test Cases**
  - Bit-cell (54 variables, effectively 36 variables as *Mp5* and *Mp6* are OFF)
    - Consider timing failure as the “rare event” of interest.
  - Sense Amplifier (117 variables, with 90 independent variables)
    - Evaluate the gain of the amplifier.
  - Build on 45nm technology node
High-Sigma Analysis on a SRAM circuit

- Functional Diagram on an SRAM circuit

- Test Cases
  - Bit-cell (54 variables, effectively 36 variables as \( Mp5 \) and \( Mp6 \) are OFF)
    - Consider timing failure as the “rare event” of interest.
  - Sense Amplifier (117 variables, with 90 independent variables)
    - Evaluate the gain of the amplifier.
  - Build on 45nm technology node
## SRAM bit-cell circuit

- Experiment results with 90% confidence level on target accuracy:

<table>
<thead>
<tr>
<th></th>
<th>MC</th>
<th>SS</th>
<th>SB</th>
<th>HDIS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Failure rate</td>
<td>2.413E-05 (0%)</td>
<td>28415E-05 (+17.7%)</td>
<td>2.7248E-05 (+12.9%)</td>
<td>2.4949E-05 (+3.39%)</td>
</tr>
<tr>
<td># of simulations (x1000)</td>
<td>4600 (1150X)</td>
<td>20 (5X)</td>
<td>816 (204X)</td>
<td>4 (1X)</td>
</tr>
</tbody>
</table>

**MC**: Monte Carlo, **SS**: Spherical Sampling, **SB**: Statistical Blockade, **HDIS**: the proposed high-dimensional importance sampling

Runtime of 1000 simulations: ~ 5 mins.
Experiment results with 90% confidence level on target accuracy:

<table>
<thead>
<tr>
<th></th>
<th>MC</th>
<th>SS</th>
<th>SB</th>
<th>HDIS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Failure rate</td>
<td>2.413E-05 (0%)</td>
<td>28415E-05 (+17.7%)</td>
<td>2.7248E-05 (+12.9%)</td>
<td>2.4949E-05 (+3.39%)</td>
</tr>
<tr>
<td># of simulations</td>
<td>4600 (1150X)</td>
<td>20 (5X)</td>
<td>816 (204X)</td>
<td>4 (1X)</td>
</tr>
</tbody>
</table>

HDIS is the most accurate approach with 3.39% failure rate estimation error.

**MC**: Monte Carlo, **SS**: Spherical Sampling, **SB**: Statistical Blockade, **HDIS**: the proposed high-dimensional importance sampling

Runtime of 1000 simulations: ~ 5 mins.
SRAM bit-cell circuit

- Experiment results with 90% confidence level on target accuracy:

<table>
<thead>
<tr>
<th></th>
<th>MC</th>
<th>SS</th>
<th>SB</th>
<th>HDIS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Failure rate</td>
<td>2.413E-05 (0%)</td>
<td>28415E-05 (+17.7%)</td>
<td>2.7248E-05 (+12.9%)</td>
<td>2.4949E-05 (+3.39%)</td>
</tr>
<tr>
<td># of simulations</td>
<td>4600 (1150X)</td>
<td>20 (5X)</td>
<td>816 (204X)</td>
<td>4 (1X)</td>
</tr>
</tbody>
</table>

- HDIS is the most accurate approach with 3.39% failure rate estimation error.
  - The performance of SS is acceptable because that’s not actually a real high dimensional circuit. (only part of the transistors operations during the SRAM reading)

**MC**: Monte Carlo, **SS**: Spherical Sampling, **SB**: Statistical Blockade, **HDIS**: the proposed high-dimensional importance sampling

Runtime of 1000 simulations: ~ 5 mins.
SRAM bit-cell circuit

- Experiment results with 90% confidence level on target accuracy:

<table>
<thead>
<tr>
<th></th>
<th>MC</th>
<th>SS</th>
<th>SB</th>
<th>HDIS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Failure rate</td>
<td>2.413E-05 (0%)</td>
<td>28415E-05 (+17.7%)</td>
<td>2.7248E-05 (+12.9%)</td>
<td>2.4949E-05 (+3.39%)</td>
</tr>
<tr>
<td># of simulations (x1000)</td>
<td>4600 (1150X)</td>
<td>20 (5X)</td>
<td>816 (204X)</td>
<td>4 (1X)</td>
</tr>
</tbody>
</table>

- HDIS is the most accurate approach with 3.39% failure rate estimation error.
  - The performance of SS is acceptable because that’s not actually a real high dimensional circuit. (only part of the transistors operations during the SRAM reading)

- It is also the most efficient one with 1150X speedup on MC method

**MC**: Monte Carlo, **SS**: Spherical Sampling, **SB**: Statistical Blockade, **HDIS**: the proposed high-dimensional importance sampling

Runtime of 1000 simulations: ~ 5 mins.
Sense Amplifier circuit

- A circuit with larger number of process variables
- Failure probability
### Sense Amplifier circuit

- A circuit with larger number of process variables

- Failure probability

![Diagram of a Sense Amplifier circuit]
Sense Amplifier circuit

- A circuit with larger number of process variables

- Failure probability

- The classifier in Statistical Blockade (SB) is not blocking any samples. So the efficiency of SB is degraded to the same as MC.
- The Spherical sampling is converging to a totally wrong failure rate.
Sense Amplifier circuit

- Failure probability

- Figure of Merit (demonstrate the fast converging rate of HDIS)
### Evaluation on different failure probabilities:

<table>
<thead>
<tr>
<th>Target failure probability</th>
<th>Monte Carlo (MC)</th>
<th>Spherical Sampling (SS)</th>
<th>Proposed Method (HDIS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>prob:(failure)</td>
<td>8.136e-4</td>
<td>0.2603</td>
<td>7.861e-3 (3.4%)</td>
</tr>
<tr>
<td>#sim. runs</td>
<td>4.800e+4 (24X)</td>
<td>16000 (8X)</td>
<td>2000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Target failure probability</th>
<th>Monte Carlo (MC)</th>
<th>Spherical Sampling (SS)</th>
<th>Proposed Method (HDIS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>prob:(failure)</td>
<td>8.044e-4</td>
<td>0.2541</td>
<td>8.787e-4 (9.2%)</td>
</tr>
<tr>
<td>#sim. runs</td>
<td>4.750e+5 (36X)</td>
<td>8.330e+4 (6.4X)</td>
<td>1.300e+4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Target failure probability</th>
<th>Monte Carlo (MC)</th>
<th>Spherical Sampling (SS)</th>
<th>Proposed Method (HDIS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>prob:(failure)</td>
<td>8.089e-5</td>
<td>0.3103</td>
<td>8.186e-5 (1.2%)</td>
</tr>
<tr>
<td>#sim. runs</td>
<td>5.156e+6 (346X)</td>
<td>1.430e+5 (10X)</td>
<td>1.500e+4</td>
</tr>
</tbody>
</table>

- The accuracy of HDIS agrees with MC on different failure probabilities.
- The efficiency is also consistent under these three cases.
Conclusions and Future Work

- We have proposed a failure probability analysis algorithm, where the failure probability is proved to be always bounded.

- Experiments demonstrated up to 1150X speedup over MC and less than 10% estimation error, while other approaches failed to capture the correct failure rate.

- The proposed algorithm uses mean-shifting, which may be invalid for multiple failure regions. This will be fixed in the future.
Thank you!

Address comments to lhe@ee.ucla.edu
Source of process variations

- 9 variables to model the variations in one CMOS transistor

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>$\sigma/\mu$</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flat-band Voltage ($V_{fb}$)</td>
<td>0.1</td>
<td>V</td>
</tr>
<tr>
<td>Gate Oxide Thickness ($t_{ox}$)</td>
<td>0.05</td>
<td>m</td>
</tr>
<tr>
<td>Mobility ($\mu_0$)</td>
<td>0.1</td>
<td>($m^2/Vs$)</td>
</tr>
<tr>
<td>Doping concentration at depletion ($N_{dep}$)</td>
<td>0.1</td>
<td>($cm^{-3}$)</td>
</tr>
<tr>
<td>Channel-length offset ($\Delta L$)</td>
<td>0.05</td>
<td>m</td>
</tr>
<tr>
<td>Channel-width offset ($\Delta W$)</td>
<td>0.05</td>
<td>m</td>
</tr>
<tr>
<td>Source/drain sheet resistance ($R_{sh}$)</td>
<td>0.1</td>
<td>(Ohm/mm$^2$)</td>
</tr>
<tr>
<td>Source-gate overlap unit capacitance ($C_{gso}$)</td>
<td>0.1</td>
<td>(F/m)</td>
</tr>
<tr>
<td>Drain-gate overlap unit capacitance ($C_{gdo}$)</td>
<td>0.1</td>
<td>(F/m)</td>
</tr>
</tbody>
</table>