Modeling, Simulation and Deployment

Dr. Michael Tiller
Xogeny
Modeling
Modeling
Modeling
Modeling
Modeling
Modeling
Modeling
Am I at the wrong talk?
Modeling and “the V”
Modelica

» Modelica is a modeling language that is:
  > Vendor-neutral
  > Multi-domain
  > Object-oriented
  > Multi-formalism

» Modelica is like LEGOs for building mathematical system models
Modelica
Modelica
Modelica
Acausal Modeling
Flexibility

- Only 1 state (vs. 2 previously)
- No feedback loop
- Completely different model!
Block Diagrams

» Textbook equations have to be constantly reformulated depending on context.

» Different “blocks” with different combinations of inputs and outputs.

» Tedious, time-consuming and error prone.

» Long-division

\[
\begin{array}{c|c|c|c}
\text{1x7} & 7 & \underline{7} & 0.1428 \\
\text{4x7=28} & 28 & -28 & 0.00000 \\
\text{2x7=14} & 14 & -14 & 0.00000 \\
\text{8x7=56} & 56 & -56 & 0.00000 \\
\end{array}
\]

Acausal Modeling

» Textbook equations are captured in reusable object-oriented component models.

» A single component for all causalities (e.g. planetary gear).

» Fun, fast and automated (and efficient!)

» Calculator
DAEs

- Natural way to describe physical behavior
  - Multi-body systems (joint constraints)
  - Fluid problems (ideal gas law)
  - Easy to express many important idealizations (stiff springs)

- Difficult to solve in a purely numerical way
  - Consistent initial conditions
  - High index DAEs

- Preferred solution methods:
  - Index reduction (Pantelides’ algorithm)
  - Dummy derivative method
  - Turn DAE into ODEs (or index-1 DAEs)
Attraction of modeling is the landscape of infinite possibilities it creates.
Simulation
In the beginning...

» What were computers invented for?

ENIAC (circa 1947-1955)

“The Giant Brain”

Artillery Firing Tables

» Simulation is as old as computing itself.
Solution Method

» Originally, solution schemes (integration) was integrated with problem:

\[
V = V + A*\Delta t; \\
X = X + V*\Delta t;
\]

» Eventually, problem and solver were cleanly partitioned:

\[
\dot{x}(t) = f(x, u, t); \\
x_{n+1} = x_n + hf(t_n + \frac{h}{2}, x_n + \frac{hf(t_n, y_n)}{2})
\]

» Performance:

> (Cost of evaluation \(f\)) \times (# of times \(f\) is evaluated)
Evaluation Costs

» The Six Blind Men and The Elephant
Don’t think of $f$ as a black-box numerical function, think of it as a representation of your system that conveys a complete representation of your problem and then optimize it.
Symbolic Manipulation

» Umbrella topic for:
  > Equation sorting
  > Index reduction
  > State selection
  > Substitutions
  > Tearing

» Goal is not a symbolic/analytical solution

» Reduces the DAEs down to ODEs
  > More natural way to express behavior
  > Reuse established numerical solvers
  > Heavily optimize evaluation costs

» Opinion: it will be impossible for purely numerical tools to compete.

Requires structural information
Generating Equations

\begin{align*}
\text{step}.n.v &= \text{resistor}.n.v \\
\text{resistor}.n.v &= \text{inductor}.n.v \\
\text{inductor}.n.v &= \text{capacitor}.n.v \\
\text{capacitor}.n.v &= \text{ground}.n.v \\
\text{step}.n.i + \text{resistor}.n.i &+ \text{inductor}.n.i + \text{capacitor}.n.i + \text{ground}.n.i = 0 \\
\text{step}.p.v &= \text{resistor}.p.v \\
\text{resistor}.p.v &= \text{inductor}.p.v \\
\text{inductor}.p.v &= \text{capacitor}.p.v \\
\text{step}.p.i + \text{resistor}.p.i &+ \text{inductor}.p.i + \text{capacitor}.p.i = 0 \\
\text{step}.p.i &+ \text{step}.n.i = 0 \\
\text{step}.p.i &= f(t) \\
\text{resistor}.p.i &+ \text{resistor}.n.i = 0 \\
\text{resistor}.p.i &* \text{resistor}.R = \text{resistor}.p.v - \text{resistor}.n.v \\
\text{inductor}.p.i &+ \text{inductor}.n.i = 0 \\
\text{der}(\text{inductor}.p.i) &* \text{inductor}.L = \text{inductor}.p.v - \text{inductor}.n.v \\
\text{capacitor}.p.i &+ \text{capacitor}.n.i = 0 \\
\text{capacitor}.p.i &= \text{capacitor}.C * \text{der}(\text{capacitor}.p.v) - \text{der}(\text{capacitor}.n.v) \\
\text{ground}.n.v &= 0
\end{align*}
Equation Structure

\[ \text{step.n.v} = \text{resistor.n.v} \]
\[ \text{resistor.n.v} = \text{inductor.n.v} \]
\[ \text{inductor.n.v} = \text{capacitor.n.v} \]
\[ \text{capacitor.n.v} = \text{ground.n.v} \]
\[ \text{step.n.i} + \text{resistor.n.i} + \text{inductor.n.i} + \text{capacitor.n.i} + \text{ground.n.i} = 0 \]
\[ \text{step.p.v} = \text{resistor.p.v} \]
\[ \text{resistor.p.v} = \text{inductor.p.v} \]
\[ \text{inductor.p.v} = \text{capacitor.p.v} \]
\[ \text{step.p.i} + \text{resistor.p.i} + \text{inductor.p.i} + \text{capacitor.p.i} = 0 \]
\[ \text{step.p.i} + \text{step.n.i} = 0 \]
\[ \text{step.p.i} = f(t) \]
\[ \text{resistor.p.i} + \text{resistor.n.i} = 0 \]
\[ \text{resistor.p.i} \times \text{resistor.R} = \text{resistor.p.v} - \text{resistor.n.v} \]
\[ \text{inductor.p.i} + \text{inductor.n.i} = 0 \]
\[ \text{der(inductor.p.i)} \times \text{inductor.L} = \text{inductor.p.v} - \text{inductor.n.v} \]
\[ \text{capacitor.p.i} + \text{capacitor.n.i} = 0 \]
\[ \text{capacitor.p.i} \times \text{capacitor.C} = \text{der(capacitor.p.v)} - \text{der(capacitor.n.v)} \]
\[ \text{ground.n.v} = 0 \]
\[
\begin{cases}
ground.n.v \\
capacitor.n.v \\
inductor.n.v \\
resistor.n.v \\
step.n.v \\
step.p.i \\
step.n.i \\
dr(inductor.p.i) \\
inductor.n.i \\
resistor.p.i \\
resistor.n.i \\
capacitor.p.i \\
capacitor.n.i \\
dr(capacitor.p.v) \\
ground.n.i \\
\end{cases}
= \begin{cases}
0 \\
ground.n.v \\
capacitor.n.v \\
inductor.n.v \\
resistor.n.v \\
f(t) \\
-\text{step.p.i} \\
-\text{step.n.i} \\
f(t) \\
-\text{inductor.p.v} \\
-\text{resistor.p.v} \\
-\text{inductor.p.v} \\
-\text{resistor.p.v} \\
\end{cases}
\frac{(\text{inductor.p.i} - \text{inductor.n.v})}{\text{inductor.L}} \\
-\text{inductor.p.i} \\
\frac{(\text{resistor.p.v} - \text{resistor.n.v})}{\text{resistor.R}} \\
-\text{resistor.p.i} \\
-\text{step.p.i} - \text{resistor.p.i} - \text{inductor.p.i} \\
-\text{capacitor.p.i} \\
\frac{\text{capacitor.p.i}}{\text{capacitor.C}} \\
-\text{step.n.i} - \text{resistor.n.i} - \text{inductor.n.i} - \text{capacitor.n.i}
\end{cases}
\]

\[
\begin{bmatrix}
1 \\
-1 & 1 \\
-1 & 1 \\
1 & 1 \\
-1 & 1 \\
-1 & 1 \\
1 & 1 \\
\end{bmatrix}
\]
Functional Mockup Interface (FMI)
Functional Mockup Interface (FMI)

Initiated by Daimler as an industry wide and vendor neutral alternative to (MathWorks) S-functions.

Initially funded by EU. Eventually became an official Modelica Association project.

A way to exchange compiled models

- Not really a modeling technology
- Limited “composability”

FMI 2.0-RC1 just released

- Support for discrete states
- Algebraic loops during events and initialization
» **Functional Mockup Unit – FMU**

» **Pre-compiled collection of files:**
  > Binaries (for various platforms)
  > Source code (in provided)
  > Resources (data files, etc)
  > Documentation
  > Model Description (XML file)

» **FMUs are instantiated**
  > Potentially multiple instances in same simulation
  > Can be formulated for “Model Exchange” or “Co-simulation”
Architectural Shifts

» Reaching limitations of classic Von Neumann architecture (CPU+memory).

» Multi-core machines and cloud computer resources are becoming increasingly common.

» Simulation is typically heavily sequential

» Exploiting future computing resources:
  > Thinking more about parallel computations
  > Loosely coupled analyses
  > Cheaper computing and timing driving analyses like optimization, monte-carlo and other parallelizable types of analysis
  > Model reduction could become more cost effective
» Xogeny proprietary platform
» Platform for running FMI compliant models “in the cloud”
» Inspiration came from dynamic programming application where desktop resources weren’t sufficient.
» Easy path to model reduction, Monte-Carlo analysis, etc.
» Wraps analyses in data management framework for persisting input and output data.
Deployment
Attraction of modeling is the landscape of infinite possibilities it creates.
Attraction of modeling is the landscape of infinite possibilities it creates.

“...characterized by anxiety in situations where the sufferer perceives certain environments as dangerous or uncomfortable, often due to the environment’s vast openness...”
Infinite Possibilities
It is worth the effort to organize features and capabilities around achieving a great user experience.
Important to understand the business questions that need answering.

- Do you need one application or many?

Models are the “functions” to capture non-trivial relationships.

- Can be recombined in different ways depending on the use case.

Applications need to provide a clear path from models to questions/solutions.

Software architectures often tend toward monolithic applications.

Xogeny applications heavily leverage declarative representations and code generation.

- Don’t write applications, write programs that write applications.
Web-Based Analyses
Interactivity
Turnkey Systems
### Stack Geometry

<table>
<thead>
<tr>
<th>Heat Exchanger</th>
<th>X</th>
<th>Y</th>
<th>Width</th>
<th>Height</th>
</tr>
</thead>
<tbody>
<tr>
<td>Charge Air</td>
<td>0</td>
<td>0</td>
<td>700 mm</td>
<td>500 mm</td>
</tr>
<tr>
<td>Transmission Oil</td>
<td>0</td>
<td>0</td>
<td>595 mm</td>
<td>555 mm</td>
</tr>
<tr>
<td>Condensor</td>
<td>0</td>
<td>0</td>
<td>330 mm</td>
<td>200 mm</td>
</tr>
<tr>
<td>Radiator</td>
<td>0</td>
<td>600</td>
<td>400 mm</td>
<td>300 mm</td>
</tr>
</tbody>
</table>
## Vehicle Data

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Driveline Ratio</td>
<td>4.1</td>
</tr>
<tr>
<td>Final gear ratio rear</td>
<td></td>
</tr>
<tr>
<td>Vehicle Body Mass</td>
<td>16000</td>
</tr>
<tr>
<td>kg</td>
<td>Maximum value is 5000</td>
</tr>
<tr>
<td>Front Tire Radius</td>
<td>0.3</td>
</tr>
<tr>
<td>m</td>
<td>Undeformed radius - front wheel</td>
</tr>
<tr>
<td>Front Tire Inertia</td>
<td>1</td>
</tr>
<tr>
<td>kg.m²</td>
<td>Inertia about the spin axis - front wheel</td>
</tr>
<tr>
<td>Rear Tire Radius</td>
<td>0.3</td>
</tr>
<tr>
<td>m</td>
<td>Undeformed radius - rear wheel</td>
</tr>
<tr>
<td>Rear Tire Inertia</td>
<td>1</td>
</tr>
<tr>
<td>kg.m²</td>
<td>Inertia about the spin axis - rear wheel</td>
</tr>
<tr>
<td>Drag Coefficient</td>
<td>0.38</td>
</tr>
<tr>
<td>Aerodynamic drag coefficient of car body</td>
<td></td>
</tr>
<tr>
<td>Frontal Area</td>
<td>2.7</td>
</tr>
<tr>
<td>m²</td>
<td>Frontal area of car body</td>
</tr>
<tr>
<td>Road Inclination</td>
<td>0</td>
</tr>
<tr>
<td>% Grade</td>
<td>Inclination in the x direction</td>
</tr>
<tr>
<td>Ambient Temperature</td>
<td>311</td>
</tr>
<tr>
<td>K</td>
<td>Ambient temperature (inlet to the stack)</td>
</tr>
</tbody>
</table>

## Scale Factors

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Drive Cycle Scaling</td>
<td>1</td>
</tr>
<tr>
<td>Engine Torque Scaling</td>
<td>1</td>
</tr>
<tr>
<td>Scaling factor for the drivecycle velocity</td>
<td></td>
</tr>
<tr>
<td>Scaling of output torque &gt;1 increases output torque</td>
<td></td>
</tr>
</tbody>
</table>
Vehicle Thermal Management

Drive Cycle

Drive Cycle

Statistics

Cycle Duration
50.0 seconds

Fuel Economy
17.69 MPG

Peak Speed
20.2 MPH
Conclusions

» Increasing pressure to connect CAD, CAE, requirements and system simulation.

» Lots of compelling technologies out there that are not being leveraged.

  > Competitive advantage in breaking away from legacy and capitalizing on these opportunities.

» Highlight the value of modeling by making it accessible to everybody.

» Exciting time for system simulation...
What Will You Build Today?
Questions?