Split-system method for simulating cyberphysical systems

James Nutaro
Oak Ridge National Laboratory

Follow this and additional works at: http://dc.engconfintl.org/power_grid

Part of the Electrical and Computer Engineering Commons

Recommended Citation

This Conference Proceeding is brought to you for free and open access by the Proceedings at ECI Digital Archives. It has been accepted for inclusion in Modeling, Simulation, And Optimization for the 21st Century Electric Power Grid by an authorized administrator of ECI Digital Archives. For more information, please contact franco@bepress.com.
Split-system method for simulating cyber-physical systems

James Nutaro, PhD
Oak Ridge National Laboratory
nutarolj@ornl.gov
Outline

- Why?
- Getting it wrong: co-simulation
- Fixing things: the split system method
- Examples
- Tools
- Economics
## Hybrid Systems in the Power Grid

<table>
<thead>
<tr>
<th>System Specification</th>
<th>AC/DC circuit models</th>
<th>Digital controllers</th>
<th>Communications</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>↓</td>
<td>↓</td>
</tr>
<tr>
<td>System Specification</td>
<td></td>
<td>↓</td>
<td>↓</td>
</tr>
<tr>
<td>Time Base</td>
<td>continuous (reals)</td>
<td>discrete (integers)</td>
<td>continuous (reals)</td>
</tr>
<tr>
<td>Inputs, States, Outputs</td>
<td>real vector space</td>
<td>arbitrary</td>
<td>arbitrary</td>
</tr>
<tr>
<td>Input Segments</td>
<td>piecewise continuous</td>
<td>sequences</td>
<td>discrete-event</td>
</tr>
<tr>
<td>State &amp; Output Segments</td>
<td>continuous</td>
<td>sequences</td>
<td>piecewise constant</td>
</tr>
</tbody>
</table>
Co-simulation: promise & peril

• Co-simulation promises
  – A simple way to integrate disparate simulation tools
  – Easy to retrofit existing simulators
  – Easy to build support for in new simulators

• But danger lurks
  – Numerical errors caused by the interaction of components cannot be controlled in a useful way
  – Verification depends on engineering insight (it looks ok to me)
  – Numerical artifacts aren’t always obvious

Polypheus reconsiders his “easy” dinner.
What is co-simulation?

• A name for this algorithm
  1) Advance time in each simulator by h
  2) Stop and exchange data
  3) Repeat
Errors inherent in co-simulation

- Interactions are delayed on average by h/2
  - Data packets are always late
  - Control signals always act on old data
  - Thresholds are passed

- Only way to reduce error is to make h very small
  - Unacceptable running times if too small
  - So how small? No good answer.

In a co-simulation, the apple goes through Newton’s head.
**Discrete event systems: no h**

**Event trajectory:** In any finite interval, there are finite changes in value (i.e., a finite number of events).

**Discrete event systems:** State, input, and output trajectories are event trajectories.
Discrete event simulation of continuous dynamics

- Numerical integration is a discrete event simulation
  - Calculates an event trajectory to approximate a continuous one
  - Explicit Euler is a simple example:
    - $dx/dt = f(x)$ approximated by $x_{k+1} = x_k + hf(x_k)$
  - Can be cast as a discrete event model
Adding threshold detection to the mix

- Next event time depends on smallest of:
  - Step size for suitable error
  - Next sampling instant
  - Next threshold crossing
  - This is a root finding problem
  - Can also be cast as a discrete event model

Events located at $x = g(x)$
For a comprehensive treatment of the underlying mathematics and algorithms


Hybrid Systems in the Power Grid

<table>
<thead>
<tr>
<th></th>
<th>AC/DC circuit models</th>
<th>Digital controllers</th>
<th>Communications</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Specification</td>
<td>Differential Equation</td>
<td>Discrete-Time</td>
<td>Discrete-Event</td>
</tr>
<tr>
<td>Time Base</td>
<td>continuous (reals)</td>
<td>discrete (integers)</td>
<td>continuous (reals)</td>
</tr>
<tr>
<td>Inputs, States, Outputs</td>
<td>real vector space</td>
<td>arbitrary</td>
<td>arbitrary</td>
</tr>
<tr>
<td>Input Segments</td>
<td>piecewise continuous</td>
<td>sequences</td>
<td>discrete-event</td>
</tr>
<tr>
<td>State &amp; Output Segments</td>
<td>continuous</td>
<td>sequences</td>
<td>piecewise constant</td>
</tr>
</tbody>
</table>
The split system method

- Overarching simulation is discrete event
- Continuous sub-systems are simulated with embedded numerical algorithms

 output at state and time events

Complex discrete events dynamics

Continuous models with state events and other simple discrete event dynamics
The market participants interact through a market that issues discrete price signals.

The price signals experience a transmission and processing delay.
Sampling, delay, and stability in a market

Small simulation artifacts may have serious consequences!
Importance of threshold detection: Perceived frequency speed in a power systems
Perceived propagation speed depends on sensor precision

- Perceived speed is inversely proportional to precision of the sensor
- Miss the detection threshold $\rightarrow$ incorrect simulation of the sensor and any control that depends on it

<table>
<thead>
<tr>
<th>$q$ (Hz)</th>
<th>speed (miles/s)</th>
<th>$q$ (Hz)</th>
<th>speed (miles/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.001</td>
<td>3391.13</td>
<td>0.006</td>
<td>2900.55</td>
</tr>
<tr>
<td>0.002</td>
<td>3192.53</td>
<td>0.007</td>
<td>2861.80</td>
</tr>
<tr>
<td>0.003</td>
<td>3081.57</td>
<td>0.008</td>
<td>2828.69</td>
</tr>
<tr>
<td>0.004</td>
<td>3005.11</td>
<td>0.009</td>
<td>2799.84</td>
</tr>
<tr>
<td>0.005</td>
<td>2947.10</td>
<td>0.010</td>
<td>2774.32</td>
</tr>
</tbody>
</table>
Adevs: A Simulation Package for Hybrid Dynamic Systems

M&S Application

Physical & Digital Systems

External Tools

Adevs

Integration
- Predictor-Corrector

Root-finding
- Sign-change, value-cross, ...

Modelica support
- Continuous system modeling

Integration with other tools
- OMNET++, NS/3, etc.

Discrete Event Simulation
Electro-mechanical simulation with Adevs: The Toolkit for Hybrid Modeling of Electric Power Systems (THYME)

Transmission modeling with THYME

Dynamic models simulate frequency oscillations in the electric transmission system after an outage - A source for understanding electric grid state.

Integration with NS-2, NS-3, and OMNET++ network simulation tools
Support for the Modelica continuous system simulation language

Code generator for OpenModelica compiler produces Adevs modules

class GeneratorWithSensor extends Thyme.Generator;
    output Integer n(start=0);
    public parameter Real freqInterval(start=0.002);
    public parameter Real nomFreq(start=60.0);
algorithm
    when nomFreq*w > (n+1)*freqInterval then
        n := n+1;
    elseif nomFreq*w < (n-1)*freqInterval then
        n := n-1;
    end when;
end GeneratorWithSensor;

class twobus
    GeneratorWithSensor Genr(Pg0=2,Qg0=0,V0=2,Theta0=0,Xd=1,Xq=0);
    Thyme.ConstImpLoad Load(R=1,X=0,V0=1,Theta0=0,P0=1,Q0=0);
    Thyme.Plink L12(R=1);
    Thyme.Bus Bus1;
    Thyme.Bus Bus2;
equation
    connect(Genr.T,Bus1.T);
    connect(L12.T1,Bus1.T);
    connect(L12.T2,Bus2.T);
    connect(Load.T,Bus2.T);
end twobus;
How much does it cost? How much should we pay? M&S costs vs. capability

Who

Persons familiar with the domain

Buy a license

How much

What it can do

$ 

DoD wargames

Gridlab-D, THYME

Recurring costs to build and maintain

Expected savings
A big future for M&S in the power industry

1910

1958

Today

Increasingly complex systems \( \rightarrow \) Increasingly sophisticated simulators

1972

2012

2020

Managed by UT-Battelle
for the U.S. Department of Energy
The End

- OpenModelica
  - https://www.openmodelica.org/

- Adevs
  - http://sourceforge.net/projects/adevs/
  - http://www.ornl.gov/~1qn/adevs/index.html

- Thyme
  - http://www.ornl.gov/~1qn/thyme/docs/index.html