IsamDAE, an Implicit Structural Analysis Tool for Multimode DAE Systems

Benêt CALLAUD, Mathias MALANDAIN, Joan THIBAULT
Inria, Univ Rennes, CNRS, IRISA, Hycomes team (Rennes, France)
benoit.callaud@inria.fr – mathias.malandain@inria.fr – joan.thibault@irisa.fr

DAE FOR THE MODELING OF (LARGE) CYBER-PHYSICAL SYSTEMS

Large physical systems → Differential-Algebraic Equations (DAE)

- Components modeled with a few differential and/or algebraic equations
- Connections: algebraic equations (balance equations, Kirchhoff laws)
- Pros: compositional modeling, high reusability of physical components

Non-smooth, or hybrid, systems → Multimode DAE (mDAE)

- In general, mDAE are poorly handled by existing tools
  - No structural analysis of mDAE at compile time
- Reconfigurable systems → Variable dimension models
  - (Dis)appearance of components would be easily handled
  - Such models are not handled by existing tools

THE RLDC2 EXAMPLE

- Two RLC circuits with ideal diodes, in parallel
- Mode-dependent structure: differentiation orders for the equations, as well as the resulting equation blocks, depend on the mode
- Not handled by current DAE-based tools (OpenModelica, Dymola)

Mode-dependent structural analysis by IsamDAE

- Checks that the model is determined in all modes
- Generates the mode-dependent scheduling graph shown below
  - In each mode: description of the (ODE-like) blocks to be compiled into simulation code + dependencies (for their scheduling)
    - Same results, in each mode, as those of standard structural analysis

STRUCTURAL ANALYSIS OF DAE

Handling DAE systems is harder than ODE

- No direct criterion to determine whether a DAE is determined
- Simulation code cannot be generated from the original model
- Compilation relies on Structural Analysis (SA)

Structural analysis

- Graph-based (Pantelides method) or linear programming (Prьеce’s Sigma-method) algorithms
- Provides diagnosis about the model at compile time (‘typechecking’)
- Enables the generation of efficient simulation code:
  - equations are carefully differentiated, and grouped into the smallest possible equation blocks;
  - blocks are ordered (‘scheduled’) for the simulation of a time step

THE ISAMDAE TOOL

Structural analysis for mDAE, in an `all-modes-at-once’ fashion

- Input: Ad hoc equation language used for declaring mDAE models
  - Implicit representation with BDD (preserves sparsity)
  - Structural analysis (from the Sigma-method) and block scheduling: adaptation of existing methods and novel algorithms
- Output: Detailed (but condensed) description of the simulation blocks and their dependencies, in all modes

Implementation

- About 25,000 lines of OCaml code
- Libraries: Guacamol, menhir, Mirbddd, pprtprint, xml-light, SnowFlake
- Software package: GraphViz

THE BUILDING EXAMPLE

- Scalable model: single-story building with N rooms and doors
- Instantaneous mode transitions when a door is opened or closed
- Variables taken into account: pressures, temperatures →, air masses, enthalpies, heat and mass flows (compressible air)
- For N rooms: 16N+5 equations and variables, 6N/2 modes

Scaling-up of the structural analysis by IsamDAE

- Total number of blocks : affine function of N
- Empirical computational times: about O(N²)
- Up to N=150, ≥ 2500 equations, ≥ 10¹⁰ modes