Equation-Based Model Data Structure for High Level Physical Modelling, Model Simplification and Modelica-Export

Hisahiro Ito*, Akira Ohata
Toyota Motor Corporation, Japan

Ken Butts
Toyota Technical Centre – Ann Arbor, USA

Jürgen Gerhard, Masoud Abbaszadeh, David Linder,
Erik Postma, Elena Shmoylova
Maplesoft, Canada

Outline

- Introduction
  - Plant Modelling Process
- Physics-Based Component Modelling
  - High Level Model Description / HLMD
  - High Level Modelling Tool / HLMT
- Equation-Based Model Data Structure
  - MSModel and Simplification
  - Modelica-Export
- Modelica & HLM/MSModel
- Conclusion

Introduction

Plant Modelling Process

In-vehicle Control System Development

- More challenging requirements
  - Better fuel economy
  - Less emission
  - Proven functional safety
    - including safety under unexpected circumstances
  - Without compromising driveability
- More time and cost are spent
  - How to keep competitiveness?

Concurrent Development

- Avoid bloat of development time and cost
  - Conventional development process:
    - Develop hardware first, software second
  - Instead, develop H/W and S/W together
  - Impact of the concurrent development
    - Experiment may be impossible at early stage of software development process
    - Key is to embrace Model-Based Development
    - Plant modelling is essential for MBD!

Need for Plant Model Creation

- Specific requirements for plant models in a control system development project
  - Not known a priori
- Existing plant model libraries
  - Useful as a starting point for further change to meet specific requriements
  - Too large library is hard to maintain, hinders swift model authoring

Desired Plant Modelling Technologies 1/2

- ***Physical component modelling … "HLMD & HLMT"***
- Data-based modelling
  - Simulated / measured
- Physics-and-data combined system modelling
  - Incremental introduction of a data-based subsystem model into/instead of a part of physics-based system model
  - As more data is made available through experiment

Desired Plant Modelling Technologies 2/2

- ***Model simplification … "MSModel"***
  - Component-level / System-level
  - Symbolic / Numeric / Symbolic-numeric combined
- Optimization
- Model / Data / Process management
  - Efficient reuse/exchange
  - Process standardization
  - Traceability

Rapid Modelling 1/2

- Plant models have to be built in a timely manner
  - Specific requirement is not known a priori
- Stance:
  - ***Have modelling methods at hand, ready to creat/modify a model for a specific need responsively.***

Rapid Modelling 2/2

- Equation-based approach?
  - Versatile, but needs additional methods to be rapid enough
- Sophisticated modelling techniques?
  - Not widely accepted for production use in Toyota, e.g.
    - Abstraction and inheritance
    - Bond-graph
  - “Simple over clever” is preferred
    - Efficient comprehensibility matters a lot

Goal of Plant Modelling Process

- Build a closed-loop simulation system with control software (XiL)
  - MiL ... Control design
  - SiL ... Software implementation verification & validation (V&V)
  - HiL ... ECU V&V - computation load, execution and communication timing etc.
  - Control software is developed using Simulink
  - Simscape is the target platform for plant models
    - Connectivity with Control Software
    - Code-Generation

Plant Modelling Process

- Start from physics-based component modelling
  - Design phase of a plant model component using high level modelling framework, HLMD/HLMT
- Simplify and export to Modelica via MSModel data structure
- Further transform to Simscape for system modelling
  - Perform integration with data-based models, system-level simplification, optimization
- Close loop with control software (MiL/SiL) or ECU (HiL)

Physics-Based Component Modelling

High Level Model Description / HLMD

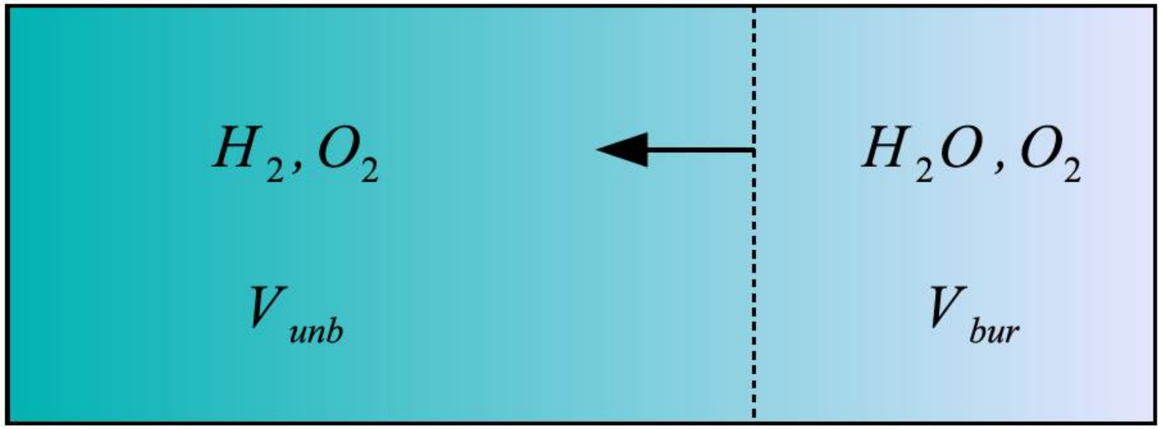
High Level Model Description / HLMD

- Description of a physical component
  - For efficient design, review and reuse
  - Control volume approach
  - Set of equations can be derived
    - Derivation requires symbolic manipulation
- Domain neutral
  - For continuous system
    - Mechanical, thermal, electric, neumatic, chemical, viscoelastic, electrochemical, ...
    - Includes “piecewise”, only for smooth switching
  - Knowledge on constitutive equations requried

HLMD Example

- Target physical component:
  - A closed chamber
    - Combustion occurs insdie:

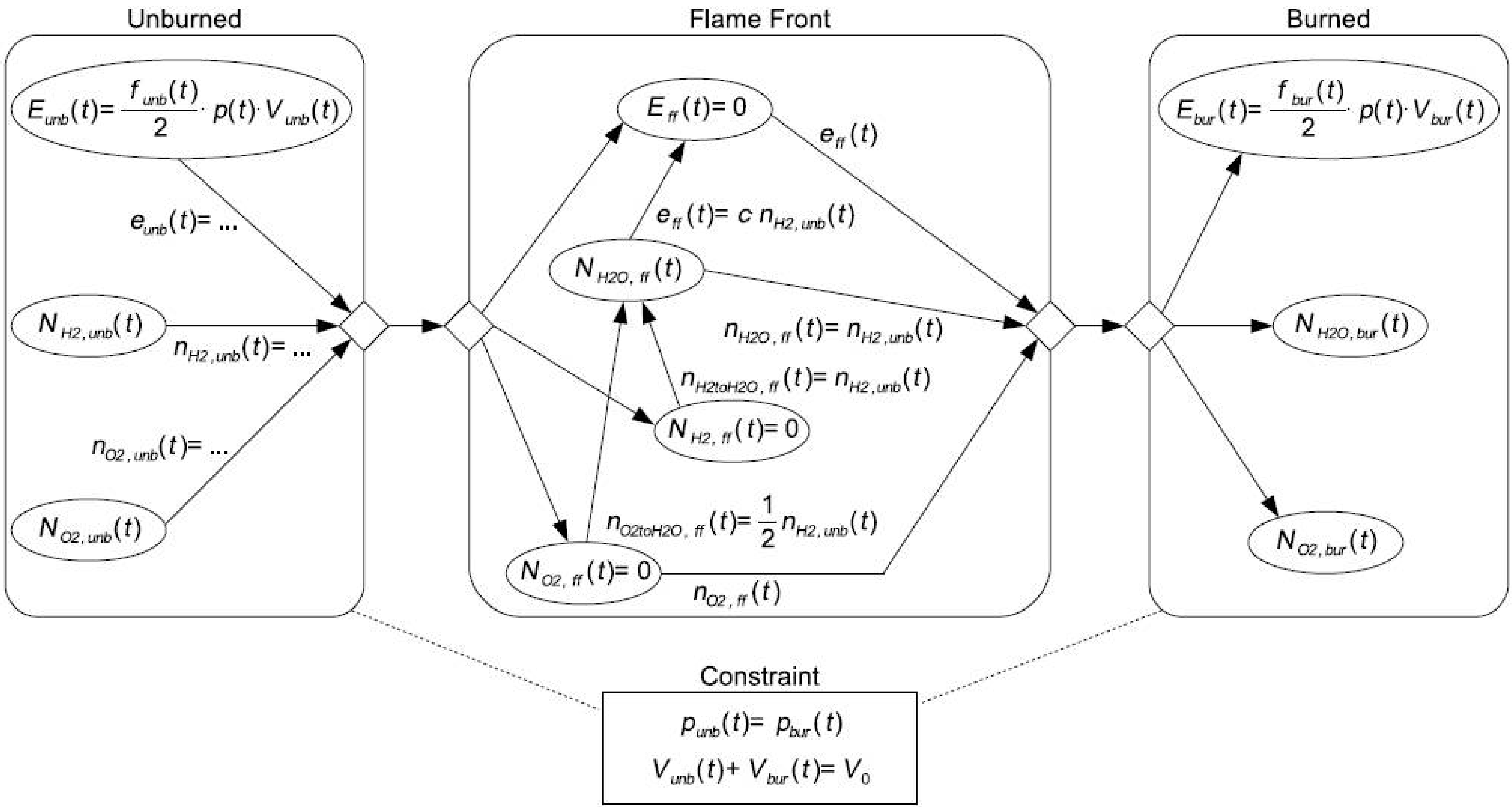
$$H_2 + O_2 \rightarrow H_2O + \frac{1}{2}O_2 + c\,[J]$$



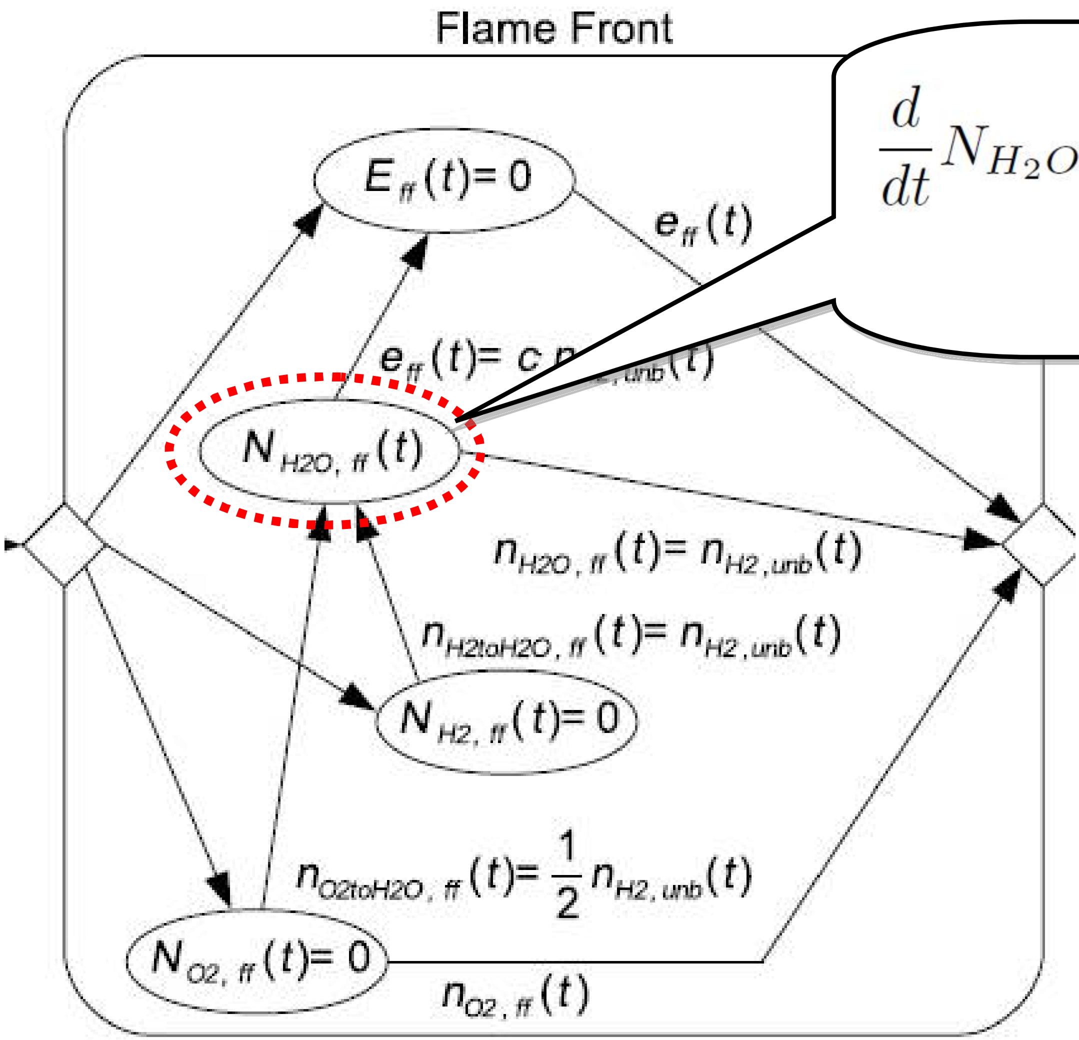
Modelling Steps to Create HLMD

- 1) Split a physical component (an HLM system) into HLM components
- 2) Define ports for each HLM component
- 3) Connect HLM components via ports
- 4) For each HLM component,
  - Define conservation quantities (CQs)
  - Define flows of CQs
- 5) Set consititutive equations for CQs and CQ flows
- 6) Set HLM system-level constraints as needed
- Additionally, intermediate variables can be defined inside an HLM component and at a port
- Parameters (constant symbols) can be defined as needed

HLMD of Combustion Model in A Closed Chamber



Equation Construction from HLMD



$$
\begin{aligned}
\frac{d}{dt} N_{H_2O,\mathrm{ff}}(t) = & n_{H_2 \mathrm{to} H_2O,\mathrm{ff}}(t) + n_{O_2 \mathrm{to} H_2O,\mathrm{ff}}(t) \\
& - n_{H_2O,\mathrm{ff}}(t) - e_{\mathrm{ff}}(t)
\end{aligned}
$$

- Equations are built to conserve CQs

Plain Set of HLM Equations

- The combustion model
  - Has 44 equations
    - 13 differential equations (DEs)
    - 31 algebraic equations (AEs)
- Not ready for simulation in general
  - Physically
    - Could have multiple solutions
  - Mathmatically
    - Could be a high-index DAE system
    - May contain redundant equations

Deriving a ***Simulatabale*** Set of Equations

- Fully automated
  - Methods
    - Isolated HLM component detection
      - for single solution selection
    - Symbolic manipulation including index reduction
      - for high-index DAE, redundant equations
  - Derives index-1 DAEs or ODEs
- Proof of single solution existance
  - Research in progress

Simulatable Set of Equations

- The combustion model
  - Plain HLM had 44 equations
    - 13 DEs
    - 31 AEs
  - Simulatable HLM has 43 equations:
    - 11 DEs (7 explicit, 4 implicit)
      - 8 differential varialbes (DVs) need initial conditions
      - 3 other variables are algebraic (AVs)
    - 32 AEs (32 explicit, 0 implicit)

FYI: 7 Explicit Differential Equations

$$1, \ldots, \frac{d}{dt} bur.Nh2o(t) = 2\, no2I(t) - 2\, no2O(t)$$

$$2, \ldots, \frac{d}{dt} bur.No2(t) = no2O(t)$$

$$3, \ldots, \frac{d}{dt} bur.f(t) = \frac{-2\, bur.f(t)\, no2I(t) + bur.f(t)\, no2O(t) + 12\, no2I(t) - 7\, no2O(t)}{bur.No2(t) + bur.Nh2o(t)}$$

$$4, \ldots, \frac{d}{dt} mid.Nh2o(t) = -4839449\, no2I(t) + 4839449\, no2O(t)$$

$$5, \ldots, \frac{d}{dt} mix.Nh2(t) = -2\, no2I(t) + 2\, no2O(t)$$

$$6, \ldots, \frac{d}{dt} mix.No2(t) = -no2I(t)$$

$$7, \ldots, \frac{d}{dt} mix.p(t) = -\frac{1}{5} \frac{5\, mix.p(t) \left(\frac{d}{dt} mix.V(t) \right) + 2\, mix.e(t)}{mix.V(t)}$$

(Parameter values have been assigned.)

FYI: 4 Implicit Differential Equations

$$1, \ldots, (16\, mix.p(t)\, mix.No2(t) + mix.p(t)\, mix.Nh2(t)) \left(\frac{d}{dt} mix.V(t)\right) + (-mix.Nh2(t) - 16\, mix.No2(t))\, mix.e(t) + 63\, mix.p(t)\, mix.V(t)\, no2I(t) - 7\, mix.p(t)\, mix.V(t)\, no2O(t) = 0$$

$$2, \ldots, (5\, mix.p(t)\, bur.No2(t)\, bur.f(t) + 5\, mix.p(t)\, bur.Nh2o(t)\, bur.f(t)) \left(\frac{d}{dt} mix.V(t)\right) + (10\, mix.V(t)\, bur.No2(t) + 10\, mix.V(t)\, bur.Nh2o(t) - 2\, bur.f(t)\, mix.V(t)\, bur.No2(t) - 2\, bur.f(t)\, mix.V(t)\, bur.Nh2o(t) + 2\, bur.No2(t)\, bur.f(t) + 2\, bur.Nh2o(t)\, bur.f(t))\, mix.e(t) + (48394500\, mix.V(t)\, bur.No2(t) + 48394500\, mix.V(t)\, bur.Nh2o(t) - 10\, mix.p(t)\, mix.V(t)^2\, bur.f(t) + 60\, mix.p(t)\, mix.V(t)^2 + 10\, bur.f(t)\, mix.p(t)\, mix.V(t) - 60\, mix.p(t)\, mix.V(t))\, no2I(t) + (-48394500\, mix.V(t)\, bur.No2(t) - 48394500\, mix.V(t)\, bur.Nh2o(t) + 5\, mix.p(t)\, mix.V(t)^2\, bur.f(t) - 35\, mix.p(t)\, mix.V(t)^2 - 5\, bur.f(t)\, mix.p(t)\, mix.V(t) + 35\, mix.p(t)\, mix.V(t))\, no2O(t) = 0$$

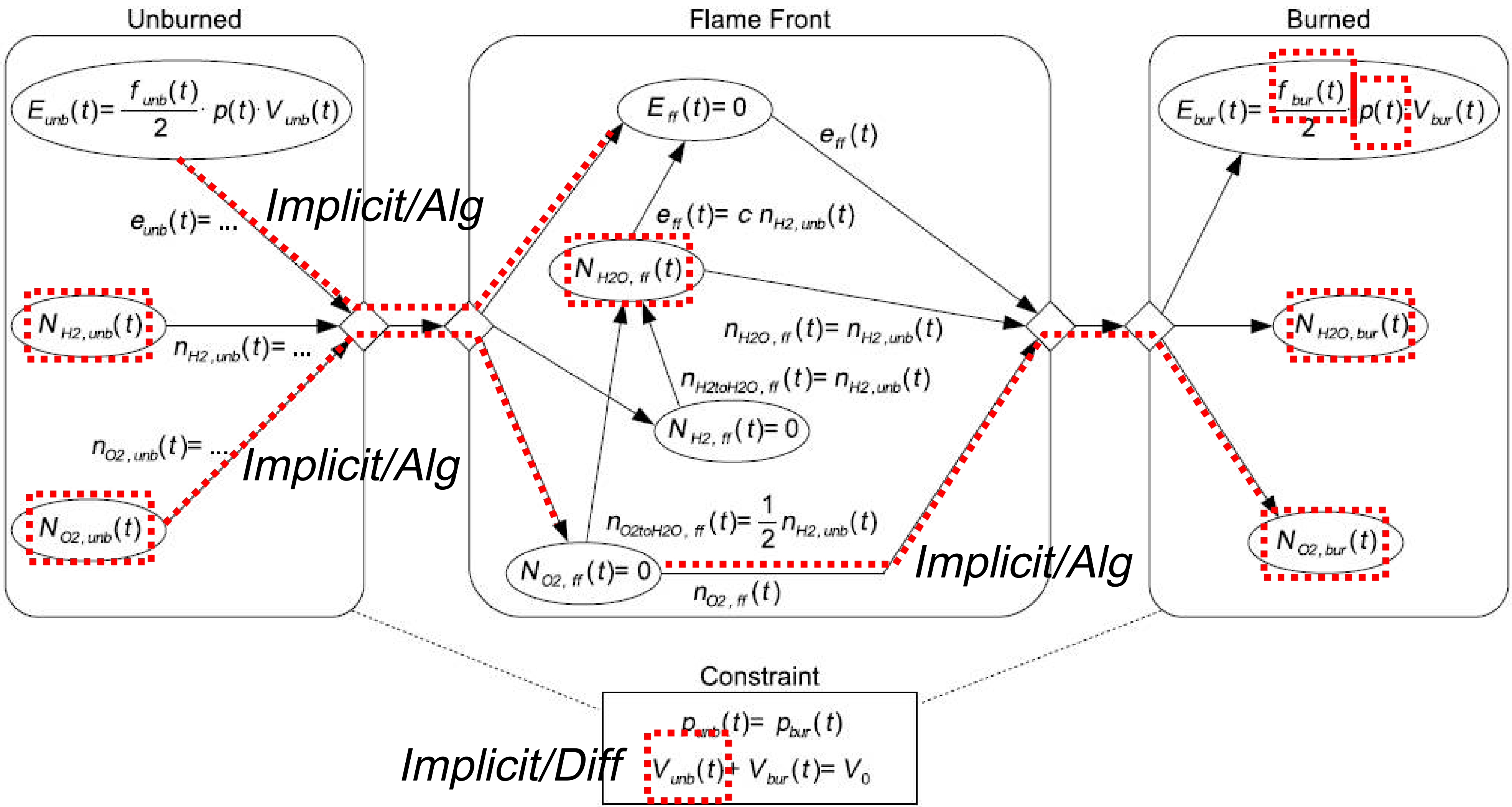
$$3, \ldots, -601365581\, mix.No2(t)\, mix.p(t)\, mix.Nh2(t) - 9621849296\, mix.No2(t)^2\, mix.p(t) + (601365581\, mix.No2(t)\, mix.p(t)\, mix.Nh2(t) + 9621849296\, mix.No2(t)^2\, mix.p(t)) \left(\frac{d}{dt} mix.V(t)\right) + (62500000000000000\, mix.Nh2(t) + 62500000000000000\, mix.No2(t))\, no2I(t) = 0$$

$$4, \ldots, -601365581\, mix.Nh2(t)^2\, mix.p(t) - 9621849296\, mix.No2(t)\, mix.p(t)\, mix.Nh2(t) + (601365581\, mix.Nh2(t)^2\, mix.p(t) + 9621849296\, mix.No2(t)\, mix.p(t)\, mix.Nh2(t)) \left(\frac{d}{dt} mix.V(t)\right) + (125000000000000000\, mix.Nh2(t) + 125000000000000000\, mix.No2(t))\, no2I(t) + (-125000000000000000\, mix.Nh2(t) - 125000000000000000\, mix.No2(t))\, no2O(t) = 0$$

(Parameter values have been assigned.)

FYI: Observation in HLMD

- Variables in 11 DEs in the combustion model



Physics-Based Component Modelling

High Level Modelling Tool / HLMT

High Level Modelling Tool / HLMT

- A software package for HLMD
  - Toyota developed with Maplesoft, owns IP
  - 1st prototype in 2008
  - 2nd prototype development plan later this year
- Modelling in HLMD and simulation possible
- Built on top of Maple, commercial symbolic manipulation software from Maplesoft
  - API available as Maple library, in addition to GUI
- Hoping to make it open
  - MSModel is the first step

HLMT GUI

Model tree

Messages

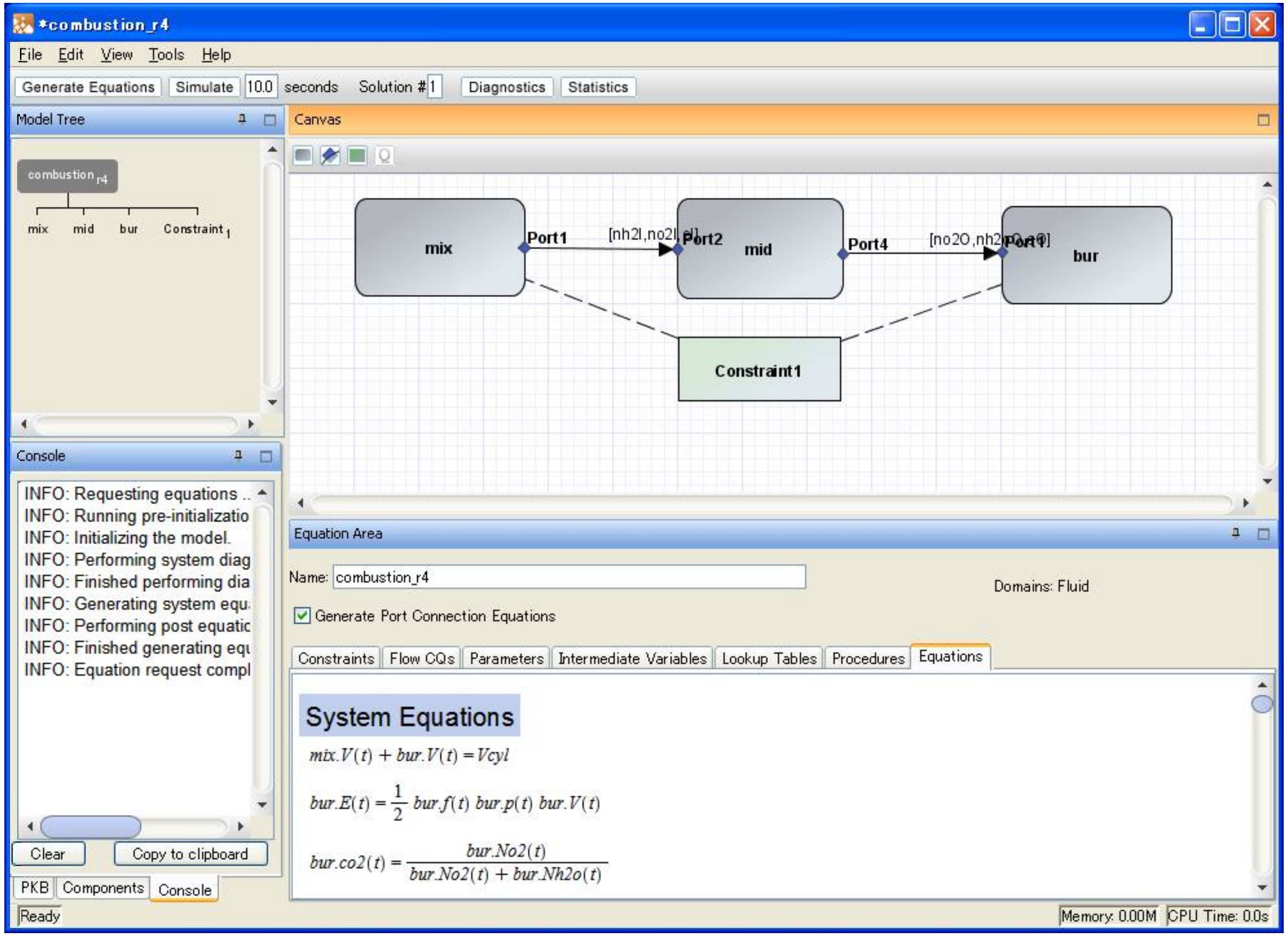


Diagram editor

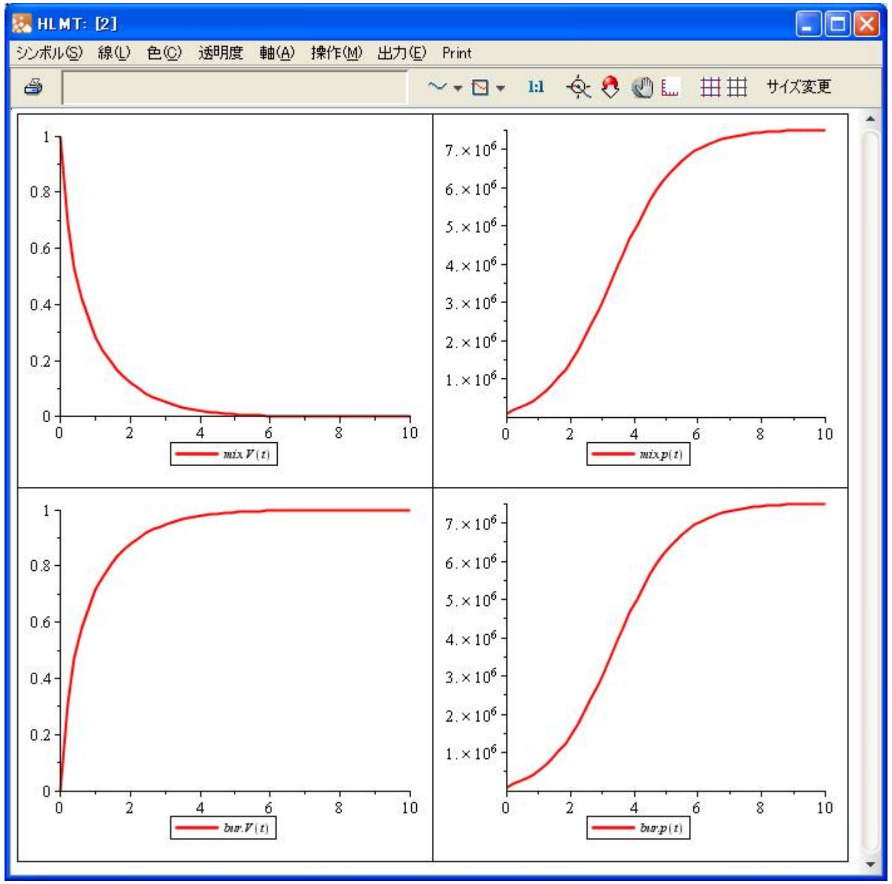
Equations editor & viewer

Simulation in HLMT

- Plots of the combustion model simulation

Volume of unbunred gas

Pressure of unbunred gas



Volume of bunred gas

Pressure of bunred gas

Future Topic for HLM Framework

- Equation-based physical knowledge base repository
  - Repository of constitutive equations in various engineering domains:
    - Mechanical, thermal, electric, neumatic, chemical, viscoelastic, electrochemical, ...
  - Necessary for efficient modelling in HLM framework
  - Open format/specification preferred
- Proof of single solution existance

Equation-Based Model Data Structure

MSModel & Simplification

Equation-Based Model Simplification Research Project

- A reserach collaboration project between Maplesoft and Toyota
  - Symbolic / Numeric / Combined methods
  - Algorithms are implemented in Maple
  - Supports HLMD models
- MSModel data structure
  - Designed in this project
  - Realized in Maple language

MSModel

- Data Structure for Model Simplification
  - Designed for open R&D collaboration on equation-based model simplification methods
  - Specification has been published in our paper
  - Neutral for equation-based languages and tools
- Requirements: it can...
  - Store information generated in HLMT
    - Either plain or simulatable equations
  - Store a simplified/simulatable set of equations
  - Provide convenience for simplification methods
  - Generate a Modelica representation

MSModel Elements

- Core equations
  - Required to compute at every integration step
- Non-core equations
  - Required to compute only when necessary
- Additional information
  - Useful pieces of information for simplification and Modelica-export
    - Variable list etc.

Core Equations

- Required to compute at every integration step:
  - Differential equations / DEs
  - Algebraic equations / AEs
  - Intermediate equations / IEs
    - Stored in straight-line causal arrangement
    - No derivatives

Non-core Equations

- Required to compute only when necessary:
  - Dependent equations
    - Stored in straight-line causal arrangement
    - May be implicit
    - May contain derivatives

Additional Information

- Parameters
- Inputs and outputs
- Name, type, value of variables
- Blackbox functions
  - lookup tables, user-defined functions

Fictitious Example of MSModel

```
msm := Record(MSMODEL,
 DE=[ ( diff(x1(t),t)=-a*x1(t)+u1(t) ), ...],
 DV=[ 'x1' , ...],
 AE=[],
 AV=[],
 t='t',
 intermediate=(Array(1..0,[])),
 intermediateVariables=[],
 dependent=(Array(1..3,{
  1=[{ e1(t)=-1/2*sin(x1(t)) },{e1(t)}],
  2=[{ e2(t)=u1(t)*e1(t) },{e2(t)}],
  3=[{ y(t)=e1(t)+e2(t) },{y(t)}] })),
 dependentVariables=[ 'e1' , ...],
 parameters=[ 'a' , ...],
 inputs=[ 'u1' , ...],
 outputs=[ 'e1' , ...],
 variables=(table([
  (x1)=Record(MSVARIABLE, name=x1, type="differential", value=.9, unit=(NULL)),
  (a)=Record(MSVARIABLE, name=a, type="parameter", value=2, unit=(NULL)),
   ...])),
 blackboxes=[]
);
```

Equation-Based Model Simplification Methods

- Potential simplification methods to apply
  - Elimination
    - Removal of elementary equations (constants, equivalences)
    - Abstraction of common subexpressions (sum, products)
  - Generalized projection method for index reduction
  - Exact parameter reduction
  - etc. etc.
- Introduce intermediate variables and equations
  - As needed when the model can be reduced
  - Opcount is used to check how much reduced

FYI: Opcount

- Operation counts
  - Weighted counts of the number of operations
    - + 5; * 6; - 1; / 10; ^ 40; eval 50; > 5; and so on
  - Parameter symbols are replaced with values before counting
  - The cost of piecewise is the most expensive branch

Model Simplification Example

- The combustion model
  - Simulatable equations had 43 equations
    - Core: 11 DEs, 8 DVs; 32 AEs, 3 AVs; 0 IEs, 0 IVs
    - Non-core: 33 depEs, 33 depVs
    - 2093 opcounts
  - Simplified model has 28 equations
    - Core: 9 DEs, 6 DVs; 0 AEs, 3 AVs; 7 IEs, 7 IVs
    - Non-core: 12 depEs, 12 depVs
    - 1733 opcounts (17% reduction)

FYI: 9 Simplified DEs

$$1, \ldots, \left(eO(t) - mix.p(t)\left(\frac{d}{dt} mix.V(t)\right) - 4839450\, no2I(t) + 4839450\, no2O(t)\right) mix.Nh2(t) + \left(16\, eO(t) - 77431200\, no2I(t) - 16\, mix.p(t)\left(\frac{d}{dt} mix.V(t)\right) + 77431200\, no2O(t)\right) mix.No2(t) - 63\, v_6(t)\left(no2I(t) - \frac{1}{9}\, no2O(t)\right) = 0$$

$$2, \ldots, v_{15}(t)\left(-1 + \frac{d}{dt} mix.V(t)\right) mix.No2(t)^2 + \left(v_{14}(t)\, mix.Nh2(t)\left(-1 + \frac{d}{dt} mix.V(t)\right) + 62500000000000000\, no2I(t)\right) mix.No2(t) + 62500000000000000\, no2I(t)\, mix.Nh2(t) = 0$$

$$3, \ldots, \left(v_{12}(t)\left(\frac{d}{dt} mix.V(t)\right) + (v_{11}(t) - 5)\left(\frac{d}{dt} mix.p(t)\right) + 2\, eO(t)\right) bur.No2(t)^2 + \left(\left(11\, mix.p(t)\left(\frac{d}{dt} mix.V(t)\right) + (11\, mix.V(t) - 11)\left(\frac{d}{dt} mix.p(t)\right) + 4\, eO(t)\right) bur.Nh2o(t) - 2\, v_{13}(t)\,(no2I(t) - no2O(t))\right) bur.No2(t) + 6\, bur.Nh2o(t)\left(\left(mix.p(t)\left(\frac{d}{dt} mix.V(t)\right) - bur.V(t)\left(\frac{d}{dt} mix.p(t)\right) + \frac{1}{3}\, eO(t)\right) bur.Nh2o(t) + \frac{1}{6}\, v_{13}(t)\, no2O(t)\right) = 0$$

$$4, \ldots, v_{11}(t)\left(\frac{d}{dt} mix.p(t)\right) + v_{12}(t)\left(\frac{d}{dt} mix.V(t)\right) + 2\, eO(t) - 9678900\, no2I(t) + 9678900\, no2O(t) = 0$$

$$5, \ldots, v_{14}(t)\left(-1 + \frac{d}{dt} mix.V(t)\right) mix.Nh2(t)^2 + \left(v_{15}(t)\left(-1 + \frac{d}{dt} mix.V(t)\right) mix.No2(t) + 125000000000000000\, no2I(t) - 125000000000000000\, no2O(t)\right) mix.Nh2(t) + 125000000000000000\, mix.No2(t)\,(no2I(t) - no2O(t)) = 0$$

$$6, \ldots, \frac{d}{dt} bur.Nh2o(t) = 2\, no2I(t) - 2\, no2O(t)$$

$$7, \ldots, \frac{d}{dt} mix.Nh2(t) = -2\, no2I(t) + 2\, no2O(t)$$

$$8, \ldots, \frac{d}{dt} bur.No2(t) = no2O(t)$$

$$9, \ldots, \frac{d}{dt} mix.No2(t) = -no2I(t)$$

(Parameter values have been assigned.)

FYI: 7 Intermediate Equations

$$1 = [\{bur.V(t) = -mix.V(t) + 1\}, \{bur.V(t)\}]$$
$$2 = [\{v_6(t) = mix.p(t)\ mix.V(t)\}, \{v_6(t)\}]$$
$$3 = [\{v_{11}(t) = 5\ mix.V(t)\}, \{v_{11}(t)\}]$$
$$4 = [\{v_{12}(t) = 5\ mix.p(t)\}, \{v_{12}(t)\}]$$
$$5 = [\{v_{14}(t) = 601365581\ mix.p(t)\}, \{v_{14}(t)\}]$$
$$6 = [\{v_{15}(t) = 9621849296\ mix.p(t)\}, \{v_{15}(t)\}]$$
$$7 = [\{v_{13}(t) = mix.p(t)\ bur.V(t)\}, \{v_{13}(t)\}]$$

(Parameter values have been assigned.)

Equation-Based Model Data Structure

MSModel & Modelica-export

Modelica-Export from MSModel

- A feature implemented in Maplesoft-Toyota MSModel realization
- Modelica is viewed as a “hub” for global plant modelling R&D effort
- But, MSModel is not limited to Modelica

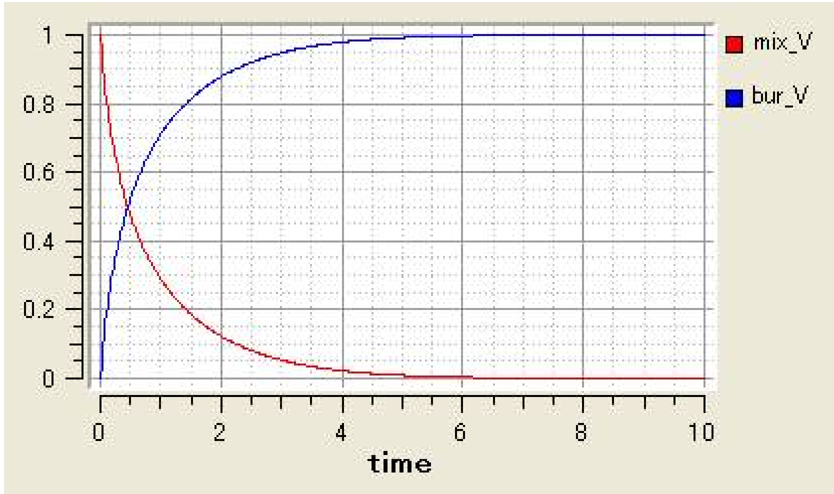
Modelica-Exported Combustion Model

```
equation
    (v_12_ * der(mix_V) + (v_11_ - 5) * der(mix_p) + 2 * eO) * bur_No2 ^ 2 + ((11 * mix_p
      * der(mix_V) + (11 * mix_V - 11) * der(mix_p) + 4 * eO) * bur_Nh2o - 2 * v_13_ * (no2I
      - no2O)) * bur_No2 + 6 * bur_Nh2o * ((mix_p * der(mix_V) - bur_V * der(mix_p) + 1/3 *
      eO) * bur_Nh2o + 1/6 * v_13_ * no2O) = 0;
    (eO - mix_p * der(mix_V) - 4839450 * no2I + 4839450 * no2O) * mix_Nh2 + (16 * eO -
      77431200 * no2I - 16 * mix_p * der(mix_V) + 77431200 * no2O) * mix_No2 - 63 * v_6_ *
      (no2I - 1/9 * no2O) = 0;
    v_14_ * (-1 + der(mix_V)) * mix_Nh2 ^ 2 + (v_15_ * (-1 + der(mix_V)) * mix_No2 +
      125000000000000000 * no2I - 125000000000000000 * no2O) * mix_Nh2 + 125000000000000000 *
      mix_No2 * (no2I - no2O) = 0;
    v_15_ * (-1 + der(mix_V)) * mix_No2 ^ 2 + (v_14_ * mix_Nh2 * (-1 + der(mix_V)) +
      6250000000000000 * no2I) * mix_No2 + 6250000000000000 * no2I * mix_Nh2 = 0;
    v_11_ * der(mix_p) + v_12_ * der(mix_V) + 2 * eO - 9678900 * no2I + 9678900 * no2O = 0;
    der(bur_Nh2o) = 2 * no2I - 2 * no2O;
    der(mix_Nh2) = -2 * no2I + 2 * no2O;
    der(bur_No2) = no2O;
    der(mix_No2) = -no2I;
    bur_V = -mix_V + 1;
    v_6_ = mix_p * mix_V;
    v_11_ = 5 * mix_V;
    v_12_ = 5 * mix_p;
    v_13_ = mix_p * bur_V;
    v_14_ = 601365581 * mix_p;
    v_15_ = 9621849296 * mix_p;
```

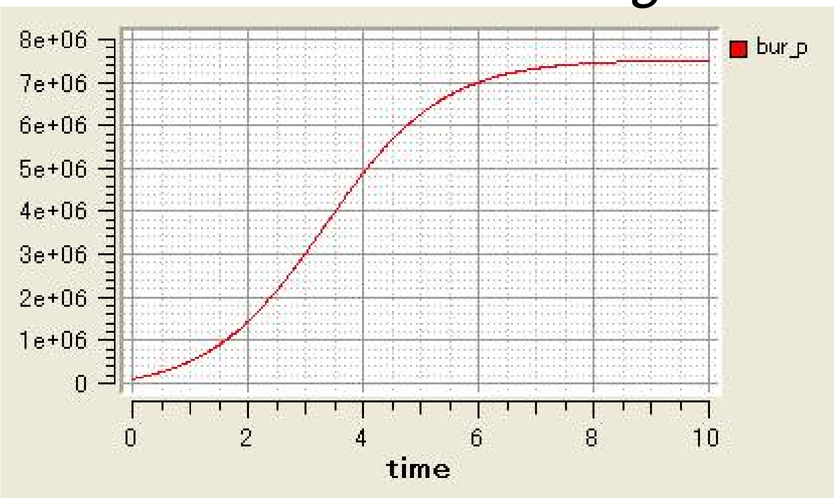
Simulation Result 1/2

Pressure of unburned gas

Volumes of burned/unbunred gas



Pressure of burned gas

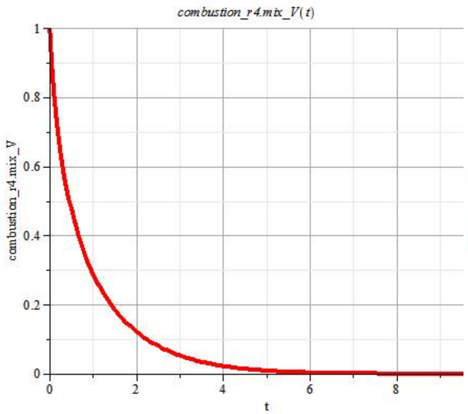


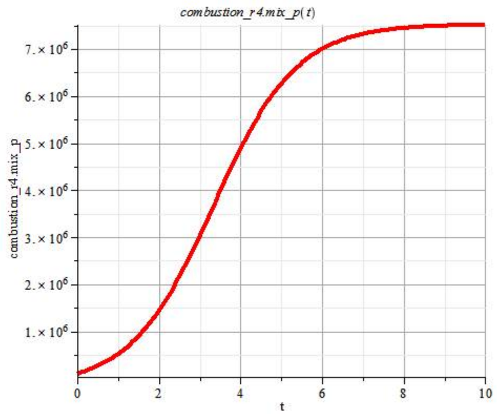
- The exported combustion model
- Produced in OpenModelica-1.7.0

Simulation Result 2/2

- Produced in MapleSim-4.5

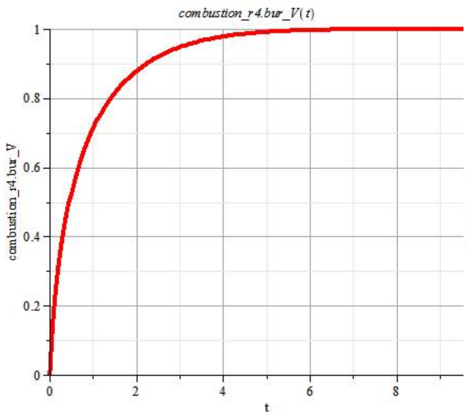
Volume of unbunred gas

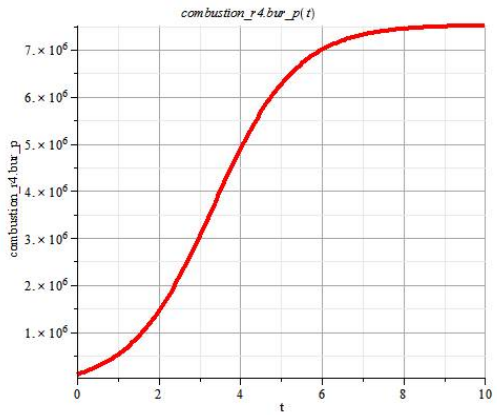




Pressure of unbunred gas

Volume of bunred gas





Pressure of bunred gas

Future Topic for MSModel

- Connectivity management for components
  - Open issue for MSModel
  - Need a design to cover HLMD, Modelica and Simscape at least
    - Modelica – Connector
    - Simscape – Domain

Modelica & HLM/MSModel

Modelica & HLM Framework 1/2

- HLMD & HLMT
  - Framework to create physical components
    - The whole purpose of HLMD/HLMT
    - Other features are out of scope (by design)
- Modelica & Modelica-based Tools
  - Modelling and simulation platform for the DAE system
    - Much more feature rich than the HLM framework
  - Inside of a Modelica component does not necessarily have to follow physical principles

Modelica & HLM Framework 2/2

- HLM framework can complement Modelica (and Simscape) for component creation
  - MSModel plays a key role to bridge HLMD and Modelica
  - Toyota is considering Simscape in addition to Modelica

Functional Mock-up Interface & MSModel

- Primary purpose of MSModel:
  - Equation-based model simplification
    - No support for many Modelica features
- Primary purpose of FMI:
  - Integrated simulation of Modelica-based components

Conclusion

Conclusion 1/2

- Introduced in this presentation
  - HLM framework
    - Physical component modelling method
    - Can complement Modelica, Simscape
  - MSModel
    - Data structure for model simplification
    - Key to bridge HLM and Modelica
    - Specificaiton has been published
- Key technology
  - Symbolic manipulation
    - For model simplification
    - Hope to have more research in an open environment

Conclusion 2/2

- Future topics
  - HLM framework
    - Proof of single solution existance
    - Physical knowledge base repository
  - MSModel
    - Connectivity management

Thank you.