Advanced Computational Methods for VLSI Systems

Lecture 5
Statistical VLSI modeling and analysis

Zhuo Feng
Example of Static Timing Analysis

- Arrival time: input -> output, take max
- Required arrival time: output -> input, take min
- Slack = required arrival time – arrival time
SSTA and Infrastructure

Variability
- Process Characterization
- Interconnect Gate Models
- Delay Calculators
Basic SSTA Research Issues

- Corner-based analysis is too pessimistic

- Process variation modeling
  - PCA, Quad-tree, Distance based, Gridless

- Arrival time (AT)/required arrival time (RAT) propagation
  - Path based, block based

- Accuracy/efficiency tradeoff
  - First order, quadratic

- Flexibility and extensibility
  - Gaussian vs. non-Gaussian

- Don’t forget the characterization underpins
  - Interconnects and gates
SSTA Literatures


- C. Visweswariah et al, “First-order incremental block-based statistical timing analysis,” DAC’04. [Reading assignment]

- Y. Zhan, A. Strojwas, X. Li, and L. Pileggi, “Correlation-aware statistical timing analysis with non-Gaussian delay distributions,” DAC’05.


- S. Onaissi and F. Najm, “A linear-time approach for static timing analysis covering all process corners,” ICCAD’06.

- Z. Feng, P. Li, Y. Zhan, “Fast second-order statistical static timing analysis using parameter dimension reduction,” DAC’07.

- And many more…
Principle Component Analysis (PCA)

- Equiprobable contours of a Gaussian vector
  - Ellipsoids
  - The axes of the ellipsoids are not parallel to the coordinate axes if the covariance matrix is not diagonal

- PCA can rotate the ellipsoids to make the axes parallel to the coordinate axes
  - Decorrelate
Principle Component Analysis (PCA)

- Compute the normalized eigenvectors of the covariance matrix $\Sigma: V = [v_1, v_2, \ldots, v_N]$

- $V_1$ are called the first principle component (PC1), $V_2$ the second principle component (PC2) …
  - These are the axes of the ellipsoid

- Perform coordinate transform for a data point $x$ (in the original coordinate), but with the mean subtracted
  - $z = x - m$
Principle Component Analysis (PCA)

- Find the new coordinates for z via projection:
  - $c = V^T z$
  - Further scale $c$ by the square roots of eigenvalues:
    $c = \text{diag} (\sqrt{\lambda_1}, \sqrt{\lambda_2}, \ldots) V^T z$

- $x = \text{diag} (\sqrt{\lambda_1}, \sqrt{\lambda_2}, \ldots) Vc + m$

- Mean of $c = \mu$

- Covariance of $c = \Sigma$

- From now on, we assume $x = [x_1, x_2, \ldots]$ are a set of PCs – i.e., PCA has been already applied
Assumptions behind PCA

- PCA assumes *normality*

- For Gaussian random variables, uncorrelatedness implies independence
  - Breaks down for non-Gaussian distributions
  - PCA does not achieve independence for non-Gaussian parameters

- More general ICA (independent component analysis) can be applied to non-Gaussian variables
PCA based Process & Delay Model

- Partition the entire chip into several bins
- Transistors/interconnects in each bin are impacted by several variability sources
- Variations coming from different bins are correlated (spatial correlations) – captured by a covariance matrix
- Decorrelate via PCA
PCA based Process & Delay Model

- Express all gate/wire delays in terms of linear functions of PCs, possibly with a local random component – canonical form:

\[ A = a_0 + \sum_{i=1}^{N} a_i x_i + a_{N+1,a} x_{N+1,a} \]

Local normal variation: \(N(0,1)\)

- Block-based first-order SSTA
  - Need to propagate ATs/RATs in the canonical form
  - Two key atomic operations: ADD & MAX
Atomic Operation: ADD

- Where do I see ADD?
- ADD is relatively simple
- How to process an ADD?
Atomic Operation: MAX

- Where do I see MAX?
- Handling of a MAX gets more involved
  - Need to play statistics tricks
Atomic Operation: MAX

- Covariance between A & B:

\[ A = a_0 + \sum_{i=1}^{N} a_i x_i + a_N x_{N+1,a} \]

\[ B = b_0 + \sum_{i=1}^{N} b_i x_i + b_N x_{N+1,b} \]
Atomic Operation: MAX

- **Tightness probability** $T_A = \Pr(A > B)$

\[
\phi(x) \equiv \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right)
\]

\[
\Phi(y) \equiv \int_{-\infty}^{y} \phi(x) \, dx
\]

\[
\theta \equiv (\sigma_A^2 + \sigma_B^2 - 2\rho \sigma_A \sigma_B)^{1/2}
\]

\[
T_A = \Phi\left(\frac{a_0 - b_0}{\theta}\right)
\]

Clark’s formulae
Atomic Operation: MAX

- Mean and variance can be computed for the max

\[ E[\max(A, B)] = a_0 T_A + b_0 (1 - T_A) + \theta \phi \left[ \frac{a_0 - b_0}{\theta} \right], \]
\[ \text{var}[\max(A, B)] = (\sigma_A^2 + a_0^2) T_A + (\sigma_B^2 + b_0^2) (1 - T_A) + (a_0 + b_0) \theta \phi \left( \frac{a_0 - b_0}{\theta} \right) - \left\{ E[\max(A, B)] \right\}^2. \]

- Match the mean and variance in the canonical form of the max
  - Use \( T_A \) to weight between each coefficients of the PCs
  - Adjust the coefficient for the local random component to match the variance
Statistical Optimization

- In addition to the timing yield, statistical sensitivities/criticality can be computed to guide statistical circuit optimization (i.e., gate/wire sizing).

- However, modeling statistical leakage power variation requires non-Gaussian models – e.g. lognormal distributions.
Issues with First-order SSTA

- Gate/wire delays can be nonlinear functions of process parameters – large process variability

- MAX operations are intrinsically nonlinear

- Linear assumptions in first-order SSTA can incur error

- Quadratic SSTA may significantly improve the accuracy at the cost of higher runtime – curse of dimensionality [Zhan et al DAC’05].

- Parameter dimension reduction technique can alleviate the cost of quadratic SSTA [Feng et al DAC’07]
Issues with First-order SSTA

- There exist works which attempt to relax the Gaussian assumption.

- This boils down to some sort of “parameterized” SSTA approach.

- Question: what does the max of two parametric functions look like in 1D?
SSTA Modeling Infrastructure

- Variational Gate delay models
- Variational Interconnect models
Current Source Gate Model (CSM)

- Efficient and accurate gate delay modeling is essential for timing analysis

- Traditional waveform dependent gate models can not handle complex input signals due to the increasing crosstalk noise or inductive coupling effects

- Waveform Independent Models (WIM) are needed to better capture the nonlinear dynamic effects
  - Waveform Independent Model: P. Li (ICCD 05’), C. Amin (DAC 06’)

- Parameterized WIM is essential to capture the PVT variations
Waveform based models encode the dynamics in the waveforms

Limited to simple signal shapes

Waveform Independent Model (WIM) is able to model dynamics directly

Can handle complex waveforms
Model Composition

- Input stage (2\textsuperscript{nd} order RC stage) + the nonlinear input/output current/charge LUTs

### Diagram

- **Input Stage (2\textsuperscript{nd} order RC stage):**
  - \( H(S) \)
  - Internal delay
  - Internal control
  - Node voltage

- **Static/Dynamic Nonlinearities:**
  - \( V_c \)
  - \( V_c \)

- **Output DC/Capacitive Nonlinearities:**
  - \( V_o \)

- **Nonlinear Components:**
  - \( R_1, R_2 \)
  - \( C_1, C_2 \)
  - \( I_n(V_C, V_{out}) \)
  - \( Q_{nc}(V_c, V_{out}) \)
Nonlinear DC Current

- **Characterize the DC characteristics of the gate**
  - Sweep both the input and output from 0 to Vdd
  - Measure the output currents and build the 2D current look up table

![Diagram showing DC voltage sweeps and DC Current LUT](image-url)
The 2nd order RC stage can be characterized by its poles:

\[
\frac{k_1}{s + p_1} + \frac{k_2}{s + p_2}
\]

\[k_1 = -k_2, \quad k_1 = \frac{p_1 p_2}{p_2 - p_1}\]

Apply a training input waveform to find the output current.
Arbitrary PWL input can be decomposed to ramp inputs:

For the ramp input $u(t) = at$, the response due to the input stage transfer function $y(t)$ is:

$$y(t) = ak_1 \left( - \frac{1}{p_1^2} + \frac{t}{p_1} + \frac{1}{p_1^2} e^{-p_1 t} \right)$$

All the $n$ ramp inputs lead to the response:

$$y_1(t) = \sum_{j=1}^{n} \frac{k_1 x_j}{p_1^2} \left( -1 + p_1(t-t_j) + e^{-p_1(t-t_j)} \right) U(t-t_j)$$

Use nonlinear least square to optimize $p_1$ and $p_2$
The internal control voltage can be expressed as:

\[ V_c(t) = \sum_{i=1}^{2} \sum_{j=1}^{n} \frac{k_i \alpha_j}{p_i^2} \left[ -1 + p_i(t - t_j) + e^{-p_i(t - t_j)} \right] U(t - t_j) \]

The error function in terms of \(p_1\) and \(p_2\) is given by:

\[ E(p_1, p_2, i) = \left( I(V_o, t_i) - I_n(V_c(t_i), V_{o,dc}) \right)^2 \]

Let \(p_2 = p_1 + \Delta p\), NL least square is applied to minimize the above error.

The above optimal poles \(p_1\) and \(p_2\) are used to characterize the NL output capacitance.
Nonlinear Output Capacitance

- Nonlinear charge w.r.t. the control voltage and output voltage is to be extracted

- 2D nonlinear charge LUT can be built based on the previously extracted 2D current LUT and the input RC stage:

- Iterations may be required to improve the accuracy
  - Iterations may not be needed in practice
Overall Extraction Flow

Nonlinear DC Current Characterization

Post tuning

Accurate?

Yes

Not yet

End

Input Stage Characterization

Nonlinear Output Capacitance Characterization

Volterra series

Integral equations

DC current LUT

Update NL

Charge LUT

Update

Input stage
- **Delay errors**
  - **Better delay**: a pure delay element in front of the input stage is included
  - **Worse delay**: two poles have to be changed to match the delay while the nonlinear charge LUT is updated based on these poles

- **Slew errors**
  - **Faster output transition**: increase a portion of the nonlinear charges in the LUT
  - **Slower output transition**: decrease a portion of the nonlinear charges in the LUT
Post Tuning

Characterize DC LUT

Optimize p1 and p2 of the input stage

Characterize input NL capacitor

Characterize output NL capacitor

Verify the model accuracy

Accurate?

Better delay?

Add a Delay element

Worse delay?

No

No

Yes

End

Increase p2

Reduce NL output cap

Increase NL output cap

Slower rising?

Faster rising?

No

No

Yes
Parameterized Waveform Independent Model

- Build parameterizable WiM models to capture the process-voltage-temperature (PVT) variations
  - The response surface modeling technique adopting the two-level composite design plan is applied

- Second order parameterized WiM is accurate for typical process variation ranges
Experimental Results (3-input OR)

- A complex input drives a four-stage buffer with a pi-model load

![Diagram of a four-stage buffer with a complex input driving it](image)

![Graph showing voltage over time for different input scenarios](image)
Experimental Results (4-stage buffer)

- 4-stage buffer gate is driven by a complex input

![Diagram of a 4-stage buffer gate](image)
Experimental Results (XOR)

- An XOR gate is driven by a complex input

![Diagram of an XOR gate with inputs A and B and output labeled as 'out']

![Graph showing experimental results with time on the x-axis and voltage on the y-axis, comparing Input, Spice, and WiM]
Experimental Results

- Delay/slew errors (%) and speedups for two non-ramp inputs

<table>
<thead>
<tr>
<th>Design</th>
<th>Input 1</th>
<th>Input 2</th>
<th>Sp. Up</th>
<th>Design</th>
<th>Input 1</th>
<th>Input 2</th>
<th>Sp. Up</th>
</tr>
</thead>
<tbody>
<tr>
<td>INVX1</td>
<td>0.08</td>
<td>0.22</td>
<td>0.21</td>
<td>0.22</td>
<td>83</td>
<td></td>
<td></td>
</tr>
<tr>
<td>INVX4</td>
<td>0.53</td>
<td>0.92</td>
<td>1.06</td>
<td>0.26</td>
<td>104</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BUF2X1</td>
<td>1.71</td>
<td>0.61</td>
<td>0.49</td>
<td>0.94</td>
<td>128</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BUF2X4</td>
<td>4.30</td>
<td>1.55</td>
<td>1.84</td>
<td>2.48</td>
<td>120</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BUF4</td>
<td>4.80</td>
<td>0.75</td>
<td>4.77</td>
<td>0.75</td>
<td>179</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND2X1</td>
<td>3.20</td>
<td>0.60</td>
<td>1.1</td>
<td>0.60</td>
<td>98</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND2X4</td>
<td>5.13</td>
<td>1.91</td>
<td>3.75</td>
<td>1.84</td>
<td>115</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND2X8</td>
<td>6.55</td>
<td>4.10</td>
<td>4.78</td>
<td>3.81</td>
<td>151</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND3X1</td>
<td>2.87</td>
<td>0.79</td>
<td>2.83</td>
<td>1.11</td>
<td>158</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND3X4</td>
<td>5.27</td>
<td>1.31</td>
<td>5.17</td>
<td>2.18</td>
<td>104</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Post Tuning

- A large 4-stage buffer via post tuning
Experimental Results (Coupling Noise)

- A 4-bit coupled bus driven by buffers
Experimental Results (Coupling Noise)

- Two coupled nets driven by an AOI and a NAND
Experimental Results (Variational Analysis)

- Variational analysis of an OR3

![Variational analysis graph]

- Input
- PWiM
- Spice

- Nominal
  - PWiM 40% L 15% Vt
  - PWiM 40% L -15% Vt

- Spice
  - 40% L 15% Vt
  - 40% L -15% Vt
Experimental Results (Variational Analysis)

- PWiM results under a non-ramp input

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Var.</td>
<td>Ave.</td>
<td>Max</td>
</tr>
<tr>
<td>INVX1</td>
<td>58</td>
<td>4.1</td>
<td>11</td>
</tr>
<tr>
<td>BUF2X4</td>
<td>78</td>
<td>7.3</td>
<td>17</td>
</tr>
<tr>
<td>BUF4</td>
<td>80</td>
<td>8.0</td>
<td>19</td>
</tr>
<tr>
<td>AOI2X2</td>
<td>129</td>
<td>7.4</td>
<td>34.2</td>
</tr>
<tr>
<td>OR3X4</td>
<td>87</td>
<td>5.4</td>
<td>16.8</td>
</tr>
</tbody>
</table>
Parameterized Interconnect MOR

- Geometrical variations can be captured in MOR via moment matching

\[
G(p_1, p_2, \ldots, p_M) \\
C(p_1, p_2, \ldots, p_M)
\]

- High dimensional parameter space is still a problem
  - Complex models and high analysis cost
Parameter Dimension Reduction

- Reduction in both circuit size and parameter dimension is desired.
- Most of available techniques only address size reduction.

### Parametric Network

- Nominal MOR: $H(s)$
- Parameterized MOR in $P$: $\hat{H}(P,s)$
- Parameterized MOR in $Z$: $\hat{H}(Z,s)$

### Diagram Elements

- $H(P, S)$: Parametric Network
- $H(s)$: Nominal MOR
- $\hat{H}(s)$: Parameterized MOR
- $s$: Frequency
- $P$: Parameter space
- $Z$: Parameter space
- $Z_1$: Parameterized MOR in $Z$
Principle Component Analysis

- Widely used as a dimension reduction technique
- Limitation: neglect the structural info of a given design

Parameter Space $X$

Performance Space $Y$

$Y = f(X)$
Performance Oriented Parameter Reduction

- Idea: find a new (smaller) set of parameters that are most statistically significant to the performances of interest
- Consider: distributions of parameters + correlation with performances

\[ X = [x_1, x_2, \ldots, x_N] \]

\[ Z = [z_1, z_2, \ldots, z_R], R \ll N \]

\[ z_i = f(x_1, x_2, \ldots, x_N) \]

[Feng & Li, ICCAD’06]
Linear Reduced Rank Regression (RRR)

- **Linear regression model**
  \[ Y = CX + \varepsilon \]
  - X: N underlying process parameters
  - Y: M performances of interest
  - C: M x N: regression matrix
  - \varepsilon: error

- **Linear reduced rank regression model**
  \[ Y = CX + \varepsilon \]
  \[ \approx A_R B_R X + \varepsilon \]
  - M x R
  - R x N

- **Objectives**
  - Find \( A_R \) and \( B_R \) to minimize the error in a statistical sense
  - Compose a new set of R input variables \( Z = B_R X \) (R << N)
Linear Reduced Rank Regression (RRR)

■ Optimal RRR model

\[ Y \approx A_R B_R X + \tilde{\varepsilon} \]

\[ A_R = \Omega^{-1/2} U, \quad B_R = U^T \Omega^{-1/2} \Sigma_{YX} \Sigma_{XX}^{-1} \]

For any SPD \( \Omega \)

\[ U = [u_1, \ldots, u_r] \]

Eigenvectors corresponding to the \( R \) largest eigenvalues of

\[ D = \Omega^{1/2} \Sigma_{YX} \Sigma_{XX}^{-1} \Sigma_{XY} \Omega^{1/2} \]

■ Minimize

\[ E[tr\{\Omega^{1/2} (Y - A_R B_R X)(Y - A_R B_R X)^T \Omega^{1/2}\}] \]

Weighted Variances of Model Error
PCA vs RRR

- Reduced rank regression (RRR)

Get input data $X$
- Covariance Matrix $\Sigma_{xx}$
- SVD on $\Sigma_{xx}$
- Data reduction

Maximize input variance

Get output data $Y$
- Covariance Matrix $\Sigma_{yx}$
- SVD on $\Sigma_{yxx}^{-\frac{1}{2}}$
- Parameter reduction

Minimize errors in $Y$
RRR for Statistical Circuit Modeling

- **Standard RRR offers parameter reduction under a linear regression model framework**
  - Limited for modeling large range process variations

- **Performance oriented parameter reduction requires statistical information of performances (Y)**
  - Expensive to obtain via Monte-Carlo sampling

- **Approach:**
  - Extend the standard RRR to nonlinear regression based RRR
  - Use easily computed performance metrics as cheap surrogates
Nonlinear RRR

- Quadratic regression
\[ Y = C_1 X + C_2 (X \otimes X) + \varepsilon \]
\[ X \otimes X = [x_1^2, x_1 x_2, \cdots, x_{N-1} x_N, x_N^2]^T \]

- Parameter reduction
\[ Y = A_{R1} (B_{R1} X) + A_{R2} \left((B_{R1} X) \otimes (B_{R1} X)\right) + \widetilde{\varepsilon} \]

New parameters Z

- Difficult to find an optimal \( B_{R1} \) that minimizes the model error variances
- Reformulate the problem instead
Nonlinear RRR

- Define an augmented parameter set

\[
\bar{X} = \begin{bmatrix} X \\ X \otimes X \end{bmatrix}
\]

- Build a reduced-rank linear regression model based on

\[
Y = A_R [B_{R1} \quad B_{R2}] \begin{bmatrix} X \\ X \otimes X \end{bmatrix} + \varepsilon
\]

\[
\Sigma_{XX}, \quad \Sigma_{XY}
\]

- Construct the new parameter set

\[
Z = B_{R1} X + B_{R2} X \otimes X \approx B_{R1} X
\]
Computation of Covariance Matrices

- Choose transfer function moments as performance metrics (Y) in RRR
  - Strongly connected with the interconnect (timing) performance

- Quadratic parametric forms of moments can be computed efficiently

\[
m_k = m_{k0} + \sum_{i=1}^{N} \alpha_{k,i} x_i + \sum_{i=1}^{N} \beta_{k,i,i} x_i^2 + \sum_{i=1}^{N} \sum_{j=1}^{i-1} \beta_{k,i,j} x_i x_j
\]

- Closed-form expressions available for covariance matrices (e.g. for Gaussian interconnect geometrical variations)
Coupled RC lines

- Two coupled RC lines
  - RC values computed using closed-formulas based on geometry
  - Divide the two lines into five regions
  - For each region, we consider the geometrical variations: wire width (W), wire thickness (T), wire spacing (S) and dielectric layer thickness (H)
  - Total number of variations: 20
Coupled RC lines

- Full 20-parameter model: 5000 LHS samples
- RRR based 3-parameter reduced-parameter model: 800 LHS samples

Delay variations
Results – Coupled RC lines

- **Errors of delay mean and standard deviation**

<table>
<thead>
<tr>
<th>set</th>
<th>(\sigma_W)</th>
<th>(\sigma_H)</th>
<th>(\sigma_T)</th>
<th>(\sigma_S)</th>
<th>4K LHS Samp. Mean/Std.</th>
<th>800 LHS Samp. Mean/Std.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 Para.</td>
<td>2 Para.</td>
</tr>
<tr>
<td>1</td>
<td>5%</td>
<td>10%</td>
<td>10%</td>
<td>10%</td>
<td>0.00%/1.11%</td>
<td>0.94%/3.96%</td>
</tr>
<tr>
<td>2</td>
<td>10%</td>
<td>5%</td>
<td>10%</td>
<td>10%</td>
<td>0.00%/0.78%</td>
<td>1.50%/5.44%</td>
</tr>
<tr>
<td>3</td>
<td>10%</td>
<td>10%</td>
<td>5%</td>
<td>10%</td>
<td>0.00%/2.07%</td>
<td>0.88%/3.32%</td>
</tr>
<tr>
<td>4</td>
<td>10%</td>
<td>10%</td>
<td>10%</td>
<td>5%</td>
<td>0.09%/0.56%</td>
<td>1.40%/5.67%</td>
</tr>
</tbody>
</table>

- **Reduced-parameter model errors: 1000 random samples**

![1-Parameter Model](image1)

![3-Parameter Model](image2)
Coupled RC lines

- Composition of the new variation parameters $Z=[Z1 \ Z2 \ Z3]'$

Most important factors!
Results – Coupled RC lines

- Reduced-order reduced-parameter model
  - Parameters: 20 → 1      Order: 204 → 12
- 20x parameter dimension reduction leads to a compact passive parameterized reduced order model
Results – An RC Mesh

- Divide into nine regions and assign 27 geometrical variations

![Diagram showing a grid with 27 regions and a graph with lines indicating mean, standard deviation, and sigma values.]
Results – An RLC Line

- 30 geometrical variations
- Reduced-order reduced-parameter model
  - Parameters: 1       Order: 16

![Frequency Response](image1)

![Composition of New Parameters](image2)
Fast Quadratic SSTA using Parameter Reduction

- Recall: quadratic SSTA is much more expensive than first-order SSTA

- The runtime grows quadratically in the number of process parameters

- Can bring parameter reduction up to the timing analysis level to alleviate the curse of dimensionality
Global & local variation sources

- Global variations (inter-die variations): impact the whole chip (introduce few variables)
- Local variations (intra-die variations): may impact smaller area (introduce many variables)
Motivations: SSTA via Moment Matching

- **Runtime of Quadratic SSTA via Moment Matching increases quickly with the number of parameters**
  - Very limited number of variation sources can be considered
  - Taking local variations into account is too expensive

[Zhan et al, DAC’05]
Theoretic framework: conceptual

- Reduced rank regression (RRR) in quadratic SSTA
  - Transform local parameters to reduced parameters using RRR
  - Perform quadratic SSTA on new parameter space
    - Including global variables and reduced local variables

![Diagram showing the process of transforming local variables into reduced variables and performing quadratic SSTA](image-url)
Theoretic framework: actual

- Propagate global parameters and reduce local ones
Formulation: reduced rank regression

- Linear regression model
  \[ Y = CX + DG + \varepsilon \]
  - \( X, G \): underlying local/global parameters
  - \( Y \): M performances of interest (signal arrival times)
  - \( C, D \): regression matrices
  - \( \varepsilon \): model error

- Linear reduced rank regression approximate \( C \) by lower rank matrices
  \[ Y \approx A_R B_R X + DG + \tilde{\varepsilon} \]

- Objectives
  - Find \( A_R \) and \( B_R \) to minimize the error in a statistical sense
  - Compose a new set of R input variables \( Z = B_R X \) (\( R \ll N \))
RRR with two regressors: general form

- **Optimal RRR model**
  \[ Y \approx A_R B_R X + DG + \tilde{\varepsilon} \]
  \[
  A_R = U, \quad B_R = U^T \Sigma_{YX,G}^{-1} \Sigma_{XX,G}^{-1} \\
  D = \Sigma_{YG}^{-1} \Sigma_{GG}^{-1} - A_R B_R \Sigma_{XG}^{-1} \Sigma_{GG}^{-1}
  \]

- **Minimize the variances of model error**
  \[ E[tr\{(Y - A_R B_R X - DG)(Y - A_R B_R X - DG)^T\}] \]

- **A new set of variable Z is obtained by:**
  \[ Z \approx B_R X \]

- **How to use them in SSTA?**

\[ U = [u_1, \ldots, u_r] \]
Implement RRR in SSTA

- Implementing RRR in SSTA:
  - No correlation between the global (G) and local (X) variables: \( \sum X_G = 0 \)

- Formulas are simplified:
  \[
  A_R = U, \quad B_R = U^T \sum_{YX} \sum_{XX}^{-1}
  \]

- Eigenvectors correspond to the \( R \) largest eigenvalues of
  \[
  Q = \sum_{YX} \sum_{XX}^{-1} \sum_{XY}
  \]

- RRR is used to identify the redundancy in the local variables

How to compute?
Implement RRR in SSTA

- Partition circuit into building blocks (divide and conquer)
- For partition $i$, the signal arrival time $AT_k$ is written as:

$$AT_k = \tilde{M}_k + \beta_k^T G + G^T A_k G + c_k^T X_i + X_i^T B_k X_i$$

where

$$X_i = \begin{bmatrix} X_{loc-i} \\ Z_{i-1} \end{bmatrix} \in \mathbb{R}^{n_i}$$

- Use all the signal arrival times in partition $i$ to compute $Q$:

$$\Sigma_{YX} = \begin{bmatrix} c_1, \cdots, c_k, \cdots, c_m \end{bmatrix}^T \rightarrow Q = \Sigma_{YX} \Sigma_{XX}^{-1} \Sigma_{XY}$$

- Form $Br$ using the few dominant eigenvectors of $Q$
  - Compute the new parameter set $Z=Br \ X$ and the inverse transform
  - Express the signal arrival time using $Z$ and $G$

$$AT_k \approx \tilde{M}_k + \beta_k^T G + G^T A_k G + c_k^T TZ_i + Z_i^T T^T B_k T Z_i$$

$X_i = TZ$

Recover parameters
The signal arrival time $AT_k$ can be converted to:

$$AT_k \approx \bar{M}_k + \beta_k^T G + G^T A_k G + \tilde{c}_k^T Z_i + Z_i^T \tilde{B}_k Z_i$$

- Moment matching based Max operation can be performed for $G$ and $Z$
Example: partitioned by logic level
Results: impact of local process variations

- **ISCAS85-c880**

<table>
<thead>
<tr>
<th>ckt</th>
<th># of Part.</th>
<th># of local variables</th>
<th># of total variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>c880</td>
<td>12</td>
<td>12</td>
<td>14</td>
</tr>
</tbody>
</table>

PDFs of ISCAS c880: errors by neglecting local variation sources
Results: impact of local process variations

- **c5315**

<table>
<thead>
<tr>
<th>ckt</th>
<th># of Part.</th>
<th># of local variables</th>
<th># of total variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>c5315</td>
<td>24</td>
<td>24</td>
<td>26</td>
</tr>
</tbody>
</table>

PDFs of ISCAS c5315: errors by neglecting local variation sources