Modelling and Simulation of Innovative Power Generation Systems in OpenModelica: The Case of PreFlexMS

Francesco Casella

(francesco.casella@polimi.it)

Stefano Trabucchi

(stetrab@hotmail.com)

POLITECNICO
MILANO 1863

A bit of history

- OpenModelica Workshop 2014:
  ThermoPower Library Simulation
  - ThermoPower library mostly covered
  - Modelica.Media models covered, including IF97
  - Test models and examples working
  - Up to 40X slower than Dymola on non-trivial test cases

A bit of history

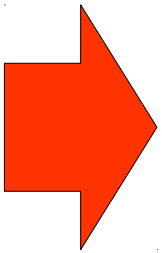
- OpenModelica Workshop 2014:
  ThermoPower Library Simulation
  - ThermoPower library mostly covered
  - Modelica.Media models covered, including IF97
  - Test models and examples working
  - Up to 40X slower than Dymola on non-trivial test cases

- OpenModelica Workshop 2015:
  Modelling of Energy Systems with OpenModelica
  - Laid out requirements for use of OMC in energy system modellling
  - Some initial interesting results on cryogenic system simulation

A bit of history

- OpenModelica Workshop 2014:
  ThermoPower Library Simulation
  - ThermoPower library mostly covered
  - Modelica.Media models covered, including IF97
  - Test models and examples working
  - Up to 40X slower than Dymola on non-trivial test cases

- OpenModelica Workshop 2015:
  Modelling of Energy Systems with OpenModelica
  - Laid out requirements for use of OMC in energy system modellling
  - Some initial interesting results on cryogenic system simulation



Where are we now?

Outline

- The Horizon 2020 PreFlexMS project
- The Plant Model
- Dynamic Controllability Analysis
- Dynamic Simulations
- Sharing the model with 3rd parties
- Outlook and conclusions

Solar Power Generation Systems

Photovoltaic (PV) systems

+ Low and decreasing costs
+ Exploit direct and diffuse radiation
+ Trivial operation and maintenance

Solar Power Generation Systems

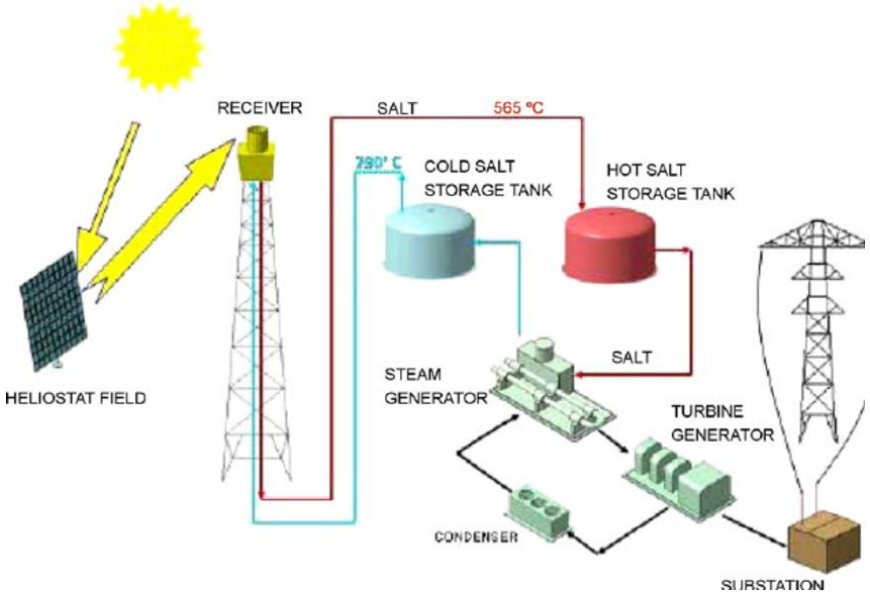
Photovoltaic (PV) systems

+ Low and decreasing costs
+ Exploit direct and diffuse radiation
+ Trivial operation and maintenance

- Only produce power when the sun is available
- Battery storage too expensive

Solar Power Generation Systems

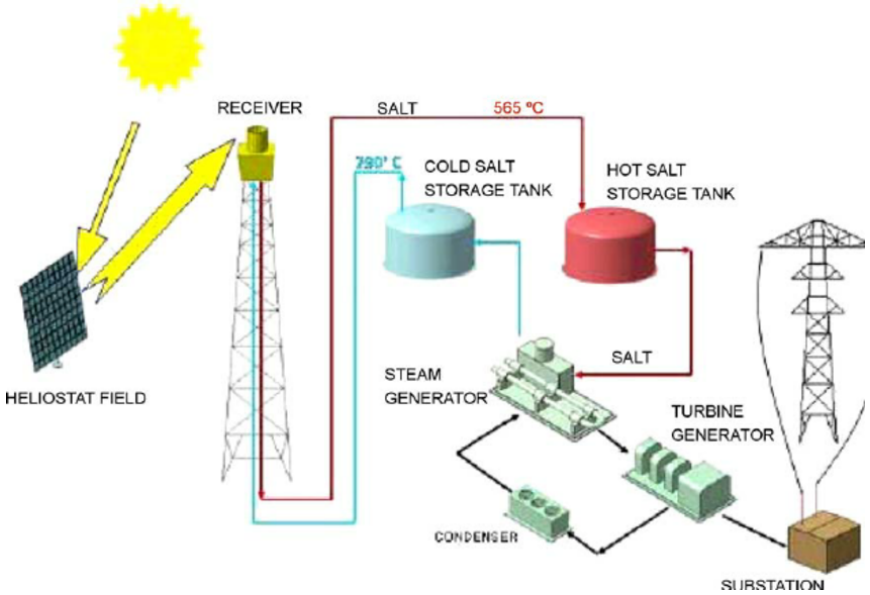
Solar-Thermal Systems with Thermal Storage



+ Thermal storage can provide 10 hrs or more of full power production
+ Power production decoupled from solar irradiation
+ Steam power generation technology well-known

Solar Power Generation Systems

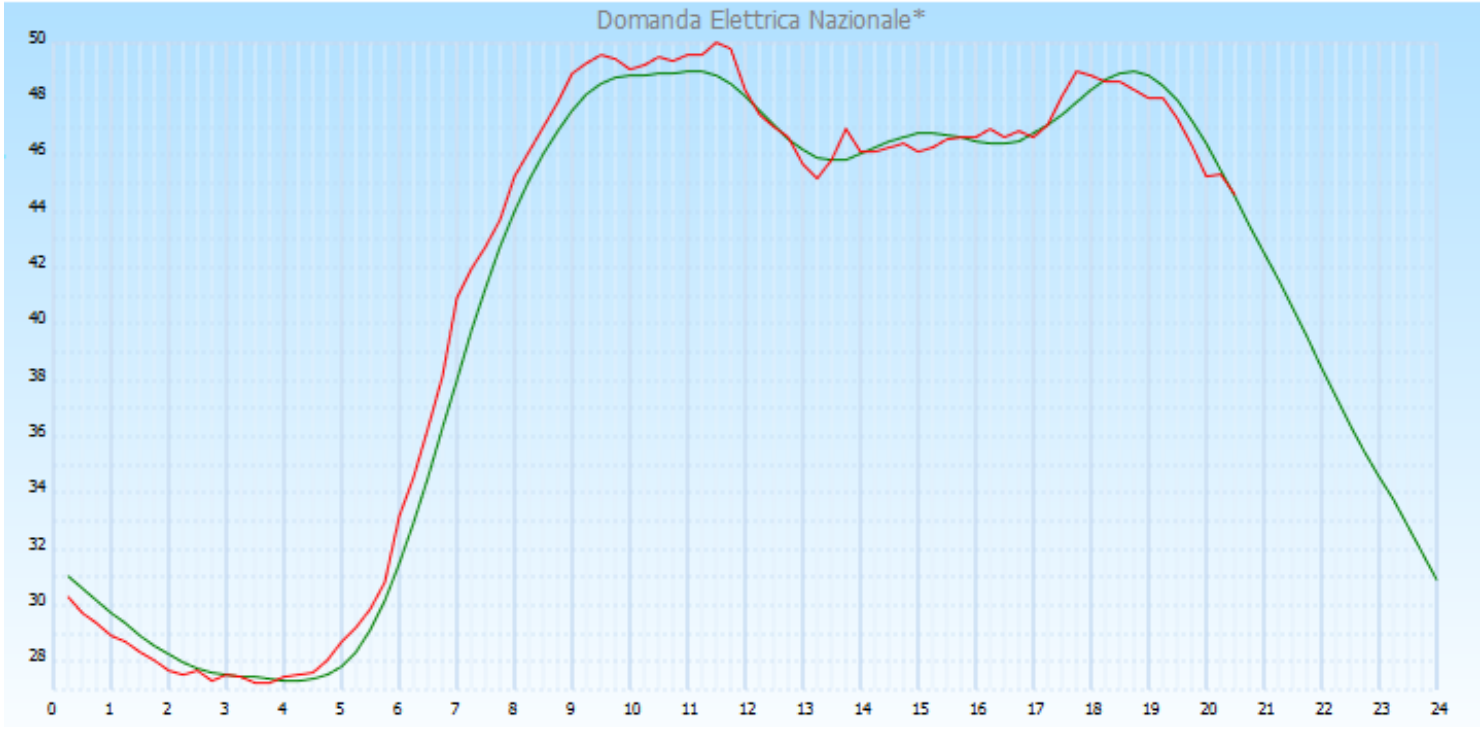
Solar-Thermal Systems with Thermal Storage

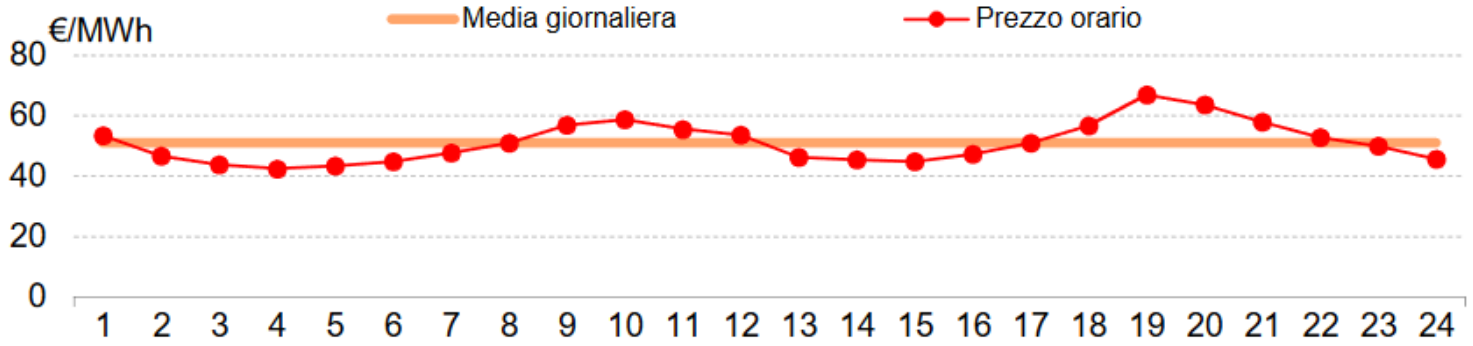


+ Thermal storage can provide 10 hrs or more of full power production
+ Power production decoupled from solar irradiation
+ Steam power generation technology well-known

- Effective system design requires the use of molten salts as working fluid
- Current designs based on flat subsidized rates (no incentives to flexibility)

Energy Markets and Prices





The PreFlexMS Horizon 2020 Project

Predictable, **Flex**ible solar power generation using **M**olten **S**alts

- Predictable
  - Advanced meteo forecast of DNI
  - Market price-based dispatch optimization
  - Exploiting thermal storage (production shifting)

The PreFlexMS Horizon 2020 Project

Predictable, **Flex**ible solar power generation using **M**olten **S**alts

- Predictable
  - Advanced meteo forecast of DNI
  - Market price-based dispatch optimization
  - Exploiting thermal storage (production shifting)
- Flexible
  - Fast plant start-up (30 min from standby to full load)
  - Fast load changes (> 10% / min)

The PreFlexMS Horizon 2020 Project

Predictable, **Flex**ible solar power generation using **M**olten **S**alts

- Predictable
  - Advanced meteo forecast of DNI
  - Market price-based dispatch optimization
  - Exploiting thermal storage (production shifting)
- Flexible
  - Fast plant start-up (30 min from standby to full load)
  - Fast load changes (> 10% / min)
- Molten Salt
  - Working fluid: 60% $NaNO_3$ 40% KNO_3 (common fertilizer)
  - Solidifies below 245 °C

The PreFlexMS Horizon 2020 Project

PreFlexMS

Predictable, **Flex**ible solar power generation using **M**olten **S**alts

- Predictable
  - Advanced meteo forecast of DNI
  - Market price-based dispatch optimization
  - Exploiting thermal storage (production shifting)
- Flexible
  - Fast plant start-up (30 min from standby to full load)
  - Fast load changes (> 10% / min)
- Molten Salt
  - Working fluid: 60% $NaNO_3$ 40% KNO_3 (common fertilizer)
  - Solidifies below 245 °C

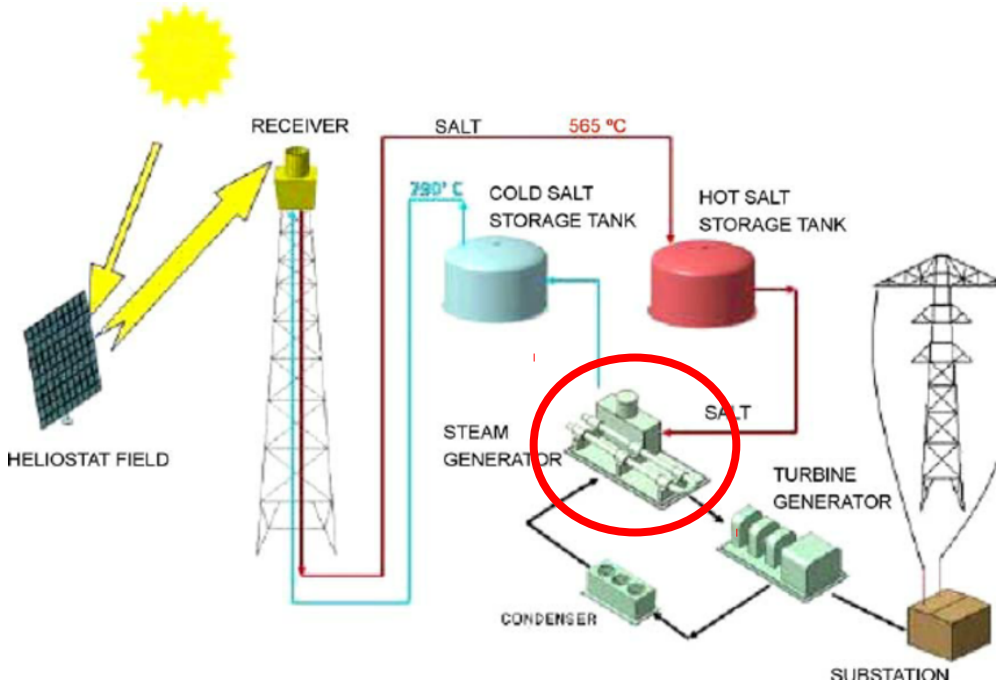
Competitive with gas-fired flexible CCPPs

The PreFlexMS Horizon 2020 Project

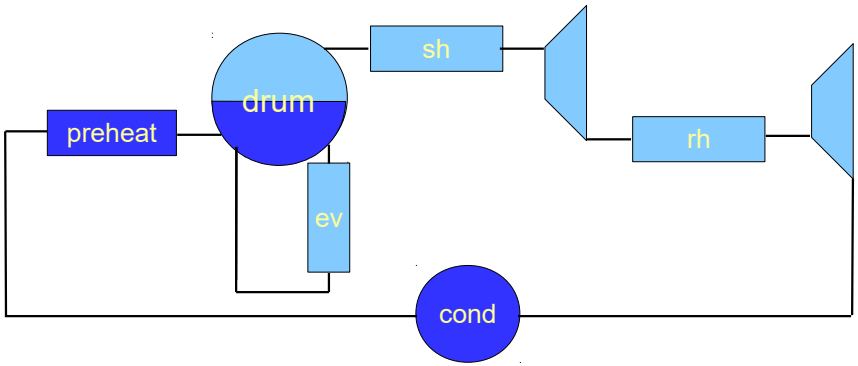
Predictable, **Flex**ible solar power generation using **M**olten **S**alts

- Main Industrial Partner: GE (formerly Alstom Power)
- 13 Consortium Partners
- Total Budget: 17.8 M€ costs, 14.3 EU funding
- Duration: 2015-2019
- Experimental demonstration on pilot-scale plant in 2019(?)

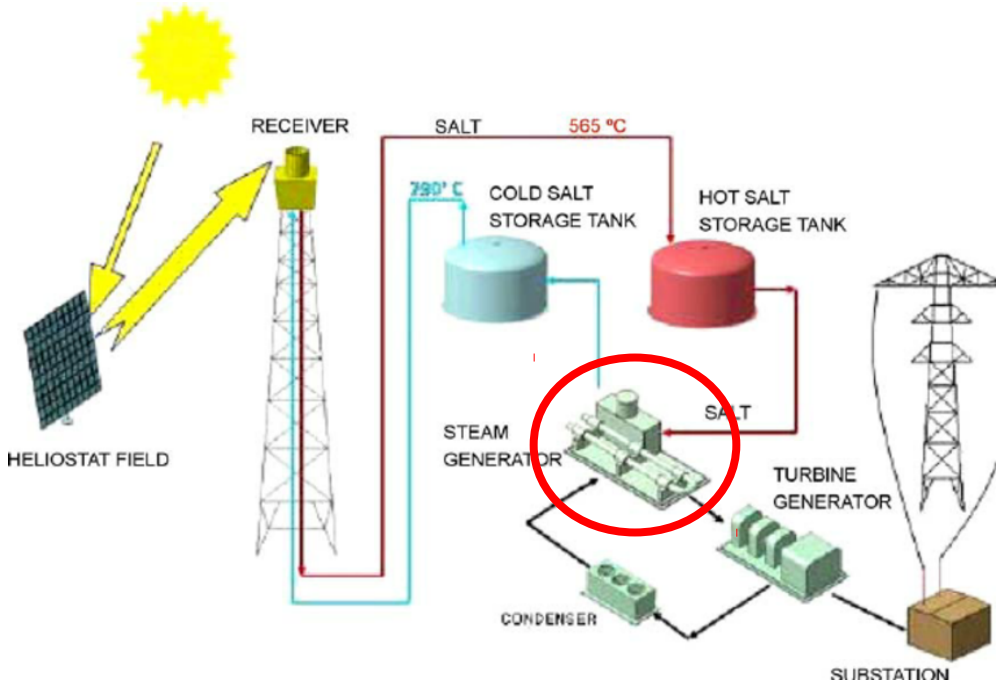
Focus of our flexibility study



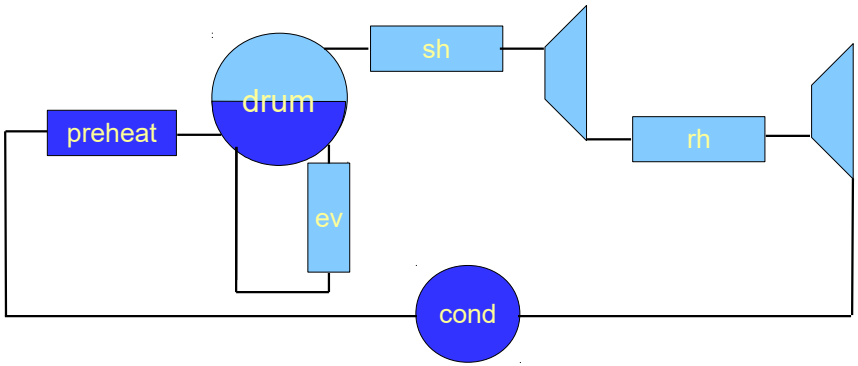
State of the art: drum-type steam generator



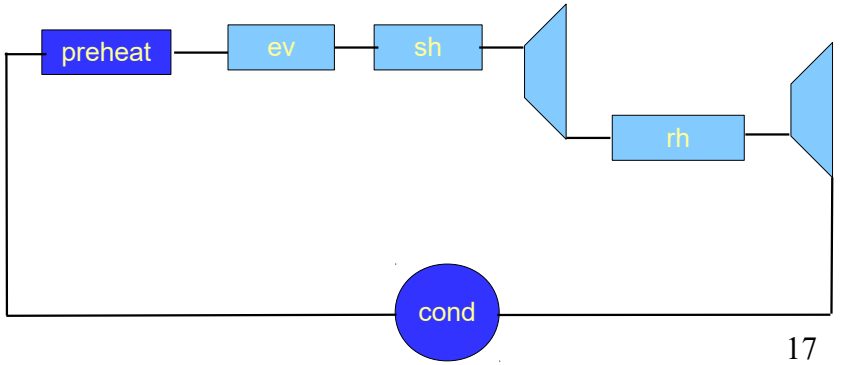
Focus of our flexibility study



State of the art: drum-type steam generator



Innovation: once-through steam generator



Challenges and research questions

- First design ever of a once-through steam generator using Molten Salt
- No previous operational experience
- Dynamic behaviour of circulation-type MS plants different
- Dynamic behaviour of gas- or coal-fired once-through boilers different

Challenges and research questions

- First design ever of a once-through steam generator using Molten Salt
- No previous operational experience
- Dynamic behaviour of circulation-type MS plants different
- Dynamic behaviour of gas- or coal-fired once-through boilers different

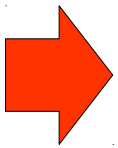
Q1: What is the dynamic behaviour of this plant?

Q2: Is this plant easy or difficult to control?

Q3: Which control strategies are more appropriate?

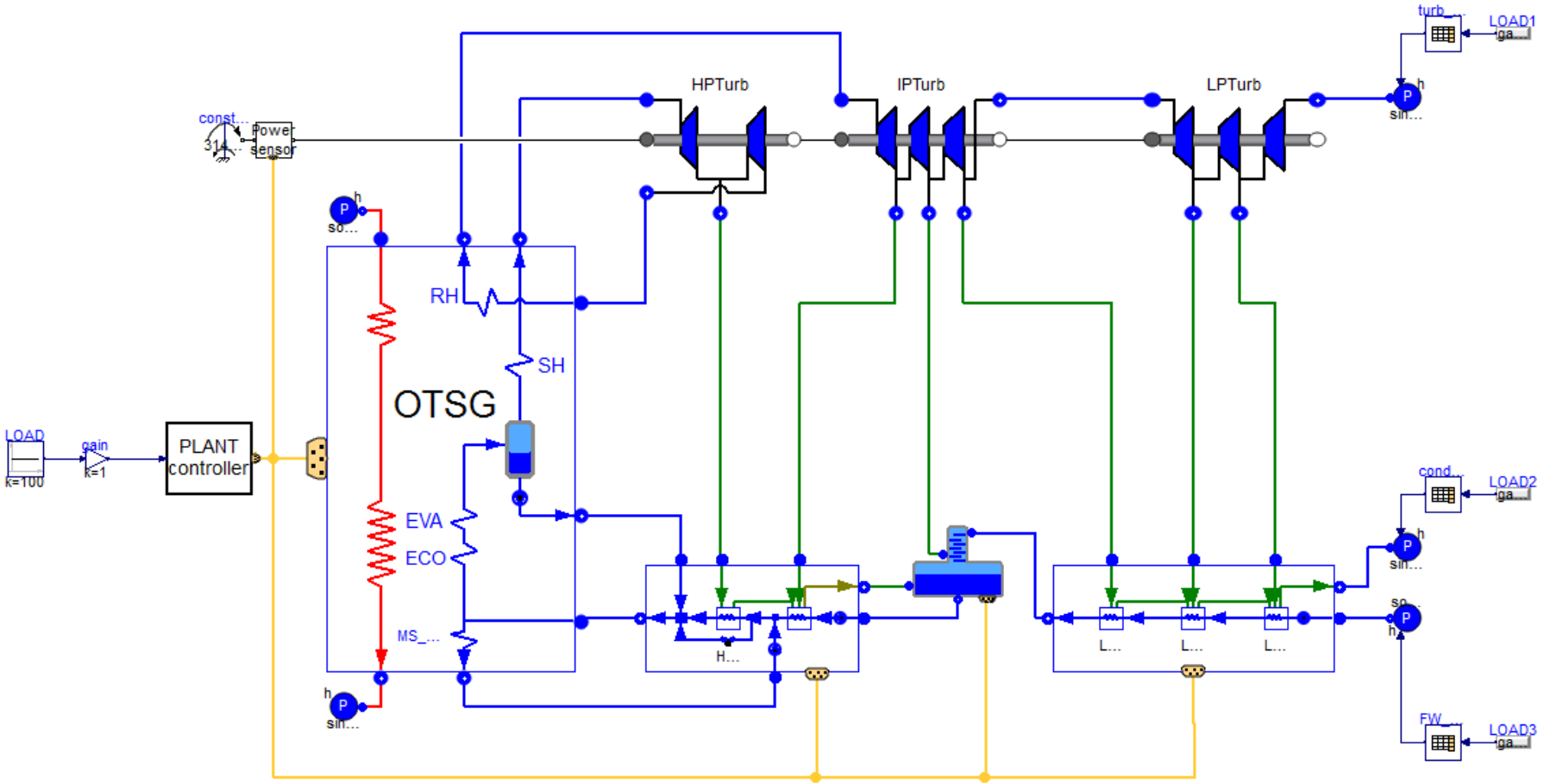
Q4: Can we start-up the plant from warm stand-by to full load in 30 mins?

Q5: What is the maximum load change rate we can afford?

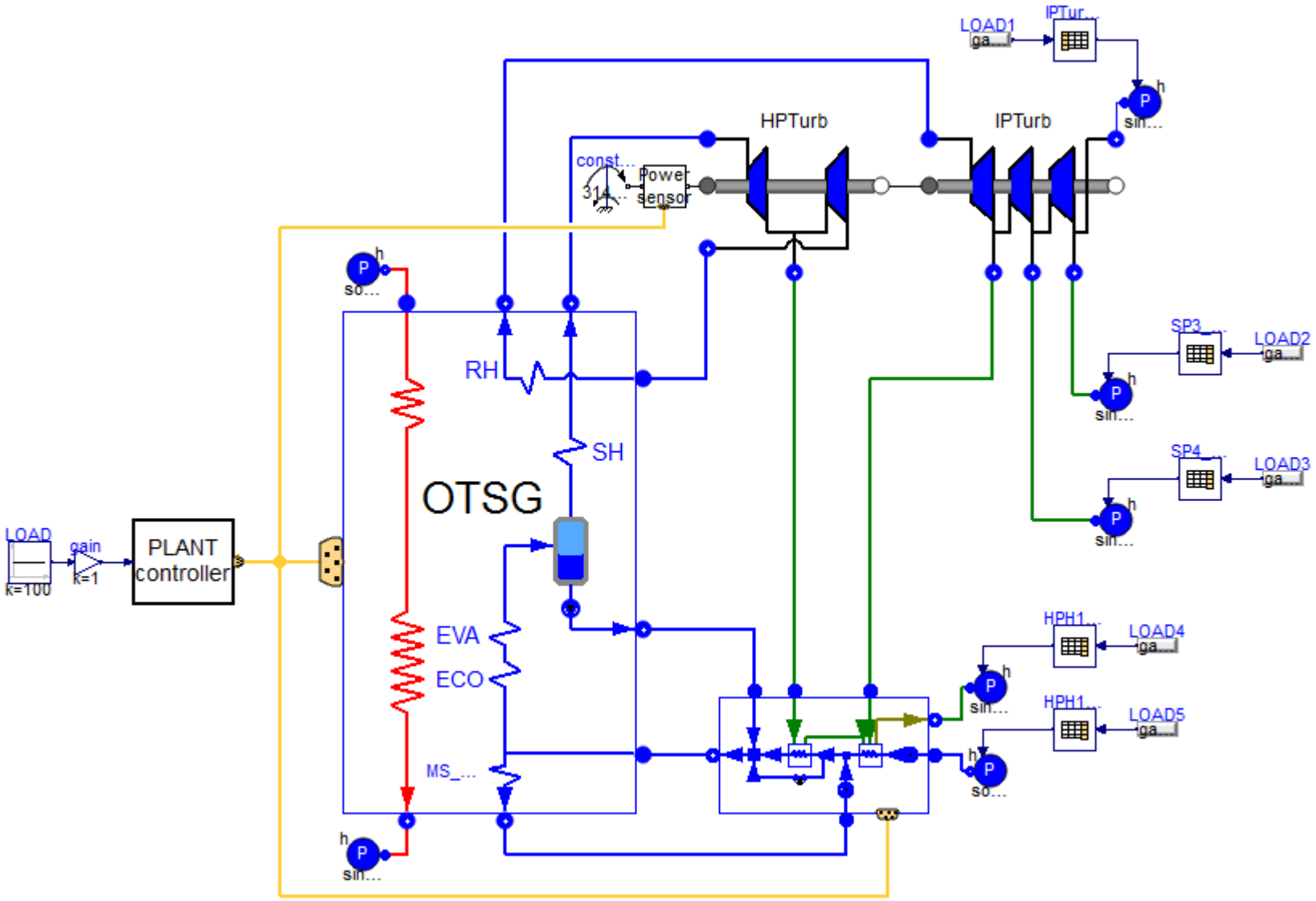


Dynamic model required

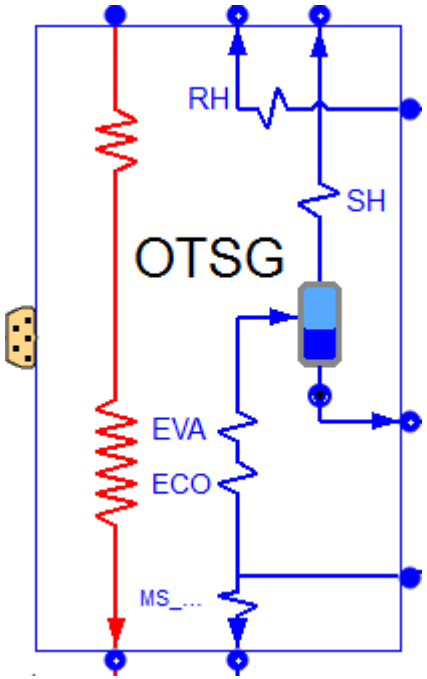
Modelica model of the PreFlexMS power block – Top view

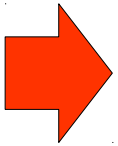


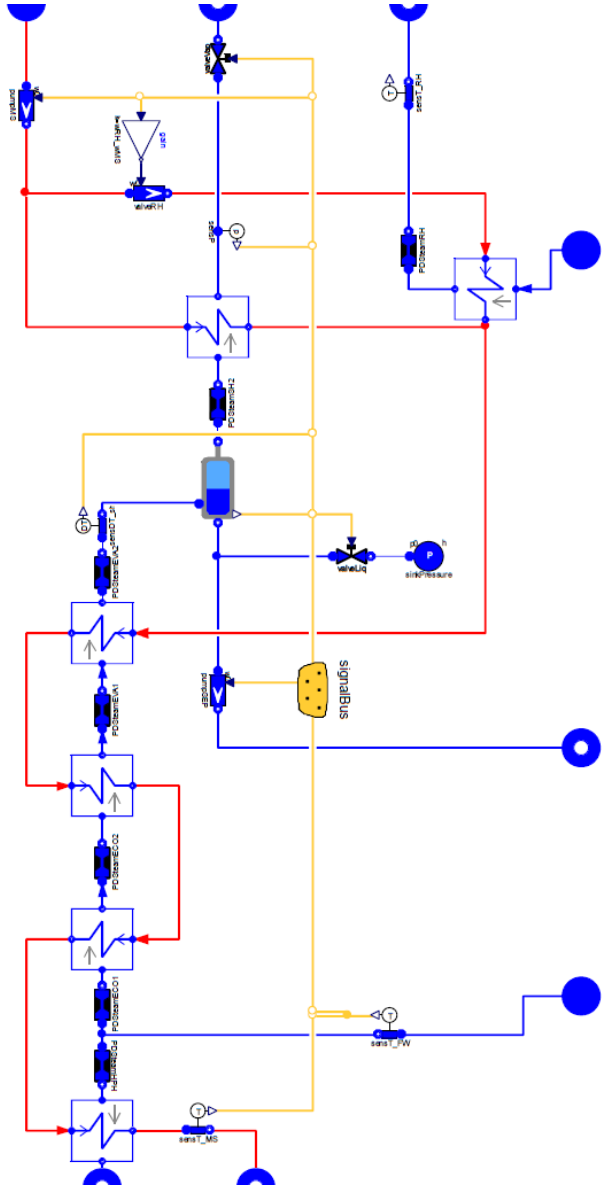
Simplified model with HP part only – Top view



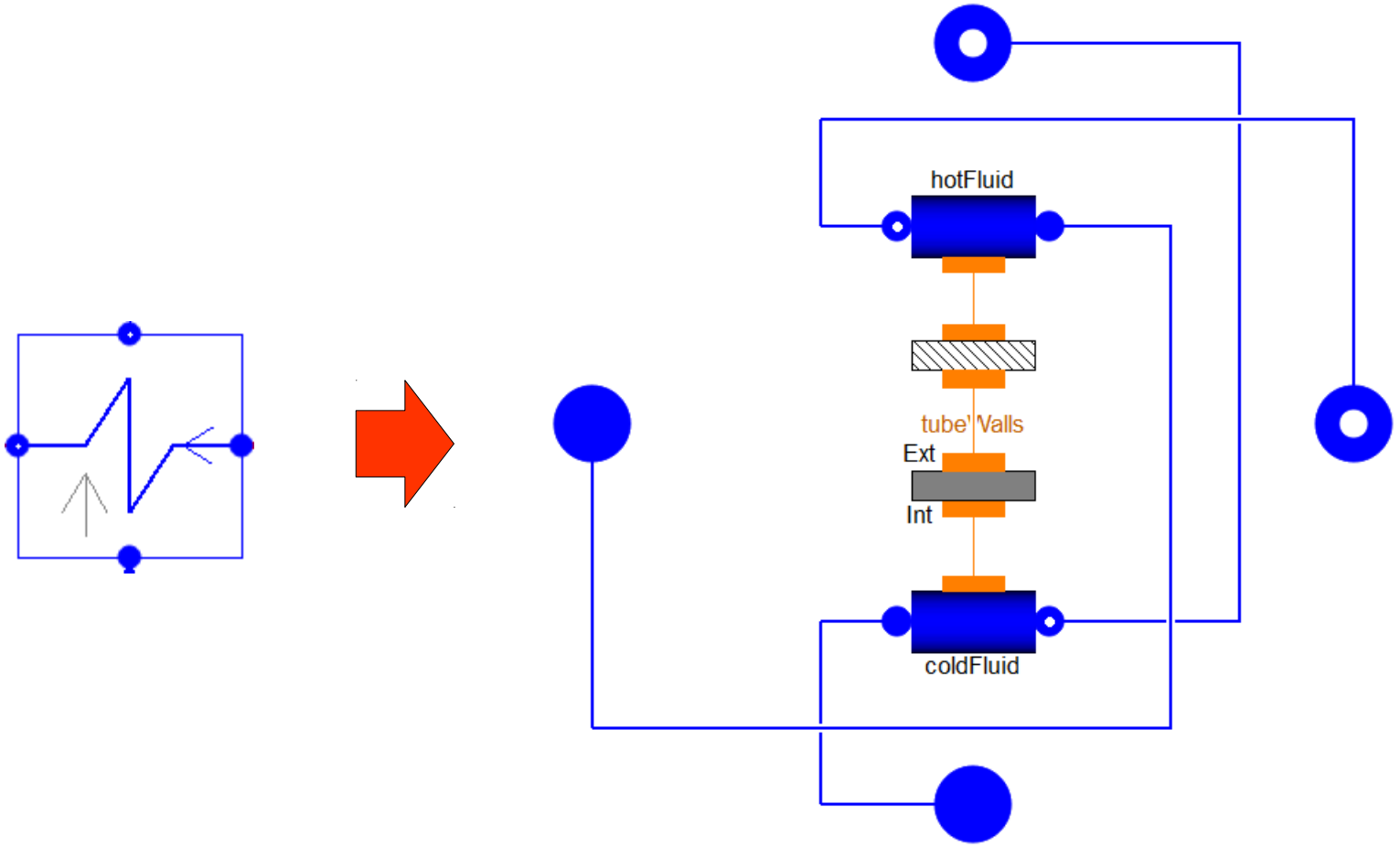
Once-through steam generator model



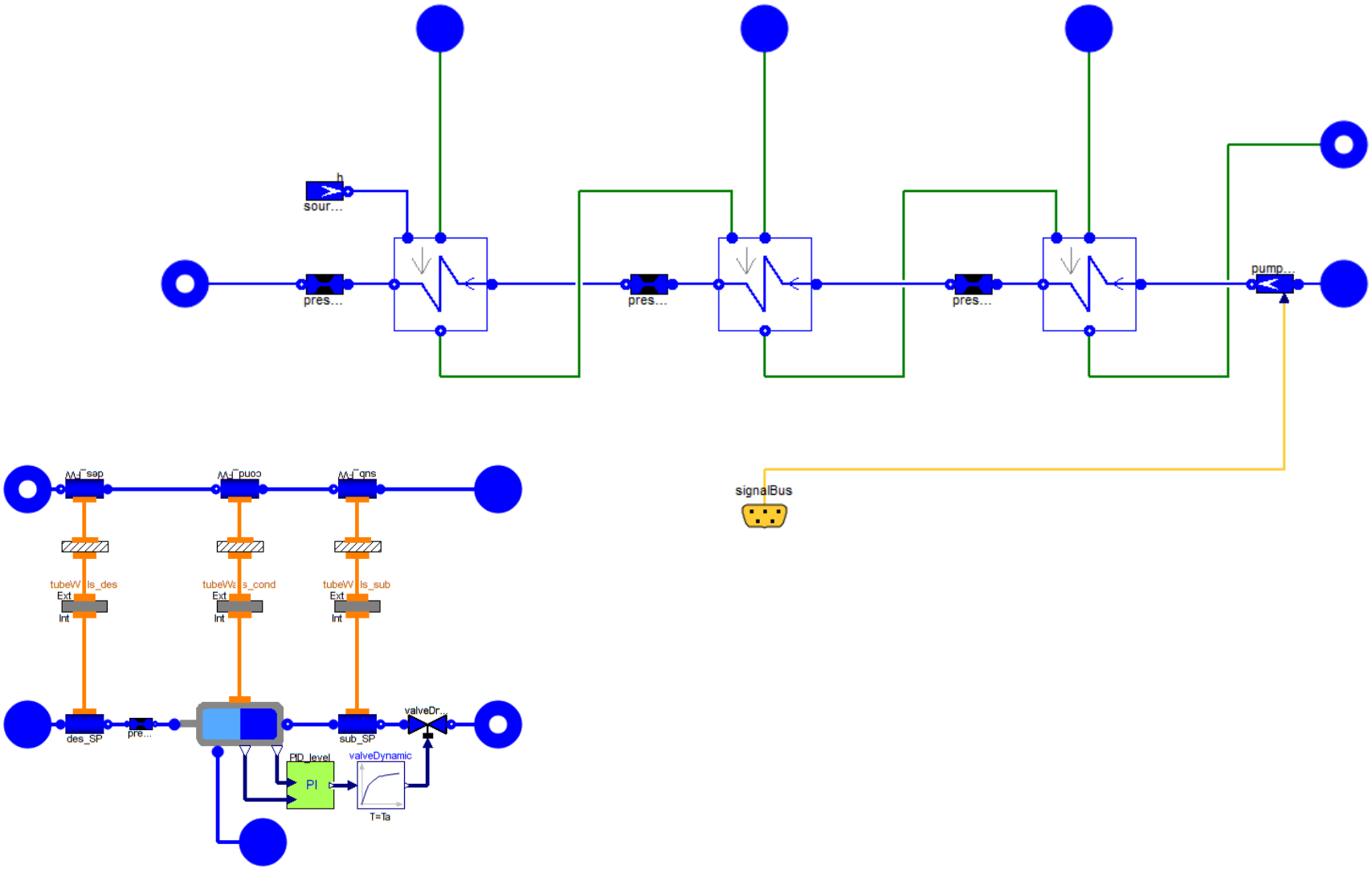




Single heat exchanger model



Preheater train model



Main Features of the models

- Full model
  - Number of equations: 9655
  - Number of states: 278
- Simplified HP model
  - Number of equations: 6974
  - Number of states: 207
- Heavy use of Modelica.Media IF97 water/steam medium model
- Highly nonlinear, very stiff
- Some flow rates are zero or near zero in certain time intervals
  → numerically challenging
- Models initially developed in Dymola in 2015
- Tested and used with OpenModelica since Apr 2016

Activity #1:

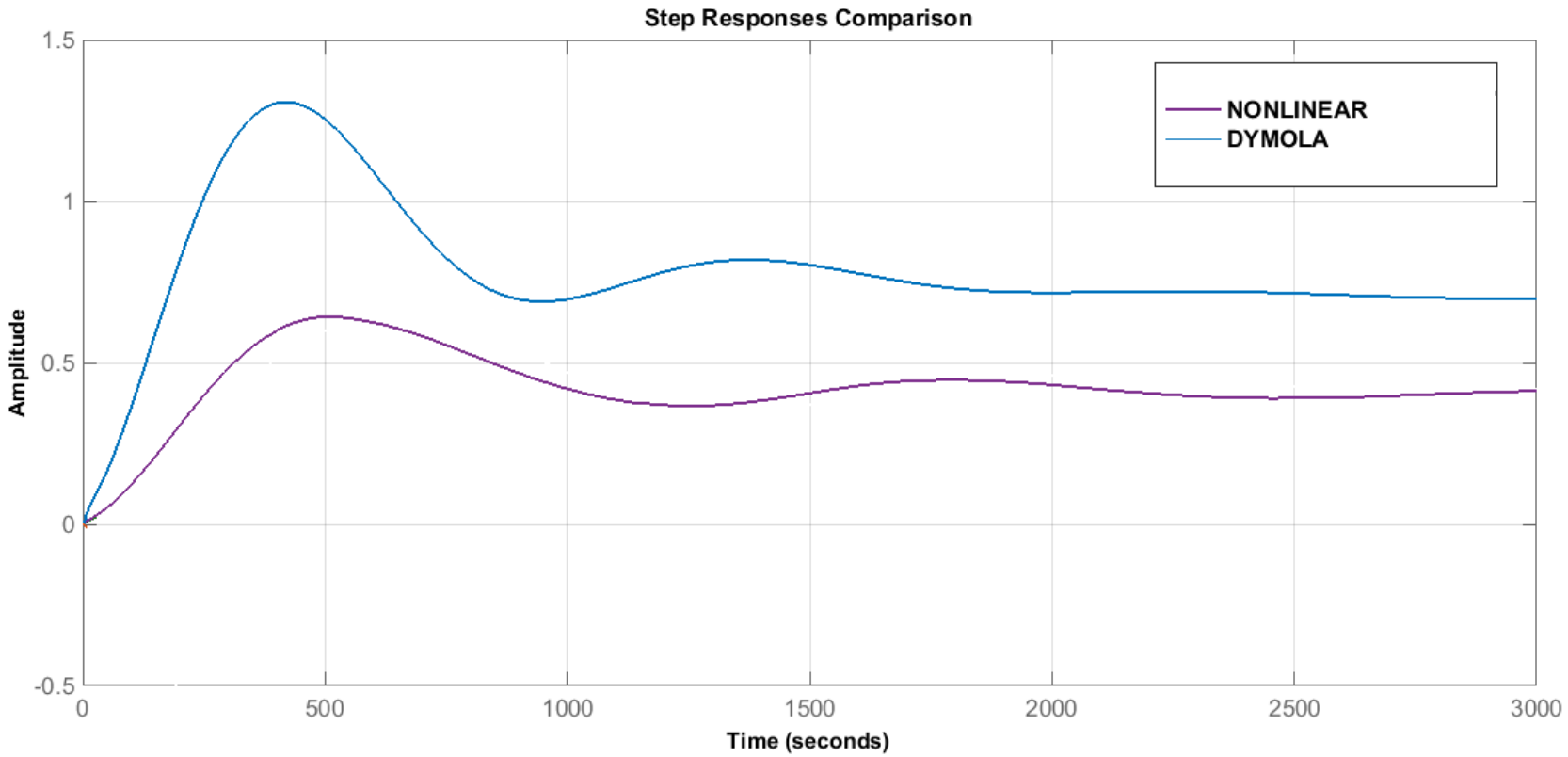
Dynamic Response Analysis for Control System Design

Outline of activities

- The model is ran to steady-state and checked against design and off-design operating points
- The model is linearized around steady-state points at different load values
- Transfer functions are computed
- Bode plots of the transfer functions are inspected and checked for nonlinear behaviour @ different loads
- RGA matrix is computed to assess controllability with decentralized PID controllers
- Transmission zeros are computed and checked for non-minimum phase behaviour → bandwitdth limitations
- PID controllers are tuned based on the transfer functions
- Advanced centralized linear MPC controllers are designed using the state-space linearized models
- Performance of designed controllers is checked by simulation

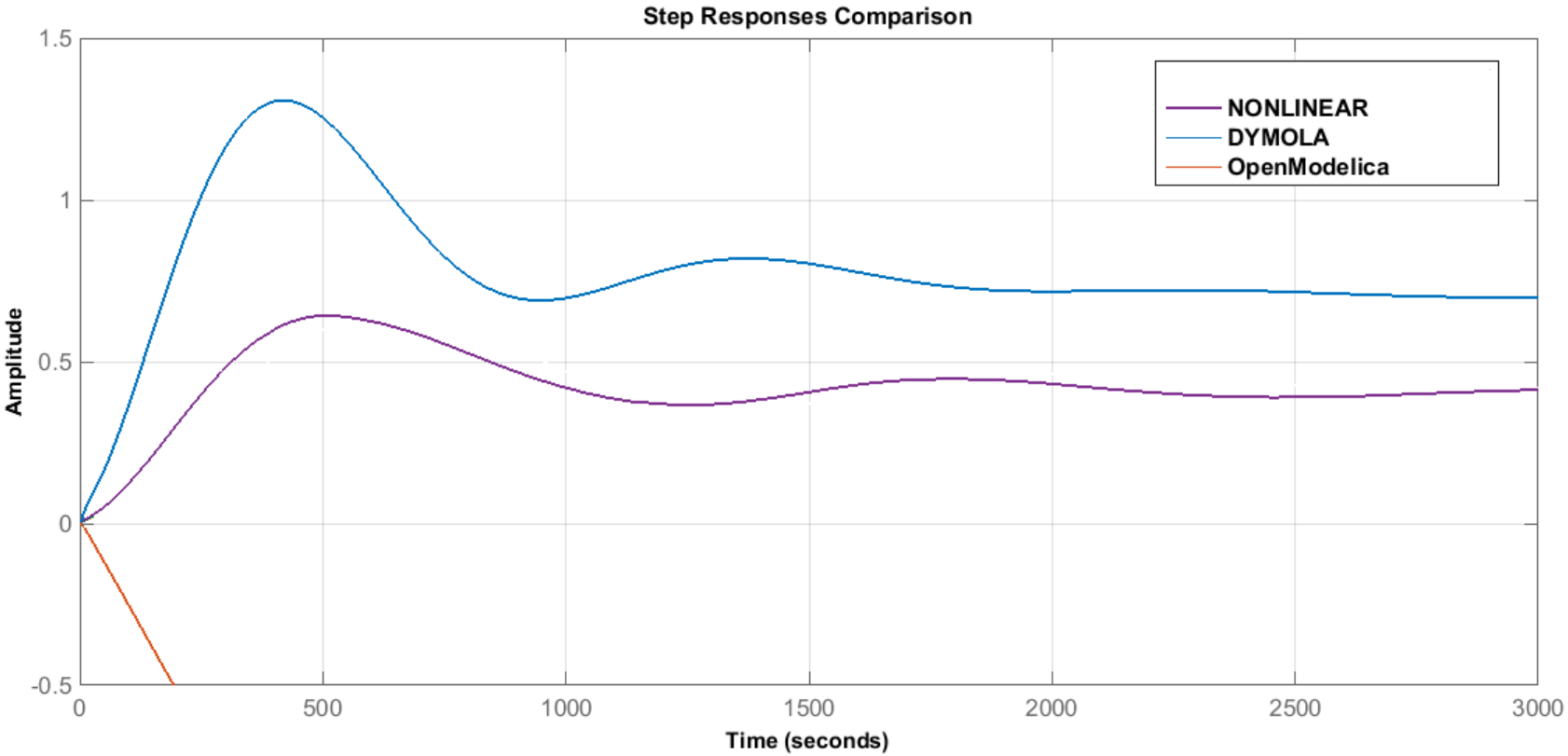
Quality of linearized models is essential!

Step response around an equilibrium point



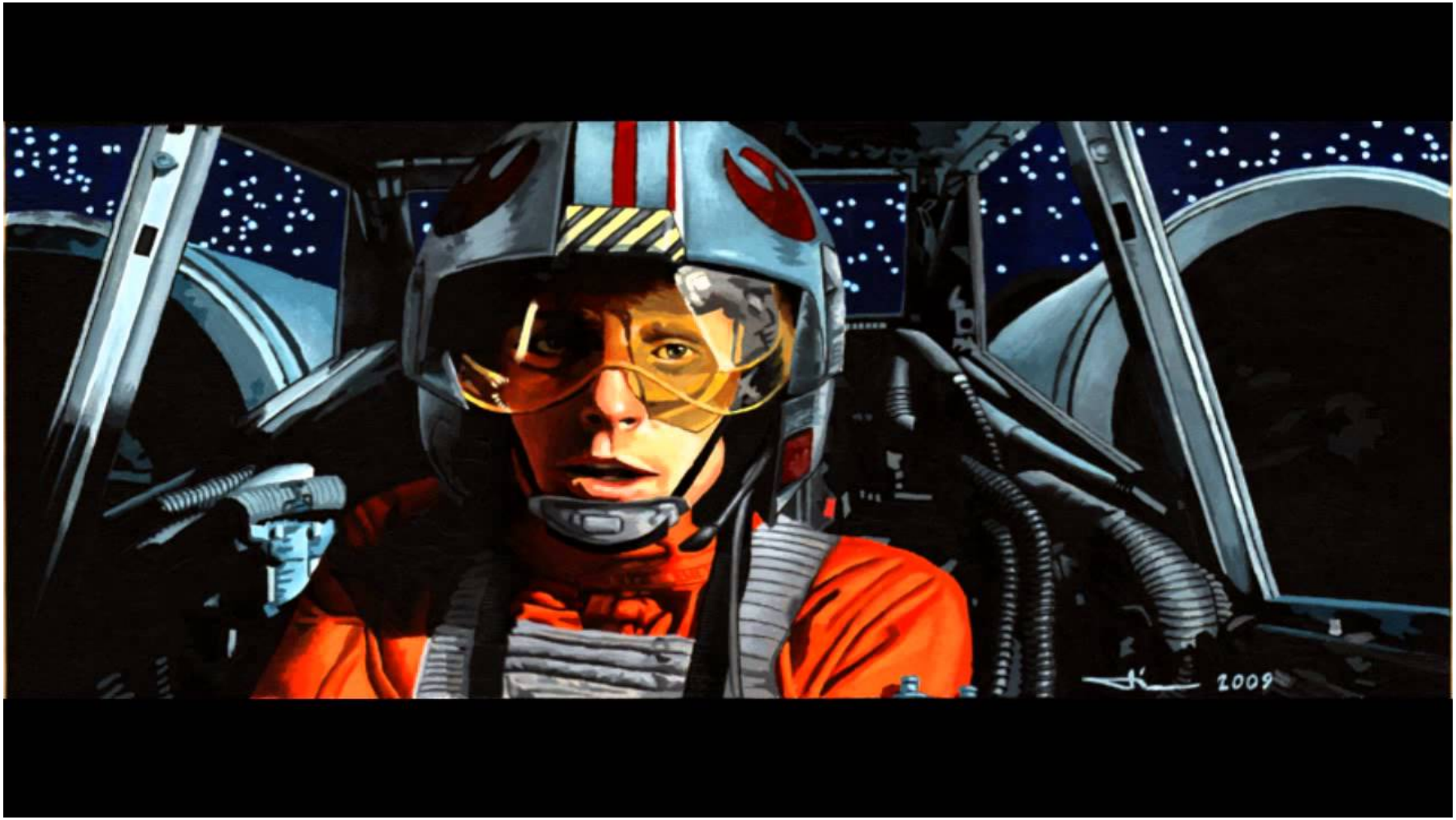
Problems with the quality of the numerically linearized model!

OpenModelica to the rescue?



Unstable linearized model!

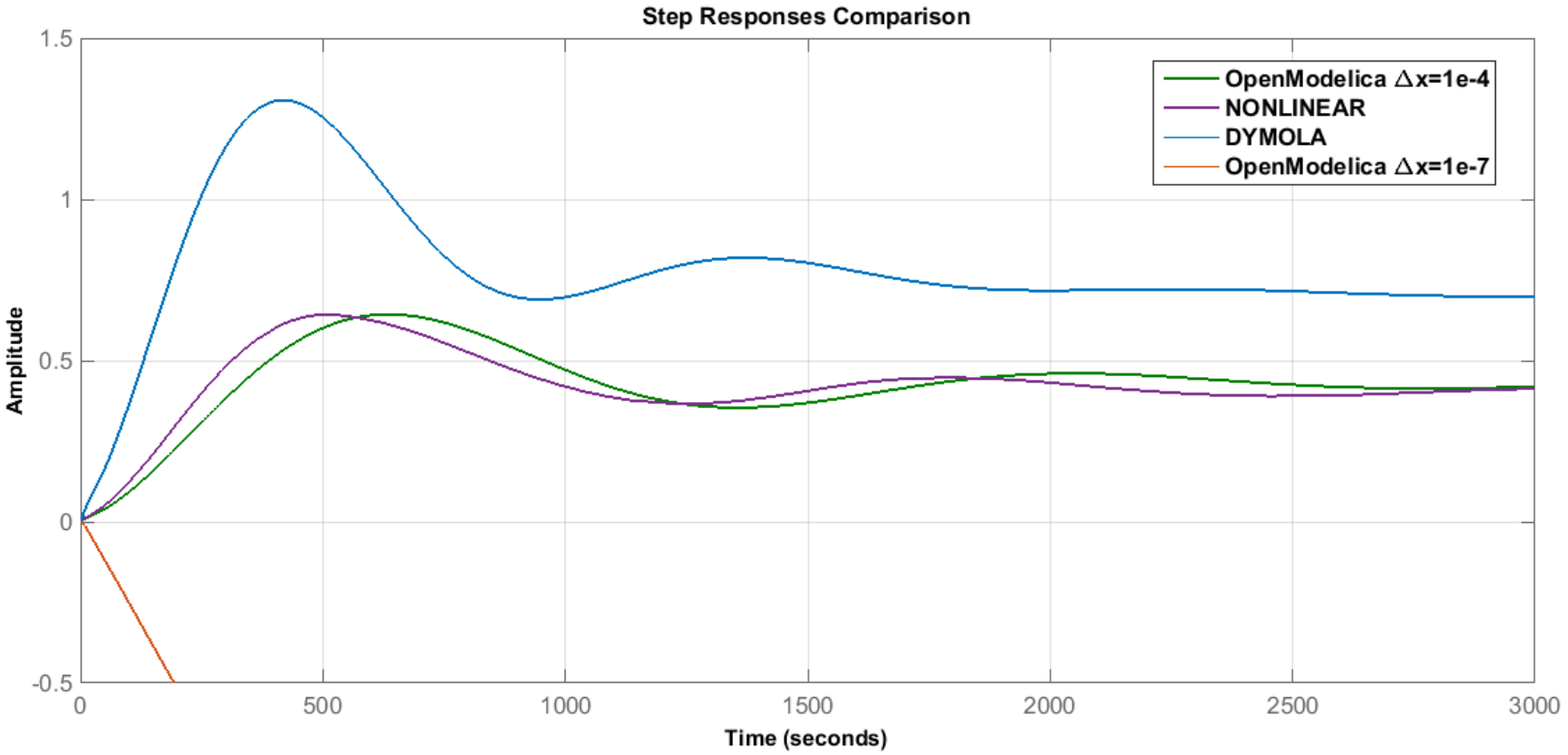
Use the Source, Luke!



Let’s get to the source of the problem

- Original version of the code
- Changed scaling
- Added parameter for deltaX

Step response around an equilibrium point



Problem solved!

All's well that end well

- Problem raised Oct 19, 2016
- Problem fixed Oct 20, 2016 (thanks Willi!)

- Advantages for research work
  - All methods are in plain sight
  - Understand exactly what the tool does
  - Improve the code according to one's need

- Maybe industrial settings prefer turnkey solutions?
  (not necessarily; hint: Rudiger Franke's work within ABB)

- What is the value of this aspect of open-source development for industrial partners of the OSMC?

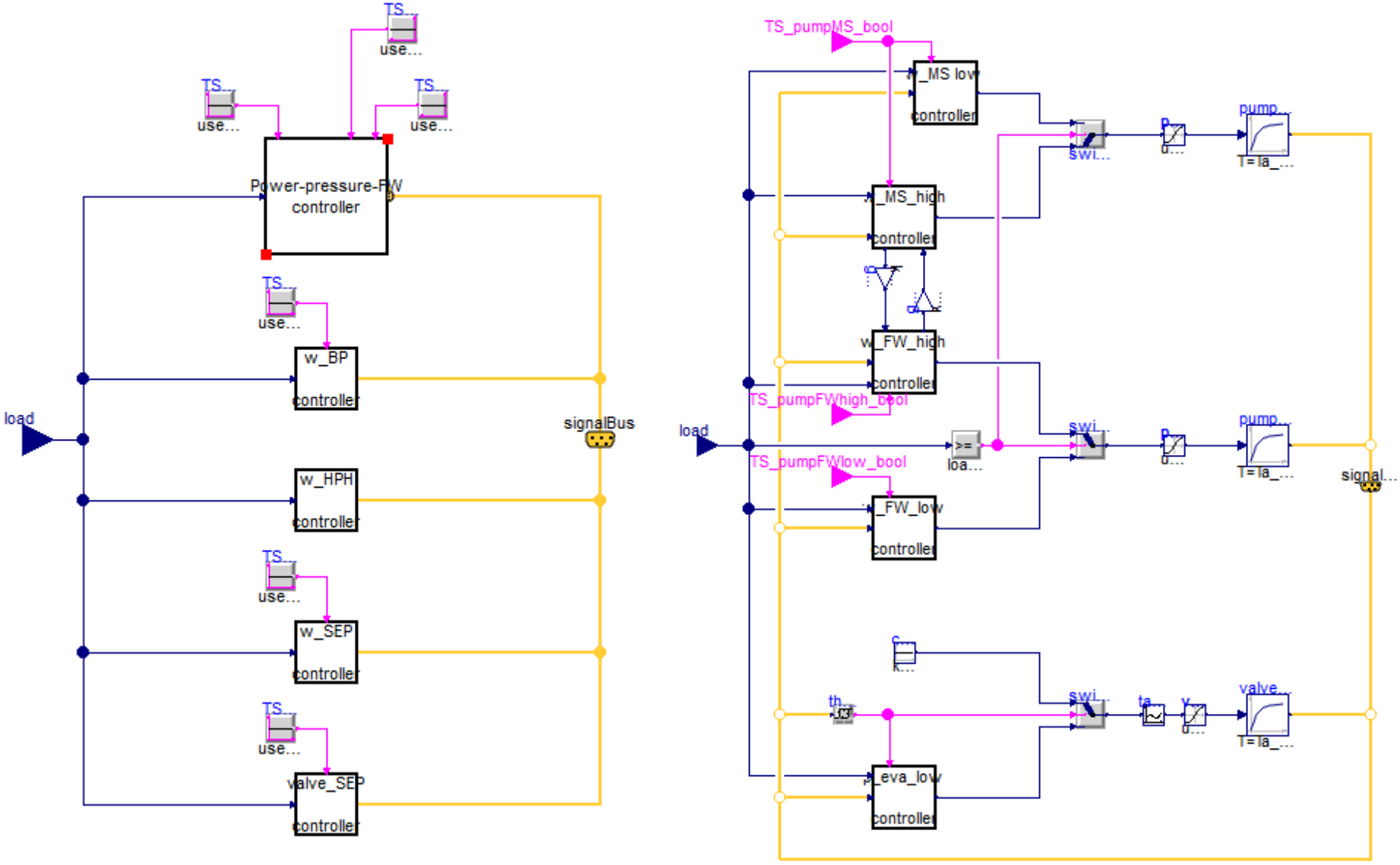
Activity #2:

Evaluation of control performance by simulation

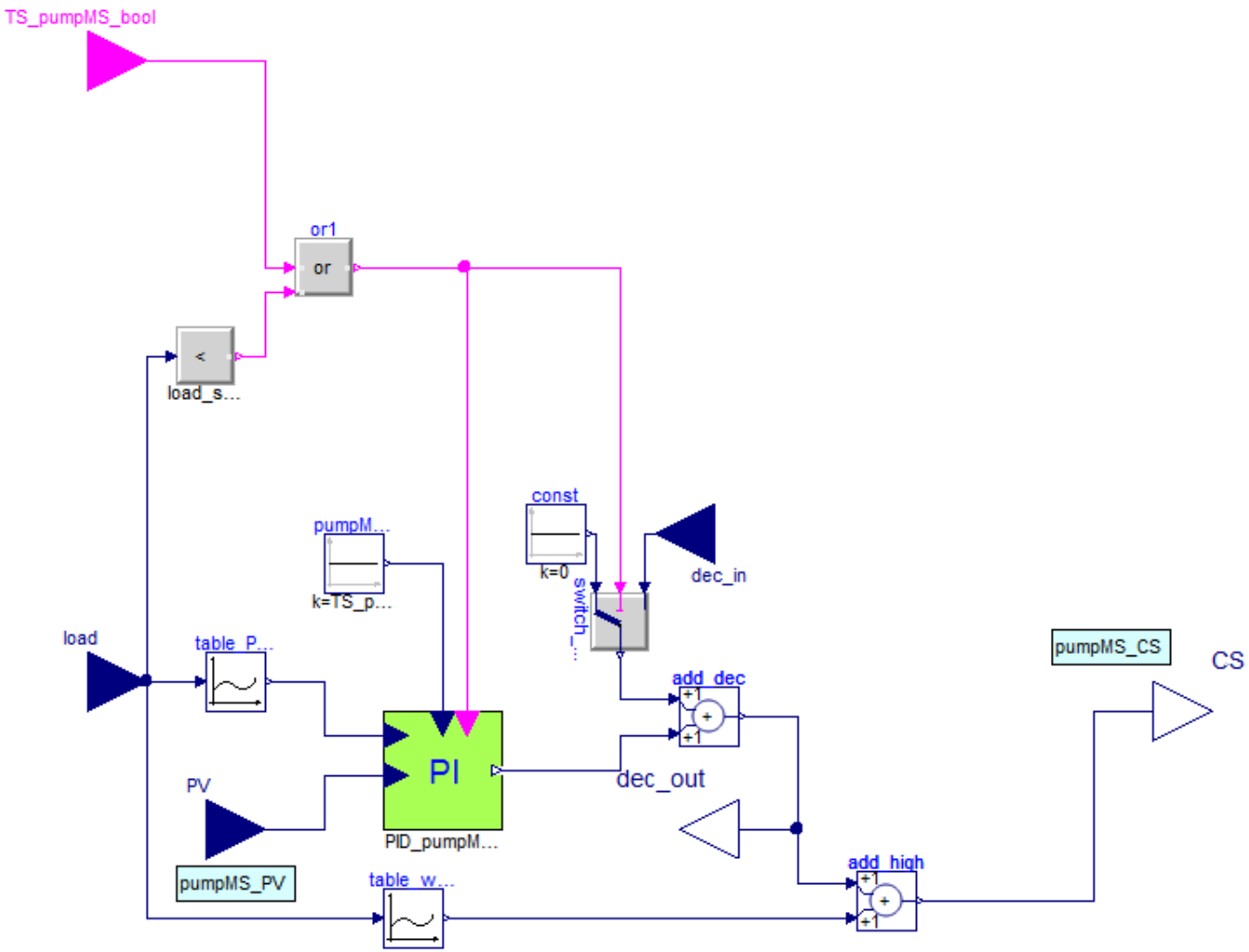
Outline of activities

- Decentralized PID-based control designed based on the results of dynamic analysis
- Control system modelled in Modelica, using the IndustrialControlSystems library (2nd library award winner @ 2012 Modelica Conference)

Control system model



Control system model - cont'd



Outline of activities

- Decentralized PID-based control designed based on the results of dynamic analysis
- Control system modelled in Modelica, using the IndustrialControlSystems library (2nd library award winner @ 2012 Modelica Conference)
- Performance is evaluated in multiple scenarios
  - Warm-keeping
  - Transition to minimum load (20%)
  - 100%-20% load change @10%/min
  - 20%-100% load change @ 10%/min
  - Other ramp load changes (dispatching)
  - Extra power supply for primary frequency control
  - etc…

Outline of activities

- Decentralized PID-based control designed based on the results of dynamic analysis
- Control system modelled in Modelica, using the IndustrialControlSystems library (2nd library award winner @ 2012 Modelica Conference)
- Performance is evaluated in multiple scenarios
  - Warm-keeping
  - Transition to minimum load (20%)
  - 100%-20% load change @10%/min
  - 20%-100% load change @ 10%/min
  - Other ramp load changes (dispatching)
  - Extra power supply for primary frequency control
  - etc…

Performance comparison

- Hardware: Intel i7-4810MQ @ 2.8 GHz 8 GB
- Solver: DASSL on causalized system, tol = 10^{-5}
- Simulation results are matching within the solver tolerance

	Dymola 2017	OMC 1.11.0 dev163 (*)	OMC 1.12.0 Dev189 (*)
Build code time [s]	12	88 (52 FE)	84 (49 FE)
Simulation time [s]	115	310	253
# of steps	3978	4310	4178
# of Jac comp.	930	956	815

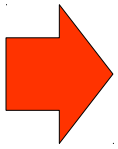
(*) using wrapFunctionCalls

Performance comparison

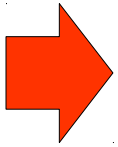
- Hardware: Intel i7-4810MQ @ 2.8 GHz 8 GB
- Solver: DASSL on causalized system, tol = 10^{-5}
- Simulation results are matching within the solver tolerance

	Dymola 2017	OMC 1.11.0 dev163 (*)	OMC 1.12.0 Dev189 (*)
Build code time [s]	12	88 (52 FE)	84 (49 FE)
Simulation time [s]	115	310	253
# of steps	3978	4310	4178
# of Jac comp.	930	956	815

(*) using wrapFunctionCalls



Acceptable for serious use



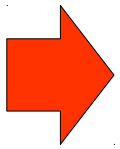
Large margins for improvement

Activity #3:

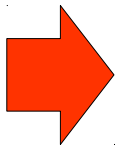
Sharing the model with other project partners

PreFlexMS partner goals

- Politecnico di Milano
  - Analysis and design of control issues
  - Validation of control performance by simulation
- AGH University of Technology (Krakow)
  - Design of machine-learning systems for performance prediction
  - Need of models to tuning and validation by simulation
- DLR
  - Complete virtual power plant model
  - Development and testing of model-based dispatch optimizer
- EC-Systems (Poland)
  - Software integration and deployment



Re-use Modelica models



Avoid extra license fees

The solution

+

OpenModelica

The solution

- Suitable interfaces added to the power block model
- FMUs 2.0 for ME generated by OMC
- FMUs embedded in Simulink system models using Modelon’s FMI Toolbox

The solution

- Suitable interfaces added to the power block model
- FMUs 2.0 for ME generated by OMC
- FMUs embedded in Simulink system models using Modelon's FMI Toolbox

- Exported models work and produce the expected results
- Some numerical problems at reduced loads
  - Maybe ode15s worse than DASSL?
  - Generated code can be improved?
  - Modelica models can be made more robust?
- No feedback from the FMU → extensive logging would be helpful
- Work currently in progress to improve the situation

Outlook and Conclusions

#1: OMC can be used for serious work

- Our experience with OMC during the PreFlexMS project was positive
- OMC can handle non-trivial power generation system models without problems
- The performance is acceptable

#2: OMC can work side-by-side with other Modelica tools

- The PreFlexMS model was initially developed in Dymola
- Minor improvement was required to *also* run in OMC
- Same code base on SVN for use with both tools
- Even for users of commercial tools, OMC can provide advantages without the need of switching to its exclusive use
  - Alternative tool for troubleshooting
  - Generation of royalty-free simulation code (e.g., FMUs)
  - Massive deployment of simulation code on the cloud for optimization purposes
  - ...

#3: OMC can be used as a royalty-free simulation code generator

- FMU/ME compiled with OMC run successfully
- FMUs compiled with OMC carry no license limitations
- The current support of FMU/CS (only forward Euler) is not enough for this kind of models
- Full support of logging in FMUs is absolutely necessary for troubleshooting

#4: Open-Source code allows to better understand problems and solve them

- We faced a (potentially fatal) numerical issue with linearized models
- Access to the code allowed to understand exactly how the current implementation worked and how to fix it
- The (small) fix was implemented and delivered overnight

#4: Open-Source code allows to better understand problems and solve them

- We faced a (potentially fatal) numerical issue with linearized models
- Access to the code allowed to understand exactly how the current implementation worked and how to fix it
- The (small) fix was implemented and delivered overnight

- Alternative model to commercial tool vendors business model
  - Source code is a trade secret
  - As little as possible is revealed about how the code works
  - Turn-key customer care by commercial tool vendors

- Discussion?

#5: Still much work to do to achieve competitive performance

- Simulations of steam-based power generation systems still 2 times slower than using Dymola
- Time to build executable simulation code is still about 7 times larger than using Dymola
    - New front-end will reduce the factor to 3-4 (June? Sept.?)
    - Further optimization of the code generation process required
- The wrapFunctionCalls optimization is crucial for good performance it should be activated by default in 1.12.0
- GUI still not up to the task
    - Too slow when opening models
    - Some problems with rendering of diagrams
    - No replaceable medium support yet
    - No hierarchical model editing yet

Acknowledgements

This work was supported by the European Commision, H2020 program, under Grant 654984

Many thanks to all the OMC developers, particularly Willi Braun, for their support

Thank you for your kind attention!