COMSOL
CONFERENCE
2019 CAMBRIDGE

Co-Simulation of Dynamic Energy System Simulation and COMSOL Multiphysics®

A. Dahash, F. Ochs and A. Tosatto

Unit of Energy Efficient Buildings, Department of Structural Engineering and Material Sciences, University of Innsbruck, Innsbruck, Austria.

Abstract

The efforts to integrate and/or expand the renewables into the overall energy scheme are drastically increasing in Europe and worldwide. This integration might somehow alter the energy scheme resulting into shortcomings (e.g. security of supply is not met). Out of the technologies, seasonal thermal energy storage (TES) systems to bridge the gap between winter heating demand and solar heat availability in summer. Yet, seasonal TES cannot be easily integrated into block and district heating (DH) systems because of a wide list of parameters and constraints influencing the integration and operation of this technology. Further, given the large-scale volume for such systems in order to fulfill the seasonal tasks, the investment cost is also a critical player that holds this technology from being experimentally examined. Therefore, numerical simulation-driven assessments and optimizations are of importance for investigation these complex systems.

In this context, COMSOL Multiphysics® is used to develop a numerical model for a large-scale TES in DH system. This model investigates the thermo-hydraulic behavior of the storage medium under different operation schemes. Whereas a dynamic simulation tool (Modelica/Dymola) is undoubtedly needed in order to capture the dynamics of the entire system. Therefore, it was used for implementation of DH system. Consequently, co-simulation approaches arise as a promising technique to couple both simulation tools.

In this research, the authors highlight the limitations as well as the opportunities of COMSOL Multiphysics® in being coupled to other dynamic simulation tools that are widely used in industry and research of energy systems (e.g. TRNSYS, Modelica/Dymola, Simulink etc.). Then, the authors discuss the coupling of COMSOL® to Modelica/Dymola tool to run a system simulation in which STES is developed in COMSOL® and the system is modeled in Dymola. Moreover, this work pinpoints the research needs, existing shortfalls and challenges needs and, then, the promising approaches.

Introduction

Towards the ultimate goals of efficient, sustainable and decarbonized energy system scheme, the integration of renewables (e.g. solar energy, geothermal) is strongly required in order to substitute the conventional fuels [1]. Therefore, the European Union has supported various actions to expand and/or introduce the renewables in the system enhancing the activities against climate change and the shortage of fossils. Nevertheless, most renewables come naturally characterized with a major shortfall shown in the intermittent pattern. For example, solar energy as one of the prominent and leading renewable energy sources fluctuates on daily and seasonally patterns. Thus, this intermittent behaviour makes those renewables being identified as undispatchable sources. In other words, renewable energy sources might be available when and where they are not required. Therefore, the integration of those sources might alter the overall energy scheme [2]. Given the risks (e.g. security of supply constraint is not met) that might arise during real experiments, simulation-based analyses found its place favourably in both of academia and industry for the promotion of energy systems.

Down to the fact that there exists a wide range of relevant aspects for the planning, design and assessment of energy systems; therefore, there exist many tools to examine various aspects and optimize the decisions. For instance, while examining a renewable-based district heating (R-DH) system, major emphases are oriented to the system capacity, optimal scheduling of heat-supply units, water flow in R-DH network, thermal losses, pumping costs and etc. For the system capacity and optimal scheduling of supply-units, one-dimensional or plug flow modelling tools might be sufficient to cover both aspects delivering better understanding for the required capacity and optimizing the control strategies for better scheduling. Therefore, dynamic system simulation tools (e.g. Modelica/Dymola, TRNSYS, Simulink) are usually used for this category of tasks. Nevertheless, those tools are fashioned to investigate the energy flow and not the entire multiphysics and, thus, they can be inaccurate for estimating the thermal losses over the R-HD grid and, hence, more challenging to optimize the pipelines design. Thus, Multiphysics tools (e.g. COMSOL Multiphysics) based on finite element (FE) method ideally tailored for such task as they provide the user with deep understanding.

To capture the ultimate advantages of simulation-driven assessments, it is important to develop new techniques to effectively simulate the interactions between the different components of the investigated system. Thus, co-simulation has received great attention in the last few years [3] [4] [5].

What is Co-Simulation?

In a co-simulation environment, the subsystems models are interconnected with each other at their behavioural levels though the models given in different tools. To execute a co-simulation case study, it is crucial to set a co-simulation scenario and an orchestrator algorithm. The importance of the c-simulation scenario arises as it interconnects one or more simulation units. Accordingly, it illustrates the inputs and outputs for each simulation unit. Herein, a simulation unit is seen as a black box that consumes inputs to produce a behaviour. In order to produce the behaviour, each simulation unit should be equipped with the following:

- A *dynamic system* which is, in an abstract way, a model with a set of assumptions;
- A *simulator* (i.e. solver) which is an algorithm to compute the behaviour of the system;
- An *input approximation* in which the model inputs are approximated over time by the solver, and
- *Inputs and outputs reactivity*, which identifies the inputs sent to each simulation unit from the orchestrator.

It is important to underline that the orchestrator is responsible to couple the different simulation units. Therefore, the orchestrator also transfers the data from the outputs of one simulation unit to the inputs of another following a co-simulation scenario. Hence, an orchestrator also regulates the development of the simulated time in each simulation time.

For a co-simulation scenario, there exist two typical orchestration patterns, one is the *discrete event* (DE) co-simulation, whereas the other is *continuous time* (CT) co-simulation. In a DE co-simulation, the data exchange takes place at prescribed events. Thus, DE co-simulation exhibits the following:

- *Reactivity* in which the events react instantaneously to the external stimuli and, consequently, the events change their state, and
- *Transiency* in which the events can lead other events to occur instantaneously and, consequently, the system can change its state multiple times.

Whilst in in CT co-simulation, the simulation units exchange the data continuously and, hence, the state of a simulation unit evolves continuously. The combination of both orchestration patterns is called the *hybrid co-simulation pattern*. Further information about co-simulation can be found in [6], [7] and [8].

Figure 1 presents a co-simulation scenario in which two simulation tools are coupled, whereby two solvers are separately used for each tool. Both tools operate simultaneously and exchange the data at the desired synchronization time steps.

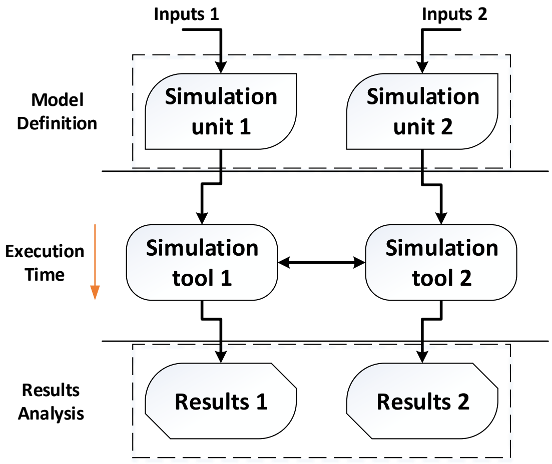


Figure 1. Schematic representation of data flow in co-simulation scenario for two simulation tools.

Co-Simulation in Energy Systems Research

In the framework of the international project “Giga-Scale Thermal Energy Storage for Renewable Districts”, an ultimate milestone is to set the planning guidelines for each construction type of seasonal thermal energy storage systems (i.e. hot-water tanks and pits) which volumes range between (100,000 m^3 and 2,000,000 m^3). Therefore, simulation-based analyses play a key role to provide the optimal planning layout and reduce the risks with maintaining an acceptable economic feasibility. Thus, TES models were implemented in COMSOL Multiphysics to assess the influence of different parameters (e.g. construction type, TES geometry, groundwater influence, ground thermal conductivity, etc.) on TES performance to derive Go/NoGo flowcharts for TES construction. However, it is found this evaluation focuses only on the component level and assumes standard ideal profiles for district heating system, whereas in reality this is not the case. Therefore, the evaluation of TES within a system simulation is highly significant and helps to examine, for example not limited, the influence of groundwater velocity on the overall system performance. Thus, a dynamic system simulation tool (i.e. Modelica/Dymola) was utilized to model the DH system. Then, it is found strongly important to couple both models; in fact, it is rather coupling both simulation tools (i.e. COMSOL Multiphysics and Modelica/Dymola) forming the so-called “co-simulation”.

Figure 2 illustrates the co-simulation scheme between the TES model and the R-DH model. TES model consists of TES envelope filled with water (hot during charging and storage phases, cold during discharging and idle phases), surroundings (ground and groundwater). Whereas R-DH model consists of different units: renewable-based heat sources, buffer storage (BS), heating backup unit (HBU) and the end-users (load). The red and light blue lines represent the supply and return sides, respectively.

The interaction between both models takes place during charging and discharging processes during which both models exchange data (namely flowrates and temperatures).

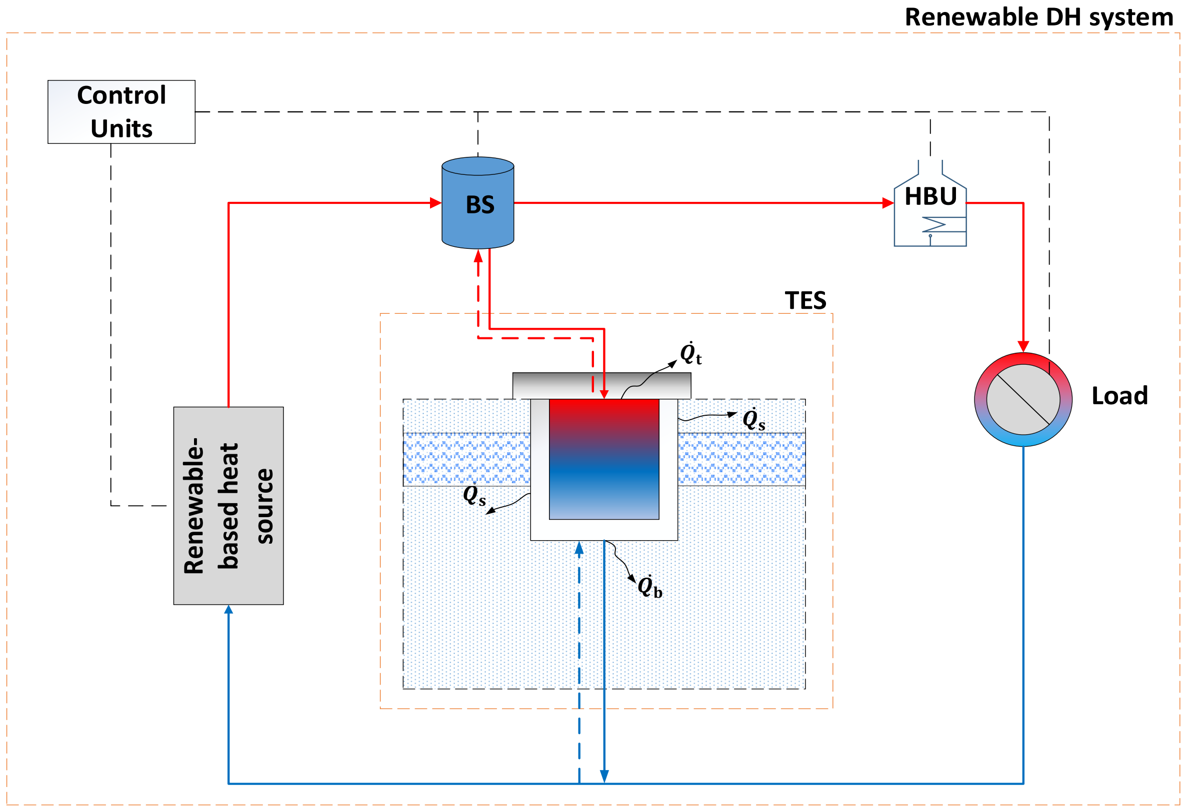


Figure 2. Schematic representation of a standard R-DH system whereby the generation (heat sources, buffer storage and back-up heating unit) and demand (load) with the control unit developed in Modelica/Dymola, whilst TES and surroundings developed in COMSOL Multiphysics.

Another example for the co-simulation is the implementation of TES model in Modelica/Dymola and the modelling of the ground in COMSOL Multiphysics. This approach also requires a robust co-simulation between the tools in order to capture the entire dynamic behaviour of TES. This approach is shown in Figure 3 and it presents a co-simulation whereby TES model (in Dymola) sends the thermal losses to the ground model (in COMSOL) and; in return, the ground model updates TES model with ground temperature.

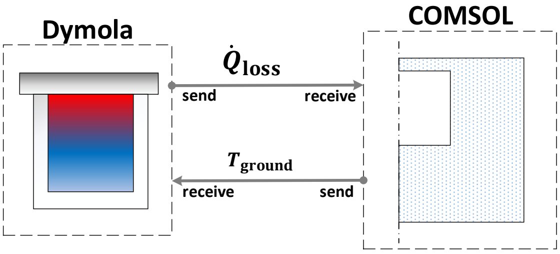


Figure 3. Possible co-simulation scenario between TES model in Dymola and ground model in COMSOL.

Therefore, it is important to explore the connectivity between both tools and other tools. Modelica/Dymola as one of the widely used tools in energy simulation, it has several connections to some other tools. It exhibits a capability to connect through function mock-up interface/function mock-up unit (FMI/FMU). However, COMSOL Multiphysics does not support this feature. Figure 4 illustrates the interfacing between COMSOL Multiphysics and some other tools.

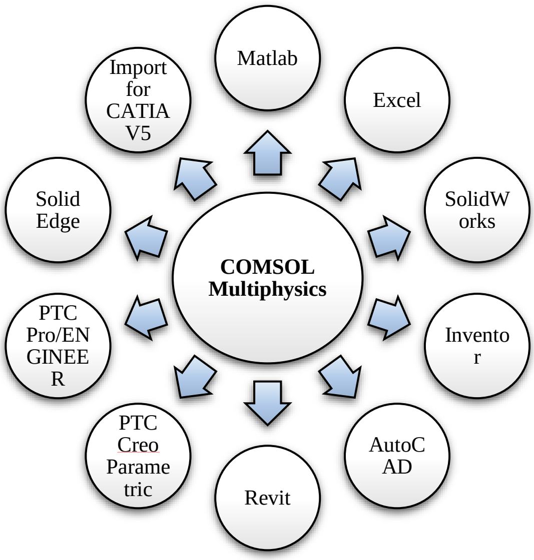
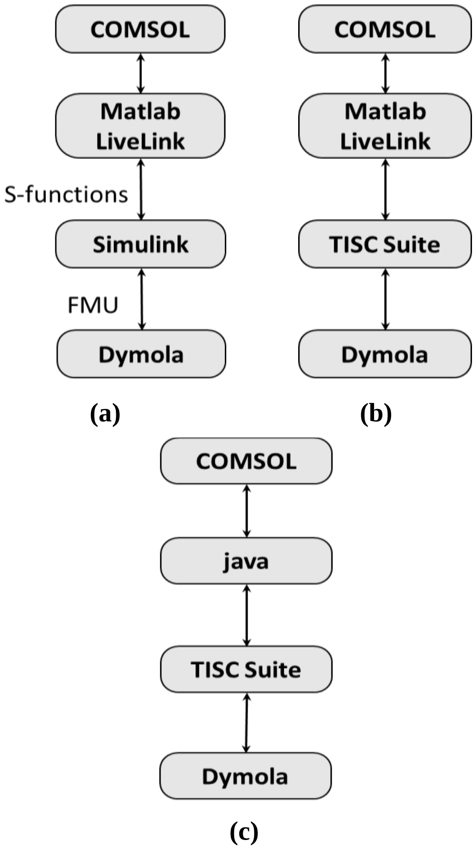


Figure 4. COMSOL Multiphysics connectivity to other tools.

Co-Simulation with TISC Suite

Apparently, Modelica/Dymola is not one of the tools that its interfacing is supported. Nor TRNSYS, which is also comparable to Dymola, was supported. Therefore, it was crucial to figure out some other workarounds between COMSOL Multiphysics and Modelica/Dymola in order to avoid shifting to other tools and start modelling from scratch again. Figure 5 reveals possible options with TISC Suite to obtain a co-simulation.



(a) **(b)**

(c)

Figure 5. Co-simulation options for COMSOL Multiphysics and Modelica/Dymola.

One of the major workarounds is the connection of COMSOL Multiphysics to Matlab as shown in Figure 5 (a). Thanks to COMSOL capability in exporting the model as a Matlab code, this code can be executed in a Simulink model by taking advantage of the so-called system-functions "S-functions". This path was partially tested by van Schijndel [9] as he examined the co-simulation between COMSOL and Simulink. The research did not further consider the co-simulation with Dymola through FMU. It was concluded that a main advantage is that COMSOL solvers can be exported with the code and used in Simulink. Moreover, given the capability of Simulink to model systems, it is argued whether further coupling to Dymola is needed.

The other two options (b) and (c) enable the connection of COMSOL indirectly to Dymola through a 3rd party interfacing tool called *"TISC*

Suite" [10]. Down to the fact that COMSOL is a closed source tool, it was difficult to have a direct connection from COMSOL to TISC Suite back and forth. Instead, it is required that the COMSOL model is converted into java code by taking advantage of COMSOL API or into Matlab code using Matlab LiveLink and then install the connection to TISC Suite within the code and, next, execute the simulation.

This work will investigate both options (b) and (c) and will not consider option (a) further since it is reported in literature [9].

Experimental Co-Simulation Setup

To test the applicability of the options (b) and (c) in Figure 5, a very simplified model was used. The model was firstly realized in COMSOL Multiphysics and, then, the required physics were added. Next, the required input values for the boundary conditions (BCs) were set as received values from Modelica/Dymola.

Herein, the model is a 2-D rectangular stainless-steel sheet with 2 m (height) by 1 m (width) and its thermophysical properties are (k = 56 W/m/K, c_p = 2150 J/kg/K and ρ = 880 kg/m^3). The physics used in the model were narrowed down to only "Heat Transfer in Solids" interface in order to avoid highly-cost simulations. Figure 6 shows the allocation of the boundary conditions over the simulated geometry.

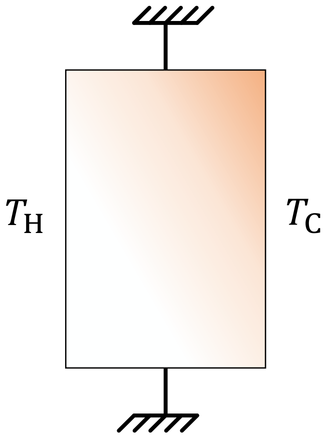


Figure 6. Representation of the investigated case with the assigned boundary conditions.

Accordingly, the simulated geometry is thermally insulated at the top and the bottom, whereas it has temperature BCs on both right and left sides. For the mesh, a finer physics-controlled mesh was used with an elements number of 312.

Both temperature boundary conditions were set up to be sent from Modelica/Dymola, where T_H is a sinus function and T_C has a constant value as follows:

$$T_\mathrm{H} = 500.15 + 150 * sin(2\pi x/(10)) \qquad (1)$$

$$T_\mathrm{C} = 293.15 \qquad (2)$$

Where the average hot temperature is 500.5 K and it has periodic function with an amplitude of 150 K over a time of 10 seconds.

Simulation Results

COMSOL Multiphysics Benchmark

In order to initialize a reference system, the model was firstly simulated using COMSOL tool, where all inputs are internally given by the tool itself.

Co-simulation Options

To investigate the option (b) shown in Figure 5, the COMSOL model was exported as Matlab code (*.m) using the interfacing feature Matlab LiveLink for COMSOL. The advantage of this path is that the solver can be also exported with the model.

Whereas for the option (c), the model was exported into java code (*.java). Both codes' options were modified firstly whereby the connection to TISC suite was introduced and initialized.

For the simulation, a time-dependent study was used. The investigation time was set to 10 seconds with a time step of 0.1 second and a synchronization rate of 1 second for the co-simulation.

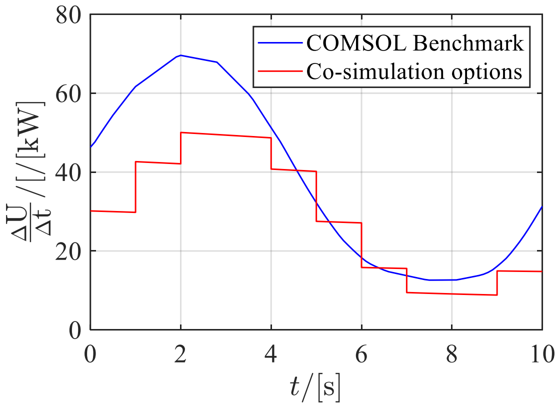


Figure 7. Sheet's internal energy change over time.

Figure 7 quantitively shows the sheet capacity over the investigated time, where the orange line represents the case if the BCs are directly implemented in COMSOL. Whilst the light blue line represents the case of co-simulation between Dymola and COMSOL for a synchronization rate of 1 second.

Discussion

Figure 7 conveys a straightforward message, which indicates that there exists a loss of information if a co-simulation is chosen. The co-simulation results were of a hybrid nature, which means they are a combination of both continuous and discrete patterns. The discrete nature is attributed to the so-called *"ping-pong"* co-simulation. This approach is described as follows:

- Tool 1 (leading tool) sends a signal to the tool 2, which runs with initial values from the start time till the first synchronization time step;
- Then, tool 2 sends a signal to tool 1 that simulates the model until next synchronization time step is reached; then
- Both leading tool (tool 1) again sends information and, therefore, tool2 utilizes the received data for the next time step and so on for each synchronization time step.

Figure 8 schematically illustrates the process sequence. Applying this to our specific example, Dymola starts the simulation as a leading tool and sends a signal (initial values of T_H and T_C) to COMSOL, which runs from 0 to 1 second and, next, Dymola receives a signal from COMSOL and runs the simulation until the time reaches 1 second. Then, Dymola sends signals to COMSOL, which carries out the simulation until the next synchronization rate. The co-simulation approach will proceed like this at each single synchronization time step. Therefore, the discrete nature will be observed. However, it is argued that using shorter synchronization rates will give better results and reduces the amount of data lost. Nevertheless, it is important to monitor the computation effort (i.e. performance).

This example run on a work station with 64-bit Windows operating system with Intel Core i7-7820HQ 2.9 GHz quad core processor and 16 GB RAM; thus, the performance is given in Table 1.

Table 1. Computational performance comparison

Case	Simulation time [sec]
COMSOL Benchmark	2
Option (b)	38
Option (c)	54

It is clearly seen that the Matlab (option b) outperforms java compiler resulting into less execution time. However, the simulation results remain the same in both cases.

Needles to remind that this time involves only the simulation time. Thus, the hand work was not included in the computational efforts. Given the experience level, the hand work can significantly vary. However, an average of 15 minutes was estimated for installing the connection between the tools and the initialization step for the given example. It is important to point out that this time can increase significantly once the user shifts to the advanced applications (co-simulation of TES in an R-DH), especially if a control unit is installed. Therefore, the main highlight is that both options seem to work and deliver results, but still both suffer from a major drawback, which is the long execution time and, therefore, both are infeasible in terms of computation efforts.

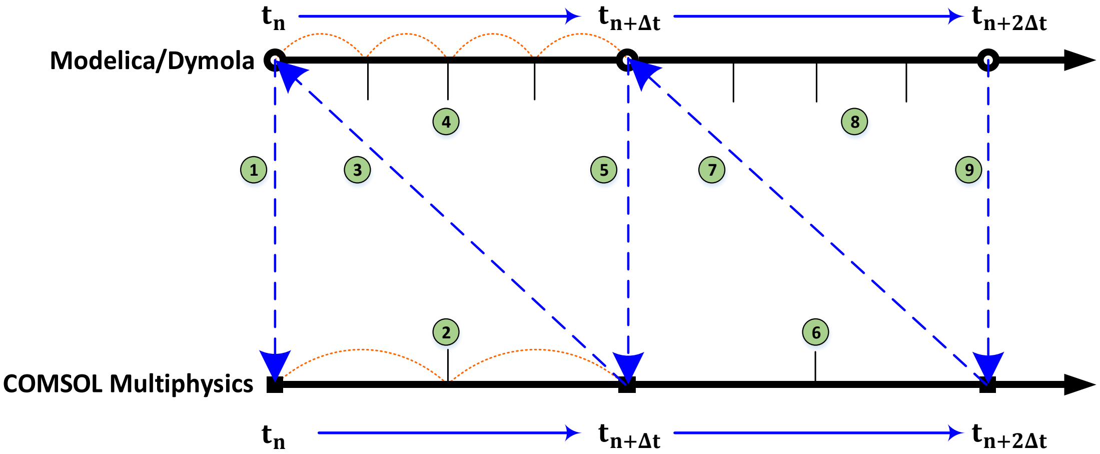


Figure 8. Schematic representation of ping-pong co-simulation approach.

Having seen the results and drawbacks, it is worthy to mention that there exists a potential to implement the co-simulation within COMSOL environment and automate this application. Consequently, this will make the entire process easier to control and it also improves the quality of results leading to no loss in data.

Conclusions

Co-simulation arises as a promising technique for modelling and simulation of complex system with multiphysics aspects. Thus, it has received great attention in the last few years. In this context, COMSOL Multiphysics appears as one of the robust tools for detailed multi-physical modelling, whereas Modelica/Dymola is being increasingly used in dynamic system simulations. Therefore, this work investigated the coupling of both tools. It is found that there exist numerous interfaces for COMSOL with other tools. However, Dymola was not supported and, therefore, the work presented possible options for co-simulation. Out of these options, two pathways were chosen, examined and compared to a benchmark. The chosen options included a 3rd-party tool “TISC Suite”.

Having considered the results, it is held that the option involves Matlab LiveLink outperforms the other option with java compiler. Compared to the benchmark, however, the computational performance was tremendously low. Besides, it is found that there exists a loss of data due to the adopted co-simulation approach (ping-pong).

References

[1] S. Nižetić, N. Djilali, A. Papadopoulos and J. J.P.C. Rodrigues, "Smart technologies for promotion of energy efficiency, utilization of sustainable resources and waste management," *Journal of Cleaner Production,* vol. 231, pp. 565-591, 2019.

[2] A. Dahash, F. Ochs, M. Bianchi Janetti and W. Streicher, "Advances in seasonal thermal energy storage for solar district heating applications: A critical review on large-scale hot-water tank and pit thermal energy storage systems," *Applied Energy,* vol. 239, pp. 296-315, 2019.

[3] C. Waibel, R. Evins and J. Carmeliet, "Co-simulation and optimization of building geometry and multi-energy systems: Interdependencies in energy supply, energy demand and solar potentials," *Applied Energy,* vol. 242, pp. 1661-1682, 2019.

[4] M. Arnaudo, M. Topel, P. Puerto, E. Widl and B. Laumert, "Heat demand peak shaving in urban integrated energy systems by demand side management - A techno-economic and environmental approach," *Energy,* 2019.

[5] van Schijndel, A.W.M. (Jos), "A review of the application of SimuLink S-functions to multi domain modelling and building simulation," *Journal of Building Performance Simulation,* vol. 7, no. 3, pp. 165-178, 2014.

[6] Trcka, M., Wetter, M. and Hensen, J.L.M. , „Comparison of co-simulation approaches for building and HVAC/R system simulation,“ in *Proceedings of the 10th IBPSA Building Simulation Conference*, 3-5 September 2007.

[7] Gomes, C., Thule, C., Broman, D., Larsen, P.G. and Vangheluwe, H., "Co-Simulation: A Survey," *ACM Computing Surveys,* vol. 51, no. 3, 2018.

[8] G. Schweiger, C. Gomes, G. Engel, I. Hafner, J. Schoeggl, A. Posch and T. Nouidui, "An empirical survey on co-simulation: Promising standards, challenges and research needs," *Simulation Modelling Practice and Theory,* vol. 95, pp. 148-163, 2019.

[9] van Schijndel, A.W.M. (Jos), "Implementation of COMSOL Multiphysics® in Simulink® S-Functions, Revisited," in *Proceedings of the Comsol Conference Cambridge 2014,* Cambridge, UK, 17-19 september 2014.

[10] "TISC Suite – Software package for coupling simulation tools," [Online]. Available: https://www.tlk-thermo.com/index.php/en/software-products/tisc-suite. [Accessed 2 August 2019].

Acknowledgements

This project is financed by the Austrian “Klima- und Energiefonds” and performed in the frame of the program “Energieforschung”. It is part of the Austrian flagship research project “Giga-Scale Thermal Energy Storage for Renewable Districts” (giga_TES, Project Nr.: 860949). Therefore, the authors wish to acknowledge the financial support for this work.

Moreover, the authors wish to thank Dipl.-Ing. Leo Gall and Dipl.-Ing. Thomas Alpoegger (LTX GmbH) and Dr.-Ing. Björn Flieger (TLK-Thermo GmbH) for their valuable support and the continuous follow-up.