A Fully Automated Method to Create Monte Carlo MOSFET Model Libraries for Statistical Circuit Simulations

Jing Wang, Henry Trombley, Josef Watts, Mai Randall, and Richard Wachnik
Outline

- **Background**
  - Motivation for work
  - Existing extraction methods

- **Sequential Variation Determination (SVD) Method**
  - Method
  - Example

- **Results**

- **Conclusions**
Background

- **Accurate Monte Carlo models are an important component of the design kit used by circuit designers.**
  - Simulations at the extremes of the process window are critical for robust designs and high product yields.

- **Monte Carlo models are created by applying random variations to a limited number of the compact model parameters.**

- **Accurate Monte Carlo models must reproduce the device variations seen in the production line. The standard deviations of the model parameters need to be determined such that**
  - Physical model parameters that can be directly measured in devices (L, W, Tox, …) should have variations that are set to match the measured device or target variations.
  - Model parameters that cannot be directly measured need to be set such that Monte Carlo simulations of electrical device characteristics (Vt, Id, Capacitance) match the measured device or target variations.
Background

- **Existing methods for setting model parameter variations:**
  - Guess and check
    - Gives user best control of setting model parameter variations to expected values.
    - Requires many Monte Carlo simulations.
    - Run-to-run Monte Carlo fluctuations can effect ability to converge to accurate solution.
  - Backwards Propagation of Variance (BPV)
    - Simple, quick, and accurate.
    - Requires self-consistent variation targets.
  - Principal Components Analysis (PCA)
    - Purely mathematical solution.
    - Matches variations and correlations in the given data set.
    - Random variables and distributions are not associated with physical variations.
    - Solution may not give expected results for extrapolated conditions.
    - May reproduce sampling artifacts in the data
    - Cannot be used before the line has achieved variation targets
Background

- All of these methods may fail or produce nonphysical solutions if the target variations and sensitivities of the chosen model parameters are not self-consistent.
  - In early phases of technology development, device variation targets may be estimates based on prior technologies.
  - In early phases of technology development a model built to targets may have nonphysical parameter sensitivities.
  - An appropriate set of model parameters may not have been selected for the Monte Carlo model.
Sequential Variation Determination

- Sequential Variation Determination (SVD) is a method related to BPV with a sequential algorithm for solving the set of variance equations.
  - Finds a solution even when inconsistencies exist in the equation set.
  - Warns the user when negative variances are encountered.
  - Solves for dominant model parameters first.
  - Allows the user to restrict the range of searched for the solution to keep model parameter variations in an expected physical range.
Sequential Variation Determination

- **Start with the same set of equations as BPV:**

\[
\left( \frac{\partial M_1}{\partial P_1} \right)^2 \Delta P_1^2 + \left( \frac{\partial M_1}{\partial P_2} \right)^2 \Delta P_2^2 + \ldots + \left( \frac{\partial M_1}{\partial P_{N_P}} \right)^2 \Delta P_{N_P}^2 = (\Delta M_1)^2
\]

\[
\ldots
\]

\[
\left( \frac{\partial M_M}{\partial P_1} \right)^2 \Delta P_1^2 + \left( \frac{\partial M_M}{\partial P_2} \right)^2 \Delta P_2^2 + \ldots + \left( \frac{\partial M_M}{\partial P_{N_P}} \right)^2 \Delta P_{N_P}^2 = (\Delta M_M)^2
\]

- **Where:**
  - \( M_i \) are the Measured device characteristics
  - \( P_j \) are the Model parameters with variance to be determined
Sequential Variation Determination

- Define new variables:
  - \( d_{i,j} = (\partial M_i / \partial P_j)^2 \cdot (\Delta P_j(0))^2 / (\Delta M_i)^2 \)
  - \( \lambda_j = (\Delta P_j / \Delta P_j(0))^2 \)
  - Where \( \Delta P_j(0) \) are the initial estimates for the variations for the model parameters.
  - The RHS which is the total variance for each device measurement has now been normalized to 1.

\[
\begin{pmatrix}
  d_{1,1} & d_{1,2} & d_{1,3} & \cdots & d_{1,N_P} \\
  d_{2,1} & d_{2,2} & d_{2,3} & \cdots & d_{2,N_P} \\
  \vdots & \vdots & \vdots & \ddots & \vdots \\
  \vdots & \vdots & \vdots & \ddots & \vdots \\
  d_{N_M,1} & d_{N_M,2} & d_{N_M,3} & \cdots & d_{N_M,N_P}
\end{pmatrix}
\begin{pmatrix}
  \lambda_1 \\
  \lambda_2 \\
  \vdots \\
  \vdots \\
  \lambda_{N_P}
\end{pmatrix}
= 
\begin{pmatrix}
  1 \\
  1 \\
  \vdots \\
  \vdots \\
  1
\end{pmatrix}
\]
Sequential Variation Determination

- For each row in Matrix $D$, search for a dominating element, defined as

$$d_{i,k} > 0.95 \cdot \sum_{j=1}^{NP} d_{i,j}$$

- If $d_{m,k}$ is identified as the dominating element of row $m$, then $\lambda_k$ is calculated

$$\lambda_k \approx \left(1 - \sum_{j \neq k} d_{m,j}\right)/d_{m,k}$$

which assumes that all unknown $\lambda_j \approx 1$. 
Sequential Variation Determination

- Subtract variance $d_{i,k} \cdot \lambda_k$ from the total normalized variance on the right hand side of the matrix.
  - If any RHS terms are less than 0, then set them equal to 0 and warn user that a negative variance was generated.

- Reduce the size of the Matrix $D$ by removing row $m$ and column $k$.

\[
\begin{pmatrix}
  d_{1,1} & d_{1,2} & \cdots & d_{1,k} & \cdots & d_{1,N_p} \\
  d_{2,1} & d_{2,2} & \cdots & d_{2,k} & \cdots & d_{2,N_p} \\
  \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
  d_{m,1} & d_{m,2} & \cdots & d_{m,k} & \cdots & d_{m,N_p} \\
  \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
  d_{N_M,1} & d_{N_M,2} & \cdots & d_{N_M,k} & \cdots & d_{N_M,N_p}
\end{pmatrix}
\begin{pmatrix}
  \lambda_1 \\
  \lambda_2 \\
  \vdots \\
  \lambda_k \\
  \vdots \\
  \lambda_{N_p}
\end{pmatrix}
= 
\begin{pmatrix}
  1 - d_{1,k} \lambda_k \\
  1 - d_{2,k} \lambda_k \\
  \vdots \\
  \lambda_k \\
  \vdots \\
  1 - d_{N_p,k} \lambda_k
\end{pmatrix}
\]
Sequential Variation Determination

- **Repeat steps in previous two pages:**
  - Search for new dominating elements in new matrix.
  - Solve for $\lambda_k$
  - Reduce matrix size

- **Continue until**
  - Matrix $D$ is reduced to a single element and therefore fully solved, or
  - No more dominating elements are found; solve the remaining matrix by standard methods.

- **Find the estimate of the solution of the model parameter variances.**
  - $\Delta P_j^2 = \lambda_j \Delta P_j(0)^2$
Sequential Variation Determination

- **Check for Convergence**
  - Do all $\lambda_i$ satisfy $0.95 \leq \lambda_i \leq 1.05$?

- If the criteria for convergence are not met, then use the estimate solution for $\Delta P_j$ for the initial guess in the next iteration.
Example

- Below is an illustrative example of the first two reductions of the matrix.
  
- TOXO is identified as the dominant term for Tinv
  
- $\lambda_k$ is calculated for TOXO using $\lambda_k \approx \left(1 - \sum_{j \neq k} d_{m,j}\right) / d_{m,k}$

- Verify that $\lambda_k$ is in expected range and adjust if necessary.

<table>
<thead>
<tr>
<th></th>
<th>MCOV</th>
<th>LOV</th>
<th>TOXO</th>
<th>VFBO</th>
<th>UO</th>
<th>RSW1</th>
<th>$\lambda$</th>
<th>$\nu$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Idlin</td>
<td>0.000021</td>
<td>0.000083</td>
<td>0.399921</td>
<td>0.250455</td>
<td>0.339129</td>
<td>0.007472</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Vtlin</td>
<td>0</td>
<td>0.000002</td>
<td>0.156025</td>
<td>0.812252</td>
<td>0.006202</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Vtsat</td>
<td>0.005751</td>
<td>0.019367</td>
<td>0.046584</td>
<td>0.360000</td>
<td>0.003117</td>
<td>0.000011</td>
<td>0.6944</td>
<td>= 1</td>
</tr>
<tr>
<td>Tinv</td>
<td>0</td>
<td>0</td>
<td>1.440000</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Idsat</td>
<td>0.004218</td>
<td>0.014498</td>
<td>0.148032</td>
<td>0.149657</td>
<td>0.055983</td>
<td>0.069135</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Cov</td>
<td>0</td>
<td>0.610233</td>
<td>0.143791</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

© 2012 IBM Corporation
Example

- Subtract the variance due to TOXO term from normalized target variance on RHS using:
  \[ v'_i = v_i - d_{i,k} \cdot \lambda_k \]
  - Any \( v'_i < 0 \) indicates a negative residual variance. This is due to inconsistent target variances or another problem. Warn the user of the problem, then set \( v'_i = 0 \) and continue.

<table>
<thead>
<tr>
<th>( MCOV )</th>
<th>( LOV )</th>
<th>( TOXO )</th>
<th>( VFBO )</th>
<th>( UO )</th>
<th>( RSW1 )</th>
<th>( \lambda )</th>
<th>( \nu )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( Idlin )</td>
<td>0.000021</td>
<td>0.000083</td>
<td>0.399921</td>
<td>0.250455</td>
<td>0.339129</td>
<td>0.007472</td>
<td>1</td>
</tr>
<tr>
<td>( Vtlin )</td>
<td>0</td>
<td>0.000002</td>
<td>0.156025</td>
<td>0.812252</td>
<td>0.006202</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>( Vtsat )</td>
<td>0.005751</td>
<td>0.019367</td>
<td>0.046584</td>
<td>0.360000</td>
<td>0.003117</td>
<td>0.000011</td>
<td>0.6944</td>
</tr>
<tr>
<td>( Tinv )</td>
<td>0</td>
<td>0</td>
<td>1.440000</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>( Idsat )</td>
<td>0.004218</td>
<td>0.014498</td>
<td>0.148032</td>
<td>0.149657</td>
<td>0.055983</td>
<td>0.069135</td>
<td>1</td>
</tr>
<tr>
<td>( Cov )</td>
<td>0</td>
<td>0.610233</td>
<td>0.143791</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
Example

- Remove row \( m \) and column \( k \) from the Matrix \( D \) and associated terms in \( \lambda \) and \( \nu \).

<table>
<thead>
<tr>
<th></th>
<th>( MCOV )</th>
<th>( LOV )</th>
<th>( TOXO )</th>
<th>( VFBO )</th>
<th>( UO )</th>
<th>( RSW1 )</th>
<th>( \lambda )</th>
<th>( \nu )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( Idlin )</td>
<td>0.000021</td>
<td>0.000083</td>
<td>0.399921</td>
<td>0.250455</td>
<td>0.339129</td>
<td>0.007472</td>
<td>1</td>
<td>0.7223</td>
</tr>
<tr>
<td>( Vtlin )</td>
<td>0</td>
<td>0.000002</td>
<td>0.156025</td>
<td>0.812252</td>
<td>0.006202</td>
<td>0</td>
<td>1</td>
<td>0.8916</td>
</tr>
<tr>
<td>( Vtsat )</td>
<td>0.005751</td>
<td>0.019367</td>
<td>0.046584</td>
<td>0.360000</td>
<td>0.003117</td>
<td>0.000011</td>
<td>0.6044</td>
<td>≈ 0.0000</td>
</tr>
<tr>
<td>( Tinv )</td>
<td>0</td>
<td>0</td>
<td>1.440000</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0.9677</td>
</tr>
<tr>
<td>( Idsat )</td>
<td>0.004218</td>
<td>0.014498</td>
<td>0.148032</td>
<td>0.149657</td>
<td>0.055983</td>
<td>0.069135</td>
<td>1</td>
<td>0.8972</td>
</tr>
<tr>
<td>( Cov )</td>
<td>0</td>
<td>0.610233</td>
<td>0.143791</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0.9001</td>
</tr>
</tbody>
</table>
Example

- Identify the next dominant term for the matrix and continue.

<table>
<thead>
<tr>
<th></th>
<th>MCOV</th>
<th>LOV</th>
<th>VFBO</th>
<th>UO</th>
<th>RSW1</th>
<th>λ</th>
<th>ν</th>
</tr>
</thead>
<tbody>
<tr>
<td>Idlin</td>
<td>0.000021</td>
<td>0.000083</td>
<td>0.250455</td>
<td>0.339129</td>
<td>0.007472</td>
<td>1</td>
<td>0.7223</td>
</tr>
<tr>
<td>Vtlin</td>
<td>0</td>
<td>0.000002</td>
<td>0.812252</td>
<td>0.006202</td>
<td>0</td>
<td>1.4751</td>
<td>0.8916</td>
</tr>
<tr>
<td>Vtsat</td>
<td>0.005751</td>
<td>0.019367</td>
<td>0.360000</td>
<td>0.003117</td>
<td>0.000011</td>
<td>1</td>
<td>0.9677</td>
</tr>
<tr>
<td>Idsat</td>
<td>0.004218</td>
<td>0.014498</td>
<td>0.149657</td>
<td>0.055983</td>
<td>0.069135</td>
<td>1</td>
<td>0.8972</td>
</tr>
<tr>
<td>Cov</td>
<td>0</td>
<td>0.610233</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0.9001</td>
</tr>
</tbody>
</table>
Results

- Monte Carlo simulation comparison to device tolerance targets.
  - Long Device Idlin/Vtlin
  - Short Device Idsat/Vtsat
- Variance of the simulated data matches the target variances.
Summary

- The Sequential Variation Determination method quickly and automatically solves for the model parameter variances required for a Monte Carlo model to match the specified tolerances on a set of measured device characteristics.

- The solution by construction has no negative variances and matches all of the device parameter tolerances.

- If the exact mathematical solution includes negative variance, then the algorithm identifies them to the users so corrections can be made if required.