13th INTERNATIONAL MODELICA CONFERENCE
March 4–6, 2019
Ostbayerische Technische Hochschule Regensburg, Germany
DACCOSIM 2019

Distributed Architecture for Controlled CO-SIMulation New Generation for "*co-simulation made simpler*"

Jean-Philippe Tavella [1] Pr Stéphane Vialle [2,3]
Dr José Évora-Gómez [4] Dr José-Juan Hernández [5] Dr Enrique Kremers [6]

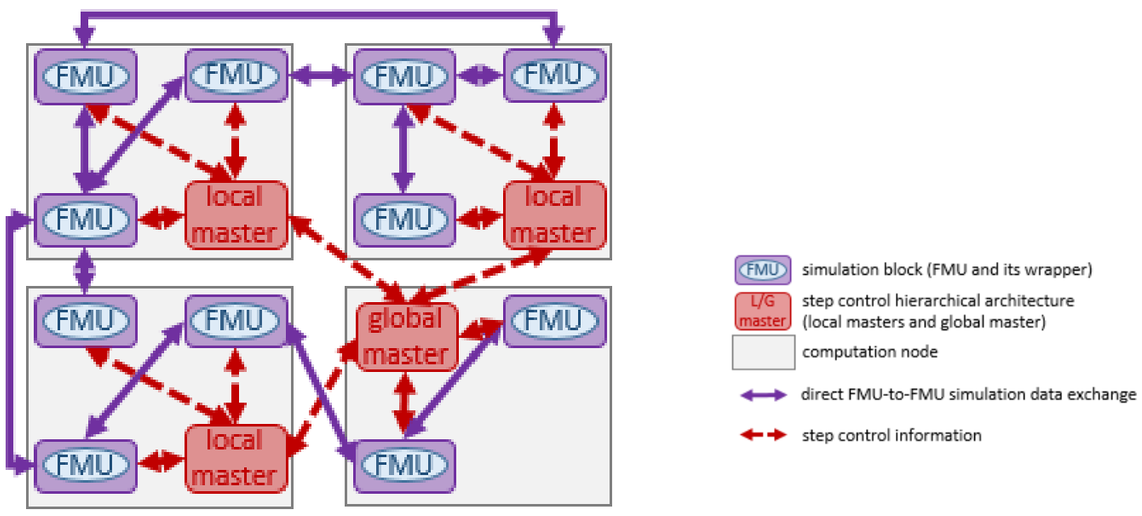
[1] EDF Lab Paris-Saclay, 91120 Palaiseau, France
[2] CentraleSupelec, Université Paris-Saclay, 57070 Metz, France [3] LRI - UMR 8623, 91190 Gif-sur-Yvette, France
[4] MONENTIA, Parque Científico Tecnológico, Campus de Tafira s/n, 35017 Las Palmas de GC, España
[5] SIANI, Instituto Universitario de Sistemas Inteligentes y Aplicaciones Numéricas en Ingeniería, Universidad de Las Palmas de GC, España
[6] EIFER, European Institute for Energy Research, 76131 Karlsruhe, Deutchland

DACCOSIM NG: "*co-simulation made simpler*"

DACCOSIM NG (**D**istributed **A**rchitecture for **C**ontrolled **CO**-**SIM**ulation, **N**ew **G**eneration) aims at enabling *co-simulations made simpler* based on the Co-Simulation part of the **FMI** 2.0 standard (July 2014). This solution deals with the cooperation of multiple active components called FMUs (Functional Mockup Units). It is currently assessed on industrial use cases, mainly for Energetic Systems simulations, albeit it is designed to be generic.
The DACCOSIM NG 2019 suite provides a graphical framework to easily design a calculation graph and automatically configure the associated master algorithm. This code also relies on the JavaFMI toolbox to wrap FMUs (https://bitbucket.org/siani/javafmi).
DACCOSIM NG 2019 engine is multithreaded in order to take advantage of multi-core computing cluster nodes. This allows to support large scale co-simulations (size up).

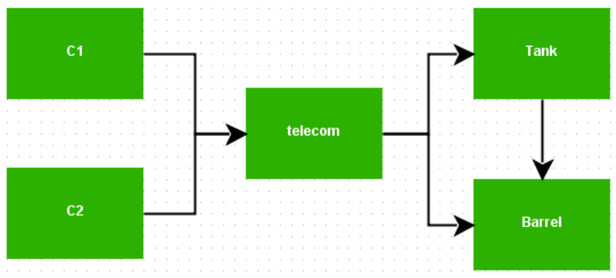
A Hierarchical Master for Parallel Simulation

The FMI-CS standard allows to locally uncouple and parallelize the FMUs as independent time simulators synchronized only at scheduled time instants, but the concept of Master Algorithm is not specified.
The DACCOSIM Master Algorithm implements a 3-layer decentralized architecture with a hierarchical and distributed step size negotiation and direct FMU-to-FMU data exchanges.



DACCOSIM NG 2019 User Interface

DACCOSIM NG 2019 includes a **Graphical User Interface** (Editor) to define calculation graphs (FMU parameters, FMU-to-FMU data exchanges), and to run co-simulations.



For complex calculation graphs (i.e. including thousands of FMUs and peer to peer connections), a Domain Script Language is available to replace the GUI, from design to execution. Conversely, a valid graphical calculation graph done with the Editor can be exported as a reusable script file.
Calculation graph scripts are runnable in batch mode (Shell) or with the Editor.

Contacts :
Jean-Philippe Tavella jean-philippe.tavella@edf.fr
Stéphane Vialle stephane.vialle@centralesupelec.fr
José Evora jose.evora@monentia.com

Main Implemented Features

DACCOSIM NG 2019 relies on the FMI-CS 2.0 features in order to offer:

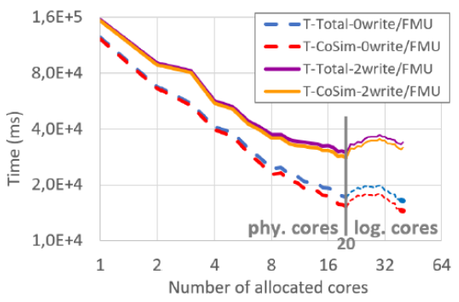
- **Global causal dependency graph** automatically generated using both the FMU internal dependencies and the calculation graph in order to determine the causal chains and loops.
- **Global and distributed co-initialization method** based on a generalized Newton-Raphson algorithm able to solve hybrid dependency graphs.
- **Distributed master kernels** operating with constant or adaptive time steps. Error control is performed using single-step method (Euler) or multiple-steps method (Adams-Bashforth).
- **Input data extrapolation** for more accurate co-simulations.
- **Experimental 'early return' and 'flow variables' features** ahead of the future FMI 3.0 version.
- **Automatic generation of a "Matryoshka"**, which is a multithreaded super-FMU encapsulating a DACCOSIM NG calculation graph and its inner FMUs.

DACCOSIM NG on cluster resources

DACCOSIM NG is not yet able to distribute massive co-simulations on cluster resources but this feature is expected for early 2020 thanks to the Java Message Service.
A Java extension named DacRun (DACCOSIM Runner) is under development and will be in charge of the deployment, the run and the collect of DACCOSIM application results on Linux computing resources compliant with OAR and SLURM cluster managers. DacRun will check every stage of the distributed execution and will be able to generate detailed reports to facilitate debugging on clusters.
Mainly usable for conducting copies of the same core code (Shell) for efficient distribution on multi-machine environment, DacRun will also be runnable on multi-core shared memory machines.

DACCOSIM NG 2019 Use Cases

DACCOSIM NG 2019 is now applied on industrial use cases provided by EDF (French electrical utility company) either for its own needs or for several French National projects, especially in the domain of Energetic Systems (e.g. ModeliScale, PSE).
All cases will be soon run on HPC clusters with Infiniband communication networks in order to exhibit more significant speedups on wide systems.



DACCOSIM NG 2019 Distribution

DACCOSIM NG 2019 is being developed in Java for both Windows and Linux 64-bit platforms. The code is distributed under the LGPL V3 open source license, downloadable at https://bitbucket.org/simulage/daccosim and accompanied by recent publications, a User's Guide documentation, and many examples.
DACCOSIM NG and associated tools are permanently improved, tested and maintained by Monentia (http://www.monentia.es), and the RISEGrid institute (http://www.centralesupelec.fr/risegrid-institute-research-institute-smarter-electric-grids).