Symbolic Model Reduction for Linear and Nonlinear DAEs

Symposium on Recent Advances in MOR
TU Eindhoven, The Netherlands
November 23rd, 2007

Thomas Halfmann
thomas.halfmann@itwm.fraunhofer.de
Overview

Introduction to symbolic analysis and approximation
motivation of symbolic methods
principles of symbolic simplification
approximation methods for linear systems
nonlinear symbolic model generation

EDA tool Analog Insydes
platform overview

Methodology transfer
mechatronics
Symbolic Methods

numerical simulation

Gds$M9$ $gm$M6
CC (Gds$M9$ $-$ $gm$M6)

symbolic analysis

Fraunhofer Institut Techno- und Wirtschaftsmathematik
Design Flow

Application fields of symbolic methods:
- error analysis
- gaining system understanding
- specification transfer between levels
- automated behavioral model generation
- multi-physics modeling and system simulation

Fraunhofer Institute for Integrated Circuits (ITWM)
Symbolic Analysis

Folded-Cascode OpAmp

Problem
What causes resonance at 10 MHz?

Classic approach
parameter variations and numerical simulation failed because of huge number of parameters

Symbolic analysis
calculation of the transfer function and derivation of a symbolic formula for the resonance frequency

Complexity problem
exact symbolic transfer function consists of more than $5 \cdot 10^{19}$ terms
printed: paper stack with a height of $15 \cdot 10^9$ miles
differential order: 19
Symbolic Analysis

Folded-Cascode OpAmp

Problem
What causes resonance at 10 MHz?

Classic approach
parameter variations and numerical simulation failed because of huge number of parameters

Symbolic analysis
calculation of the transfer function and derivation of a symbolic formula for the resonance frequency

Complexity problem ⇒ symbolic approximation

\[ \frac{1}{s^2} = \frac{(CC0 + CL) \cdot gmMN6}{2 \cdot CC0 \cdot CL} + \frac{\sqrt{CgsMP15 \cdot gmMN6 \cdot CgsMP15 \cdot (CC0 + CL)^2 \cdot gmMN6 - 4 \cdot CC0^2 \cdot CL \cdot gmMN6 \cdot 2 \cdot CC0 \cdot CgsMP15 \cdot CL}}{2 \cdot CC0 \cdot CgsMP15 \cdot CL} \]
Numerical Validation

first design

design after symbolic analysis
Complexity Problem

Text book: \[ V_{out} = \frac{R_C}{R_E} \]

Exact solution using computer algebra

![Diagram with 132 terms]
Symbolic Approximation: Basic Idea

**Netlist**

\[ R_1 = R_2 = 10 \, \Omega \]
\[ R_3 = R_4 = 1000 \, \Omega \]

**Symbolic analysis: exact transfer function**

\[ \frac{R_2 R_4}{R_2 + R_1 R_3 + R_1 R_4 + R_2 R_3 + R_2 R_4} \]

**Numerical evaluation**

\[ \frac{10^4}{1 \times 10^2 + 10^4 + 10^4 + 10^4 + 10^4} \]

**Transfer function after symbolic approximation**

\[ \frac{R_2 R_4}{(R_1 + R_2)(R_3 + R_4)} \]

Relative error: 0.0025
Symbolic Approximation: Basic Idea

Netlist

Symbolic analysis: exact transfer function

Numerical evaluation

Transfer function after symbolic approximation

Problem:

Exact symbolic transfer function can not be calculated for real world problems.

Relative error: 0.0025
## Symbolic Approximation: Ranking

<table>
<thead>
<tr>
<th>symbol</th>
<th>position</th>
<th>relative error</th>
</tr>
</thead>
<tbody>
<tr>
<td>-$R_2$</td>
<td>(1, 2)</td>
<td>2.488E-3</td>
</tr>
<tr>
<td>$R_2$</td>
<td>(2, 2)</td>
<td>5.013E-3</td>
</tr>
<tr>
<td>$R_3$</td>
<td>(2, 2)</td>
<td>9.950E-1</td>
</tr>
<tr>
<td>$R_4$</td>
<td>(2, 2)</td>
<td>9.950E-1</td>
</tr>
<tr>
<td>-$R_2$</td>
<td>(2, 1)</td>
<td>1.000E+0</td>
</tr>
<tr>
<td>$R_1$</td>
<td>(1, 1)</td>
<td>1.000E+0</td>
</tr>
</tbody>
</table>

### Netlist

\[
\begin{align*}
\text{equations:} & \quad \begin{bmatrix} R_1 + R_2 & -R_2 \\ -R_2 & R_2 + R_3 + R_4 \end{bmatrix} \begin{bmatrix} i_1 \\ i_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\
\text{numeric result:} & \quad R_1 = R_2 = 10 \, \Omega \\
& \quad R_3 = R_4 = 1000 \, \Omega \\
& \quad \Rightarrow i_2 = 2.493 \cdot 10^{-4} \, \text{A}
\end{align*}
\]
Symbolic Approximation: Ranking

<table>
<thead>
<tr>
<th>symbol</th>
<th>position</th>
<th>relative error</th>
</tr>
</thead>
<tbody>
<tr>
<td>$-R_2$</td>
<td>(1, 2)</td>
<td>2.488E-3</td>
</tr>
<tr>
<td>$R_2$</td>
<td>(2, 2)</td>
<td>5.013E-3</td>
</tr>
<tr>
<td>$R_3$</td>
<td>(2, 2)</td>
<td>9.950E-1</td>
</tr>
<tr>
<td>$R_4$</td>
<td>(2, 2)</td>
<td>9.950E-1</td>
</tr>
<tr>
<td>$-R_2$</td>
<td>(2, 1)</td>
<td>1.000E+0</td>
</tr>
<tr>
<td>$R_1$</td>
<td>(1, 1)</td>
<td>1.000E+0</td>
</tr>
<tr>
<td>$R_2$</td>
<td>(1, 1)</td>
<td>1.000E+0</td>
</tr>
</tbody>
</table>

**netlist**

\[
\begin{bmatrix}
R_1 + R_2 & -R_2 \\
-R_2 & R_2 + R_3 + R_4
\end{bmatrix}
\begin{bmatrix}
i_1 \\
i_2
\end{bmatrix} =
\begin{bmatrix} 1 \\ 0 \end{bmatrix}
\]

**equations**

**numeric result**

\[R_1 = R_2 = 10 \, \Omega\]
\[R_3 = R_4 = 1000 \, \Omega\]
\[\Rightarrow i_2 = 2.493 \cdot 10^{-4} \, \text{A}\]

**cumulative error**

0.002488
### Symbolic Approximation: Ranking

<table>
<thead>
<tr>
<th>symbol</th>
<th>position</th>
<th>relative error</th>
</tr>
</thead>
<tbody>
<tr>
<td>-R₂</td>
<td>(1, 2)</td>
<td>2.488E-3</td>
</tr>
<tr>
<td>R₂</td>
<td>(2, 2)</td>
<td>5.013E-3</td>
</tr>
<tr>
<td>R₃</td>
<td>(2, 2)</td>
<td>9.950E-1</td>
</tr>
<tr>
<td>R₄</td>
<td>(2, 2)</td>
<td>9.950E-1</td>
</tr>
<tr>
<td>-R₂</td>
<td>(2, 1)</td>
<td>1.000E+0</td>
</tr>
<tr>
<td>R₁</td>
<td>(1, 1)</td>
<td>1.000E+0</td>
</tr>
<tr>
<td>R₂</td>
<td>(1, 1)</td>
<td>1.000E+0</td>
</tr>
</tbody>
</table>

**Netlist**

\[
\begin{align*}
R₁ & \quad R₂ \\
- & \quad -
\end{align*}
\]

**Equations**

\[
\begin{bmatrix}
R₁ + R₂ & -R₂ \\
-R₂ & -R₂ + R₃ + R₄
\end{bmatrix}
\begin{bmatrix}
i₁ \\
i₂
\end{bmatrix}
= \begin{bmatrix} 1 \\ 0 \end{bmatrix}
\]

**Numeric result**

\[
R₁ = R₂ = 10 \ \Omega \\
R₃ = R₄ = 1000 \ \Omega \\
\Rightarrow i₂ = 2.493 \cdot 10^{-4} \text{ A}
\]

**Cumulative error** 0.0025
## Symbolic Approximation: Ranking

<table>
<thead>
<tr>
<th>symbol</th>
<th>position</th>
<th>relative error</th>
</tr>
</thead>
<tbody>
<tr>
<td>-R₂</td>
<td>(1, 2)</td>
<td>2.488E-3</td>
</tr>
<tr>
<td>R₂</td>
<td>(2, 2)</td>
<td>5.013E-3</td>
</tr>
<tr>
<td>R₃</td>
<td>(2, 2)</td>
<td>9.950E-1</td>
</tr>
<tr>
<td>R₄</td>
<td>(2, 2)</td>
<td>9.950E-1</td>
</tr>
<tr>
<td>-R₂</td>
<td>(2, 1)</td>
<td>1.000E+0</td>
</tr>
<tr>
<td>R₁</td>
<td>(1, 1)</td>
<td>1.000E+0</td>
</tr>
<tr>
<td>R₂</td>
<td>(1, 1)</td>
<td>1.000E+0</td>
</tr>
</tbody>
</table>

**netlist**

\[
\begin{bmatrix}
R₁ + R₂ \\
- R₂ \\
- R₂ + R₃ + R₄
\end{bmatrix}
\begin{bmatrix}
i₁ \\
i₂
\end{bmatrix}
= \begin{bmatrix} 1 \\ 0 \end{bmatrix}
\]

**numeric result**

\[
\begin{align*}
R₁ &= R₂ = 10 \, \Omega \\
R₃ &= R₄ = 1000 \, \Omega \\
\Rightarrow i₂ &= 2.493 \cdot 10^{-4} \, \text{A}
\end{align*}
\]

**cumulative error** 1.005

**STOP**
Symbolic Approximation: Ranking

<table>
<thead>
<tr>
<th>symbol</th>
<th>position</th>
<th>relative error</th>
</tr>
</thead>
<tbody>
<tr>
<td>-R₂</td>
<td>(1, 2)</td>
<td>2.488E-3</td>
</tr>
<tr>
<td>R₂</td>
<td>(2, 2)</td>
<td>5.013E-3</td>
</tr>
<tr>
<td>R₃</td>
<td>(2, 2)</td>
<td>9.950E-1</td>
</tr>
<tr>
<td>R₄</td>
<td>(2, 2)</td>
<td>9.950E-1</td>
</tr>
<tr>
<td>-R₂</td>
<td>(2, 1)</td>
<td>1.000E+0</td>
</tr>
<tr>
<td>R₁</td>
<td>(1, 1)</td>
<td>1.000E+0</td>
</tr>
<tr>
<td>R₂</td>
<td>(1, 1)</td>
<td>1.000E+0</td>
</tr>
</tbody>
</table>

Netlist

\[
\begin{bmatrix}
R₁ + R₂ & -R₂ \\
-R₂ & -R₂ + R₃ + R₄
\end{bmatrix}
\begin{bmatrix}
i₁ \\
i₂
\end{bmatrix}
= \begin{bmatrix}1 \\
0
\end{bmatrix}
\]

Numeric result

\[R₁ = R₂ = 10 \, \Omega\]
\[R₃ = R₄ = 1000 \, \Omega\]
\[\Rightarrow i₂ = 2.493 \times 10^{-4} \, \text{A}\]

Cumulative error 0.0025
## Symbolic Approximation: Ranking

<table>
<thead>
<tr>
<th>symbol</th>
<th>position</th>
<th>relative error</th>
</tr>
</thead>
<tbody>
<tr>
<td>-$R_2$</td>
<td>(1, 2)</td>
<td>2.488E-3</td>
</tr>
<tr>
<td>$R_2$</td>
<td>(2, 2)</td>
<td>5.013E-3</td>
</tr>
<tr>
<td>$R_3$</td>
<td>(2, 2)</td>
<td>9.950E-1</td>
</tr>
<tr>
<td>$R_4$</td>
<td>(2, 2)</td>
<td>9.950E-1</td>
</tr>
<tr>
<td>-$R_2$</td>
<td>(2, 1)</td>
<td>1.000E+0</td>
</tr>
<tr>
<td>$R_1$</td>
<td>(1, 1)</td>
<td>1.000E+0</td>
</tr>
<tr>
<td>$R_2$</td>
<td>(1, 1)</td>
<td>1.000E+0</td>
</tr>
</tbody>
</table>

###netlist

\[
\begin{align*}
U_0 & \quad i_1 \quad R_2 \quad i_2 \quad R_3 \quad U_4 \\
\end{align*}
\]

###equations

\[
\begin{bmatrix}
R_1 + R_2 & -R_2 \\
-R_2 & -R_2 + R_3 + R_4 
\end{bmatrix}
\begin{bmatrix}
i_1 \\
i_2
\end{bmatrix} =
\begin{bmatrix}
1 \\
0
\end{bmatrix}
\]

###symbolic approximated transfer function

\[
i_2 = \frac{R_2 R_4}{(R_1 + R_2)(R_3 + R_4)}
\]
Overview Symbolic Simplification Methods

Design task

Linear equations
- AC, P/Z, Noise
- Simplified equations
- Transfer functions
- Insights

Nonlinear equations
- AC, DC, Transient, TEMP
- Simplified equations
- ABM generation
- Insights

Automatic error control
Linear Symbolic Analysis

Symbolic methods

- Starting from netlist description
- Standard analysis techniques (MNA, STA) for setting up analytic system equations
Linear Symbolic Analysis

Simplification methods

- **SBG** (Simplification Before Generation)
  - matrix approximation based on Sherman-Morrison formula for error prediction
- **SAG** (Simplification After Generation)
  - approximation of transfer function in canonical SOP form to equal coefficient error
- Pole/Zero techniques
Mathematical Formulation

- Small-signal behavior can mathematically be described by a linear system of equations:

\[
A(s; p) \cdot \hat{x} = b(s; p)
\]

\[
a_{ij} = \sum a_i(p) \cdot s^j
\]

- Formulated in Laplace frequency \( s \)
- Coefficients of \( A \)

- With:
  - Internal variables: \( \hat{x} \in \mathbb{R}^n \)
  - Parameters: \( p = (p_1, L, p_N) \)
  - Output component: \( y = \hat{x}_k \)
Linear Symbolic Analysis: Benefits and Limits

Benefits

• support for symbolic computation of transfer functions, etc.
• support for symbolic pole/zero analyses
• application of industrial-sized circuits (up to 100 transistors) in interactive computation time
• advanced research area: efficient and sophisticated algorithms available

Limits

• symbolic analysis is a tradeoff between accuracy and complexity
• symbolic extraction of poles and zeros possible for transfer functions of order $\leq 3$ only
Nonlinear Symbolic Analysis

library disciplines;
use disciplines.electromagnetic_system;
library ieee;
use ieee.math_real.all;

entity sqrtblock is
generic (
  \LB\ : real := 0.1e-3;
  \ISSQ1\ : real := 1.0e-16;
  \ISSQ2\ : real := 1.0e-16;
);

AHDL model

netlist description

reference simulation

 original DAE system

original error control

simplified DAE system

simplified index control

library disciplines;
use disciplines.electromagnetic_system;
library ieee;
use ieee.math_real.all;

entity sqrtblock is
generic (
  \LB\ : real := 0.1e-3;
  \ISSQ1\ : real := 1.0e-16;
  \ISSQ2\ : real := 1.0e-16;
);

AHDL model

netlist description

reference simulation

 original DAE system

original error control

simplified DAE system

simplified index control

library disciplines;
use disciplines.electromagnetic_system;
library ieee;
use ieee.math_real.all;

entity sqrtblock is
generic (
  \LB\ : real := 0.1e-3;
  \ISSQ1\ : real := 1.0e-16;
  \ISSQ2\ : real := 1.0e-16;
);

AHDL model

netlist description

reference simulation

 original DAE system

original error control

simplified DAE system

simplified index control

library disciplines;
use disciplines.electromagnetic_system;
library ieee;
use ieee.math_real.all;

entity sqrtblock is
generic (
  \LB\ : real := 0.1e-3;
  \ISSQ1\ : real := 1.0e-16;
  \ISSQ2\ : real := 1.0e-16;
);

AHDL model

netlist description

reference simulation

 original DAE system

original error control

simplified DAE system

simplified index control

library disciplines;
use disciplines.electromagnetic_system;
library ieee;
use ieee.math_real.all;

entity sqrtblock is
generic (
  \LB\ : real := 0.1e-3;
  \ISSQ1\ : real := 1.0e-16;
  \ISSQ2\ : real := 1.0e-16;
);

AHDL model

netlist description

reference simulation

 original DAE system

original error control

simplified DAE system

simplified index control
Mathematical Formulation

- System behavior can mathematically be described by a set of differential algebraic equations (DAE system):

$$F(f, g)$$

- With:

  - differential part
  - algebraic part

- inputs
  - internal variables
  - outputs
  - parameters

$$f(x(t), \xi(t), y(t), u(t); p) = 0$$
$$g(x(t), y(t), u(t); p) = 0$$

$$u = (u_1, \ldots, u_m)$$
$$x = (x_1, \ldots, x_n)$$
$$y = (y_1, \ldots, y_f)$$
$$p = (p_1, \ldots, p_N)$$
Numerical Analyses

- **Transient analysis**
  - dynamic behavior
  - nonlinear DAE system

- **DC/DТ analysis**
  - static behavior
  - nonlinear equation system

- **AC analysis**
  - linear behavior
  - linear equation system in frequency domain

![Diagram showing transient analysis results](image1)

![Diagram showing DC/DT analysis results](image2)

![Diagram showing AC analysis results](image3)
Symbolic Model Reduction

- **Specifications**
  - inputs $u$, outputs $v$
  - numerical analyses $A$

- **Error control**
  - error function $E$
  - error bound $\varepsilon$

- **Goal**
  - find DAE system $G$ with reduced complexity and defined accuracy

\[ v_F = A(F, u) \quad \text{and} \quad v_G = A(G, u) \]

\[ E(v_F, v_G) < \varepsilon \]
Symbolic Model Reduction

- **Specifications**
  - inputs $u$, outputs $v$
  - numerical analyses $A$

- **Error control**
  - error function $E$
  - error bound $\varepsilon$

- **Goal**
  - find DAE system $G$ with reduced complexity and defined accuracy

- **Simplification process**
  - iterative application of reduction techniques $R$
Symbolic Reduction Techniques (1)

- **Algebraic manipulation**
  - elimination of variables
  - removal of independent blocks of equations

\[
\begin{align*}
  x &= f(y) \\
  0 &= g(x, y)
\end{align*}
\]

\[
\begin{align*}
  0 &= g(f(y), y)
\end{align*}
\]
Symbolic Reduction Techniques (2)

Branch reduction

detection and removal of unused branches of piecewise-defined functions

\[ f(x) = \begin{cases} f_1(x) & x < a \\ f_2(x) & a \leq x \leq b \\ f_3(x) & x > b \end{cases} \]

\[ f(x) = f_2(x) \quad \text{für alle } x \]
Symbolic Reduction Techniques (3)

\[ F_j : \sum_{i=1}^{N} t_i(x) = 0 \]

\[ G_j : \sum_{i=1, i \neq k}^{N} t_i(\tilde{x}) = 0 \]

- **Term reduction**
  - replace terms of equations with zero
  - applicable to nested expressions

\[-\text{AREA}$D1 \left(-1 + e^{38.6635 (V1[t] - V2[t])}\right) \text{IS}$D1 + \text{I}$AC$D1[t] - \text{GMIN}(V1[t] - V2[t]) = 0\]

\[-\text{AREA}$D1 \left(-1 + e^{38.6635 (V1[t] - V2[t])}\right) \text{IS}$D1 + \text{I}$AC$D1[t] = 0\]
Symbolic Reduction Techniques (4)

\[ F_j : \sum_{i=1}^{N} t_i(x) = 0 \]

\[ G_j : \sum_{i=1}^{N} t_i(\bar{x}) + \kappa = 0 \]

- **Term substitution**
  - replace terms of equations with constant value
  - applicable to nested expressions

\[ -\text{AREA}D1 \left( -1 + e^{38.6635 (V$1[t] - V$2[t])} \right) \text{ IS}D1 + I$AC$D1[t] - \text{GMIN} [V$1[t] - V$2[t]] = 0 \]

\[ -\text{AREA}D1 \left( -1 + e^{38.6635 (V$1[t] - V$2[t])} \right) \text{ IS}D1 + I$AC$D1[t] - \text{GMIN} [12 - V$2[t]] = 0 \]

Fraunhofer Institut Techno- und Wirtschaftsmathematik
Algorithm

- DAE system
  - perform reductions
    - error analysis
      - error ok?
        - yes: reference simulation
        - no: reject
      - simplified DAE system

Ranking

**Definition**

*Ranking* is the estimation of the influence of each reduction on the output behavior and a suitable ordering.

**Advantages**

- increased number of reductions
- decreased computation time

**Implementation**

- accurate estimation
- simple and efficient computation
- tradeoff between accuracy and efficiency
Clustering

- **Definition**
  Clustering is the bundling of reduction steps with similar influence on the output behavior.

- **Advantages**
  - many simultaneous reduction steps
  - reduced effort for error analyses
  - highly decreased computation time

- **Implementation**
  - ranking data can be used for clustering
Index Monitor

- **Definition**
  Index \( k \) of a DAE system is a measure for its *distance* to a regular ODE system. For \( k > 1 \) the numerical solving is an ill-posed problem.

- **Advantages**
  - monitoring of the index during model reduction
  - assuring numerical stability of reduced system

- **Implementation**
  - many index concepts not appropriate: algebraic/symbolic due to complexity, structural/graph-based due to modified topology
  - numerical computation of tractability index and strangeness index
Original Algorithm

1. DAE system
2. perform reductions
3. error analysis
   - no
   - yes
5. error ok?
6. simplified DAE system
7. reject
8. reference simulation

Flowchart:
- DAE system
- perform reductions
- error analysis
- error ok?
- simplified DAE system
- reject
- reference simulation
Improved Algorithm

1. DAE system
2. Compute ranking
3. (Re)compute clustering
4. Perform reductions
5. Error analysis, index analysis
6. Error ok? Index ok?
7. Yes: Simplified DAE system
   - No: Reject
     - Reference simulation
     - (Re)compute clustering
     - Compute ranking

Error analysis, index analysis: Yes, error ok? Index ok? Yes: Simplified DAE system
Example Application: Operational Amplifier

- 8 bipolar transistors
- 7 sources
- 5 resistors
- 1 capacitor

■ Task
- computation of a behavioral model
- input: \( V_{\text{sig}} \), output: \( V_9 \)
Example Application: Operational Amplifier

■ DT specification
  ■ error bound \( \varepsilon_{DT} = 0.25 \text{V} \)
  ■ error function \( E_{DT} = \| \cdot \|_{\infty} \)

■ Transient specification
  ■ error bound \( \varepsilon_{Tran} = 1.0 \text{V} \)
  ■ error function \( E_{Tran} = \| \cdot \|_{\infty} \)
Example Application: Operational Amplifier

- **Original system**
  - 73 equations
  - 350 terms
  - 94 parameters
Example Application: Operational Amplifier

Symbolic reduction
- 73 → 6 equations
- 350 → 24 terms
- 94 → 21 parameters

Model reduction time
- 792 s
Example Application: Operational Amplifier

- **Symbolic reduction**
  - 73 → 6 equations
  - 350 → 24 terms
  - 94 → 21 parameters

- **Behavioral model**
  - original: 166.0 s
  - model: 2.3 s
Infineon Design Problem: Folded-cascode OpAmp

<table>
<thead>
<tr>
<th>cfcamp</th>
<th>equations</th>
<th>top-level terms</th>
<th>parameters</th>
<th>factor</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>original model</strong></td>
<td>1120</td>
<td>16306</td>
<td>1977</td>
<td>1</td>
</tr>
<tr>
<td><strong>simplified model</strong></td>
<td>22</td>
<td>54</td>
<td>94</td>
<td>137</td>
</tr>
</tbody>
</table>

- **Application**
  - automatic generation of behavioral models (e.g. VHDL-AMS, Verilog-A)
  - ⇒ system simulation, optimization, system understanding
Nonlinear Symbolic Analysis: Benefits and Limits

**Benefits**
- support for automated generation of behavioral models for different behavioral modeling languages and simulators
- automatic and error-controlled complexity reduction
- supported analysis modes: DC-Transfer, AC, Transient

**Limits**
- limited circuit size (currently up to 20 transistors)
- algorithms under development
- explicit symbolic results for static systems in general not possible

**Synergies**
- due to the general mathematical concept the methods are also applicable to other domains (e.g. system simulation of mechatronical systems)
EDA Tool Analog Insydes

Tool for analysis, modeling, and optimization of analog circuits

Commercially distributed since 1998

Current version: 2.1.1

Mathematica Add-on

Evaluation version available at: www.analog-insydes.de
EDA Tool Analog Insydes

Mathematica Kernel

High-performance MathLink Binaries

Online Help Mathematica Frontend

Interfaces: PSpice, Eldo, Saber, Spectre, Titan

Fraunhofer Institute Techno- und Wirtschaftsmathematik
MOR Techniques – Current Activities

- **bottom-up behavioral modeling**
- **device model compilation**
- **coupling of numeric/symbolic MOR**

**Analog Insydes**

- **Netlist Import**
  - **Setup of Circuit Equations**
  - **Model Approximation**
  - **Performance Optimization**
  - **Model Export**

- **Circuit Schematic**
- **Behavioral Model AHDL**
- **AHDL Import**

**Involved in public project initiatives:**
- VeronA
- SymTecO
- SyreNe
Transfer of Methodology

Enzyme kinetics
interface between
Analog Insydes and
PathwayLab (simulator for enzyme kinetics)

Hydraulics
modeling of
gas pipeline networks

Energy distribution networks
model reduction for high-voltage energy distribution networks with dispersed power generation
Transfer of Methodology: Mechatronical Systems

Usage of analogies for methodology transfer

<table>
<thead>
<tr>
<th>analogies</th>
<th>electronics</th>
<th>mechanics</th>
</tr>
</thead>
<tbody>
<tr>
<td>through variables</td>
<td>current</td>
<td>force, torque</td>
</tr>
<tr>
<td>across variables</td>
<td>voltage</td>
<td>rotation, displacement</td>
</tr>
<tr>
<td>equation setup</td>
<td>Kirchhoff laws</td>
<td>d’Alemberts principle</td>
</tr>
</tbody>
</table>

Adding support for mechanics

- Decompose mechanical system in finite elements
- Implement elements in terms of symbolic component models
- Support for vector-valued variables
- Automated mapping to an equivalent netlist with scalar-type nodes and branches

Fraunhofer Institut Techno- und Wirtschaftsmathematik
Application Example: Symbolic Analysis of Mechanical System

Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>density</td>
<td>$\rho$</td>
</tr>
<tr>
<td>elasticity</td>
<td>$E$</td>
</tr>
<tr>
<td>shear modulus</td>
<td>$G$</td>
</tr>
<tr>
<td>length A1</td>
<td>$l_{A1}$</td>
</tr>
<tr>
<td>length A2</td>
<td>$l_{A2}$</td>
</tr>
<tr>
<td>height A1, A2</td>
<td>$h_A$</td>
</tr>
<tr>
<td>length B</td>
<td>$l_B$</td>
</tr>
<tr>
<td>height B</td>
<td>$h_B$</td>
</tr>
<tr>
<td>width A1, A2, B</td>
<td>$w$</td>
</tr>
</tbody>
</table>

Goal:

Find symbolic formula for displacement as a function of force

System consists of silicon beams:

- $l_{A2} = 12 \, \mu m$
- $l_B = 250 \, \mu m$
- $l_{A1} = 223.5 \, \mu m$
Application Example: Symbolic Analysis of Mechanical System

**Goal:**
Find symbolic formula for displacement as a function of force

**Step 1:**
Analyze the system numerically in order to find out a suitable discretisation. Therefore we compare the dynamics with different discretisations.

**Step 2:**
Derive the symbolic model equations for the static model from dynamic model.

**Step 3:**
Apply symbolic model reduction in order to reduce the static model and solve the reduced symbolic model equations for displacement in C3.

**Step 4:**
Validate the derived solution against the numerical solution of the full model.

Fraunhofer Institut Techno- und Wirtschaftsmathematik
Result Validation

\[ u = F \frac{\frac{l^3}{2l_{A1}}}{Eh^3 w} \frac{l_{A1} + 2l_{A2}}{2l_{A1} + l_{A2}} \]

Compare **reduced** against **full model**:
Result Validation

\[ u = F \frac{l_{A1}^3}{Eh_{A1}^3} \frac{l_{A1} + 2l_{A2}}{2l_{A1} + l_{A2}} \]

Compare reduced against full model:
Result Validation

\[ u = F \frac{l_{A1}^3}{Eh_A^3w} \frac{l_{A1} + 2l_{A2}}{2l_{A1} + l_{A2}} \]

Compare **reduced** against **full model**:
Application Example: Acceleration Sensor

Simple mechatronical system
- Acceleration sensor with electronic circuit attached to a point mass
- Force $F$ accelerates the system

Acceleration sensor component
- Three metal plates forming series connection of two capacities
- Movable center plate with mass $m_{\text{dyn}}$ and Hooks law
  In case of acceleration
  $\rightarrow$ Plate moves
  $\rightarrow$ Change of capacities
  $\rightarrow$ Voltage drop at $V_{\text{out}}$
Application Example: Model Reduction

Analog Insydes command `CancelTerms` approximates the DAE system:
Application Example: Model Reduction

Analog Insydes command **CancelTerms** approximates the DAE system
Application Example: Reduced Model Equations

Post processing with Analog Insydes command `CompressDAE`:

Reduced equation system
5 equations

Comparison of simulation results
original model
reduced model
## Summary and Outlook

**Summary**
- Application of symbolic analysis only possible using approximation techniques coupled with numeric simulation
- Linear symbolic analysis used successfully for industrial circuit analysis tasks
- Nonlinear symbolic analysis under development with promising results

**Outlook**
- Extension of nonlinear simplification methods
- Extension towards support for multi-physics systems
- Integration of symbolic analysis into industrial design flows