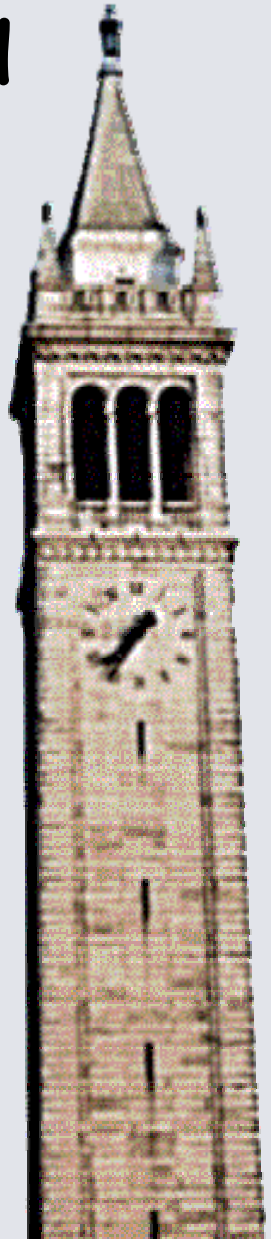
Efficient Simulation Model for Hybrid Bond Graphs

Edited and presented by
Christopher Beers
ISIS, Vanderbilt University



Chess Review
November 21, 2005
Berkeley, CA

Introduction

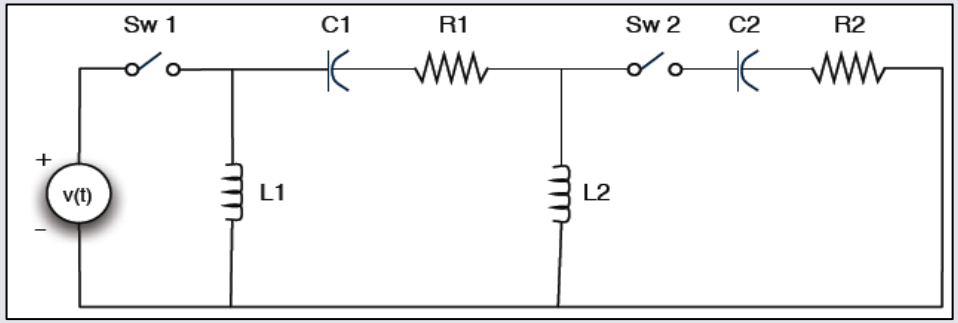
- ***Simulation of hybrid systems must combine two models of computation***
- Hybrid Bond Graphs (HBGs) combine
  - Continuous bond graph (BG) models with
  - Switching junctions controlled by FSM

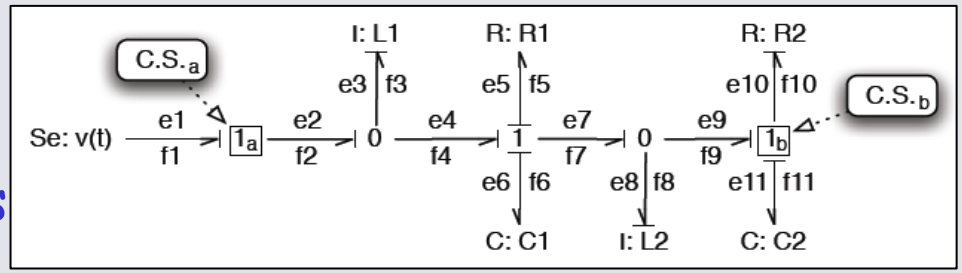
  to provide a topological framework that supports run-time model reconfiguration
- However, no computational model associated with HBGs

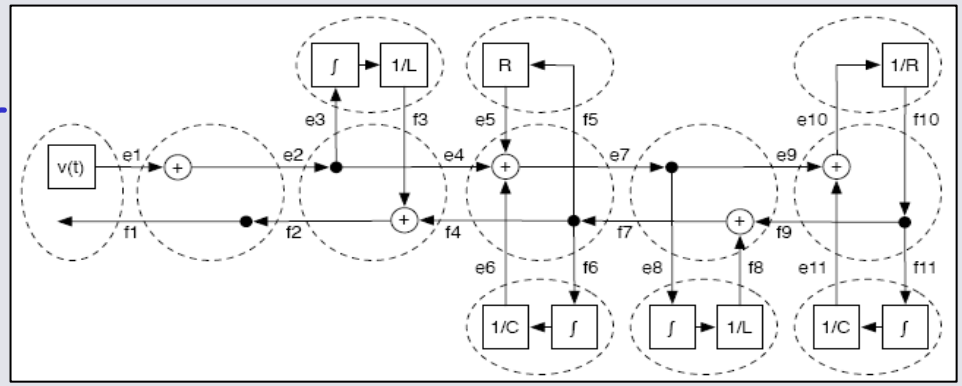
  Question: *How does one systematically derive simulation models from HBGs?*
- Approach: Use causal structure implied by BG to derive block diagram models for simulation (SCAP algorithm)

HBG Overview

- BG to Block Diagram Computational Model
  - Constituent element blocks + algebraic relations at junctions
- Determining Bond (DB)
  - One per junction, derived from causality at junction
  - Determines algebraic relations
- HBG Complexities
  - Junction switches (on and off) may cause causality changes at runtime, thus block diagram may change
  - Only changes in DBs will change algebraic relations at junction
  - These changes can propagate



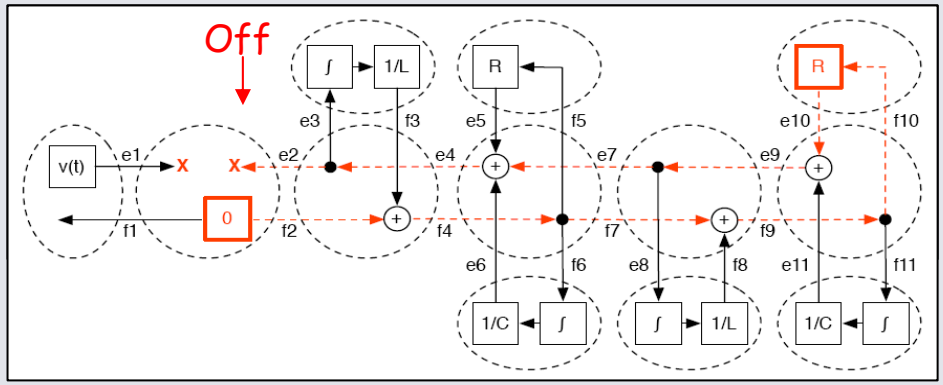




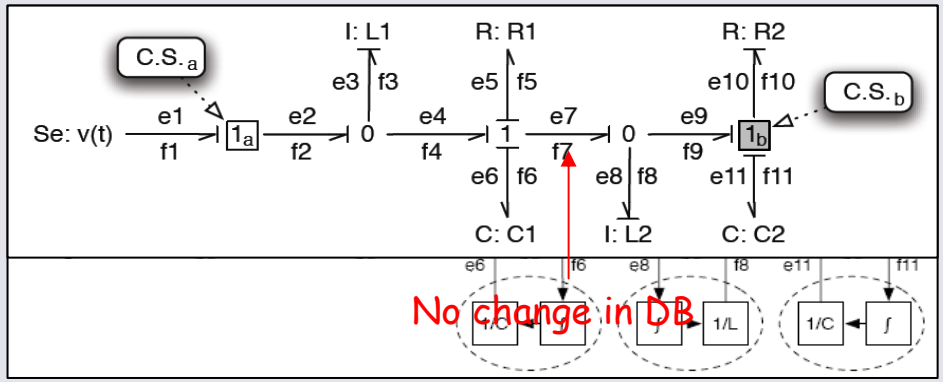
Approaches

- Pre-generate block diagrams for all modes
  - 2^n possible configurations
- Generate block diagrams from scratch after every mode change
  - Can be computationally expensive at switches
- Smarter approach: derive new block diagram incrementally from old
  - Start with block diagram in initial mode
  - Look for changes in DBs
    - Update block diagram at changes

Junction 1a off, junction 1b on:

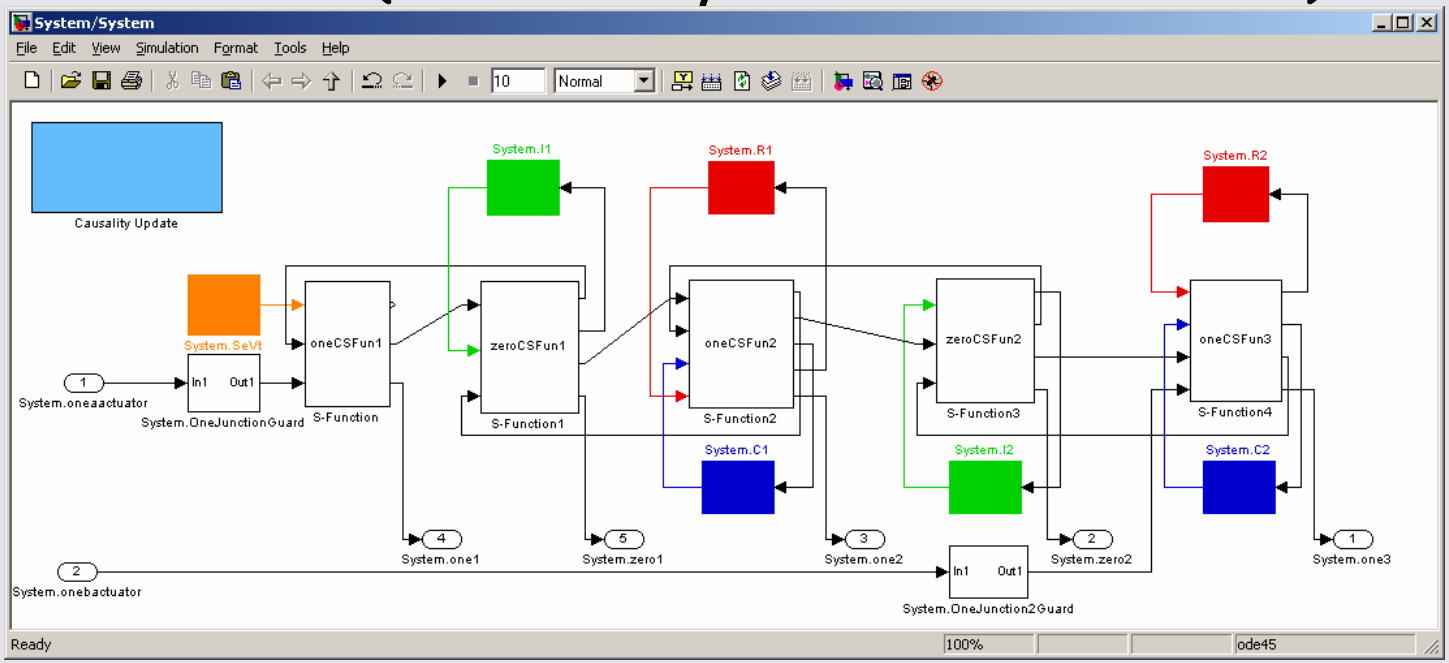


Junction 1a on, junction 1b off:



Efficient Simulation Model

- Causality update triggered by change in discrete state
  - Start at junctions which switch
  - If they cause changes in adjacent junction DBs then
    - update DB's algebraic constraints
  - Continue till no DB change or all junctions visited
- For efficiency, junctions implemented as S-functions; use global variables (cf. Ptolemy's director function)



Questions?