MultiBody Simulation with OpenModelica and MathModelica

Gerhard Schmitz, Peter Aronsson, Adrian Pop

2010-02-08
2nd OpenModelica Workshop
Linköping, Sweden

www.OpenModelica.org
Own experiences with libraries

Gerhard Schmitz, Applied Thermodynamics
Institute of Thermo-Fluid Dynamics
Hamburg-Harburg, University of Technology

- **AClib** (precursor of AirConditioning®, Torge Pfafferott)
- Contributions to **Modelica_Fluid** and **AirConditioning** (Katrin Proelss)
- **ACClib** (ECS of Aircrafts, Karin Dietl, Jens Vasel, Philipp Jordan)
- **KATO** (CabinFlow in Aircrafts, Henning Knigge)
- **Thermal Separation Library** (Karin Dietl, Andreas Joos)
- **DIM** (Water vapor transport, Andreas Joos, Stefan Wischhusen, XRG)
- Energy Nets (Stefan Storace, Lichtblick)
- Battery cooling systems (based on AirConditioning, Imke Krüger)
- Building HVAC library (Jan Wrobel, Wilson Casas)
Demands and users

Reasons for using Modelica libraries

- Developing of new systems (pre development)
- Modefication and optimization of exiting systems
- Understanding the dynamic behavior of complex systems

Different users:

- Drag and drop users
- Modelling users
- Library developers
## Library issues – level of importance

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>User</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Graphical User interface</td>
<td>0</td>
<td>++</td>
<td>++</td>
<td>+</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Numerical Stability in diff. environments</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
</tr>
<tr>
<td>Initialisation procedures</td>
<td>--</td>
<td>0</td>
<td>0</td>
<td>++</td>
<td>++</td>
<td>+</td>
</tr>
<tr>
<td>Steady state solutions</td>
<td>--</td>
<td>-</td>
<td>-</td>
<td>++</td>
<td>++</td>
<td>+</td>
</tr>
<tr>
<td><strong>Developer</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Structure</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>++</td>
<td>++</td>
<td>++</td>
</tr>
<tr>
<td>Modelica Conformism</td>
<td>++</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Debugging Tools</td>
<td>0</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
</tr>
</tbody>
</table>

++ very important, + important, o nice to have, - not important, - - needless
Most important: error messages (1)

class HelloWorld
    Real x(start = 1);
    parameter Real a = 1;
    equation
        der(x) = - a * x;
end HelloWorld;

>> simulate(HelloWorld)
record SimulationResult
    resultFile = "HelloWorld_res.plt"
end SimulationResult;

class HelloWorld
    Real x(start = 1);
    parameter Real a = 1;
    equation
        der(x) = - a * x / 0;
end HelloWorld;

>> simulate(HelloWorld)
record SimulationResult
    resultFile = "Simulation failed."
end SimulationResult;

Other Modelica Simulation Environments:
The following error was detected at time: 0

Model error-division by zero: (a*x) / (0) = (1) / (0)
Most important: error messages (2)

```model Pendel
  Real theta(start=2);
  Real omega;
  parameter Real L=2;
  constant Real g=9.81;
  equation
    der(theta) = omega;
    der(omega) = -(g/L)*(theta);
end Pendel;
```

```model Pendel
  Real theta(start=2);
  Real omega;
  parameter Real L=2;
  constant Real g=9.81;
  equation
    der(theta) = omega;
    der(omega) = -(g/0)*(theta);
end Pendel;
```

>> simulate(Pendel, stopTime=20)
record SimulationResult
resultFile = "Simulation failed. Error: Division by zero in 9.81 / 0.0"
end SimulationResult;
```
Conclusion: Priority of libraries

1. Modelica Standard Library
2. Multi Body Library
3. Modelica_Media
4. Modelica_Fluid
5. Others

But for OpenModelica most important:
→ Plausible error messages
MultiBody simulation in MathModelica

Peter Aronsson, peter.aronsson@mathcore.com

www.mathcore.com
info@mathcore.com
+46 13 32 85 00
Old Multibody library

• MathModelica currently support old version of MultiBody library:
  – Visualization of 3D mechanics
  – Kinematic loops require special consideration in user models
  – Complicated connection rules
  – Non-standardized operator constrain() used.
MathModelica Multibody visualization
New Multibody library

• Main features:
  – About 60 main components (joints, force, part, body, etc)
  – About 75 functions to operate on Orientation object/coordinate systems
  – Builtin animation properties of all components
  – Automatic handling of kinematic loops (using new language constructs: Over constrained connection sets)
  – Automatic state selection using the stateSelect attribute.
  – Analytic solution of a class of special mechanical constructs that leads to nonlinear equation systems.
Requirements on backend of a Modelica translator

• For efficient simulation equation tearing is required. (already available in MathModelica)

• New MultiBody library heavily relies on Orientation record and functions operating on it. This requires
  – Function inlining (using annotation(inline))
    • Both req. before and after index reduction
  – Derivative information (required for e.g. index reduction)
  – Record constructor expansion & simplification
Requirements cont’

• Dynamic state selection
  – Mostly needed for e.g. "PointGravity" models.

• Analytic solution of certain nonlinear equations (useful for real time simulations)
Issues in OpenModelica frontend and MathModelica backend

• Multibody introduces lookup of functions through instances
  – Required changes to frontend for handling such lookup
  – Redesign of Modelica function code generation for backend also required.

• Index reduction with differentiation of vector & matrix expressions (was previously not implemented)
Status

• MathModelica backend will soon be able to simulate new MultiBody models
  – Much functionality already in place
    • Tearing
    • Common subexpression elimination
    • Analytic jacobians for nonlinear systems
  – Remaining issues:
    • Index reduction with simplification of vector/matrix expressions including record constructors.
Next MathModelica release

• Support for new MultiBody library
• 3D visualization with custom DXF CAD objects
• Efficient Multibody simulations
• Improved sensitivity analysis
• And much more…
Technical Issues solved in the OpenModelica Compiler to support flattening of the MultiBody library

Adrian Pop

2010-02-08

Open Source Modelica Consortium
Programming Environment Laboratory
Department of Computer and Information Science
Linköping University
Issues solved to support the MultiBody library

**General Modelica issues**
- array aliases (100%) (Percentage of implementation work)
- enumerations (95%)
- inner outer with modifications on inner (95%)

**MultiBody specific**
- calling functions via component i.e. `world.gravityAcceleration` (100%)
- constraint types (100%)
- breaking of over constrained connection graph (90%)
- performance issues (40%)
- expandable connectors (90%)
Thank You!
Questions?

OpenModelica Project
http://www.OpenModelica.org