Available online at www.sciencedirect.com

ScienceDirect

IFAC-PapersOnLine 48-1 (2015) 920–921

A *Modelica*-based Modeling and Simulation Framework for Large-scale Cyber-physical Systems of Systems

S. Nazari*, C. Sonntag**, S. Engell*

*Process Dynamics and Operations Group, Department of Biochemical and Chemical Engineering, Technische Universität Dortmund, Dortmund, Germany (email: shaghayegh.nazari@bci.tu-dortmund.de, sebastian.engell@bci.tu-dortmund.de)
**euTeXoo GmbH, Dortmund, Germany (e-mail: christian@eutexoo.de)

Abstract: This paper proposes a modeling, simulation, and validation framework for large-scale technical *Cyber-physical Systems of Systems* with distributed management architectures. The goal of the framework, which is based on the freely available *Modelica* language for object-oriented heterogeneous modeling, is to reduce the (currently very large) engineering effort for distributed management architectures while improving the quality of the designed system. The framework allows engineers to equip the management system with standardized interfaces, which will significantly increase re-usability of newly developed and legacy models. Furthermore, the overall CPSoS model, including the communication architecture, is generated automatically, making tedious manual implementation superfluous while reducing the potential of modeling errors, and the standardized, generic interfaces to which model components must connect to will provide a straightforward avenue for the deployment of management solutions to industrial hardware systems.

Keywords: Modeling, Simulation, Distributed Management, Modelica, Cyber-physical Systems of Systems, Complex Systems, Distributed Management, Distributed Control

1. INTRODUCTION

Modern large-scale, complex technical systems, such as electric power networks, railway systems, or large industrial production sites, consist of many, partly autonomous, physical and cyber-components with complex interactions. The automation/management architecture of such *Cyber-physical Systems of Systems* (*CPSoS*)[1] reflects this complexity in that it is highly distributed and consists of several interacting hierarchical layers. In particular, many of the components are controlled locally (e.g. production plants in a chemical production site) while global coordination is needed to drive the complete CPSoS towards the fulfilment of global performance and safety criteria (such as coordination of the steam and energy generation in chemical production sites).

Model-based engineering is essential to guide and validate the design of these systems, to engineer and test automation architectures, to determine performance measures, to design and validate safety measures, and to train operators. Consequently, the use of model-based technologies is now well established in industrial practice. While powerful modeling and simulation languages and tools are available, the lack of frameworks that are tailored to CPSoS with distributed management architectures poses severe limitations. The complex automation and communication architectures of management systems, including their connections to models of the technical system, must be implemented manually. This:

1. Leads to a large engineering effort (and, consequently, financial) overhead,
2. May introduce additional errors into the model,
3. Makes it very difficult to re-use existing model components, and
4. Complicates the testing of an existing management scheme on different technical systems (or even different models of the same technical system) since manual implementations usually focus on a specific system model due to the lack of pre-defined standard interfaces for the automation system.

In this contribution, a new modeling and simulation framework for CPSoS with distributed management is presented that is currently developed within the European research and innovation project DYMASOS[2] and that provides automated support for the model-based design and validation of CPSoS under distributed management.

2. THE MODELING AND SIMULATION FRAMEWORK

The modeling and simulation framework will provide a structured approach to the implementation of the large-scale CPSoS models. It is based on the freely available, object-oriented *Modelica* language which was specifically designed for heterogeneous systems modeling, see e.g. [1, 2], which is a very rich language for equation-based modeling that has achieved wide adoption in a variety of industrial branches.

The framework represents each subsystem using one of four model components, as shown in Fig. 1. The management architecture consists of communicating *local coordination algorithms* and an (optional) *global coordination algorithm*.

[1] See http://www.cpsos.eu

[2] DYMASOS: Dynamic Management of Physically Coupled Systems of Systems. Supported by the European Commission under the FP7-ICT Programme (contract no. 611281), http://www.dymasos.eu

10.1016/j.ifacol.2015.05.190

The local coordination algorithms optionally have access to a *design model* to determine a locally optimal subsystem operation with respect to a *local problem formulation*. The local coordination algorithms perform real-time control of the physical CPSoS that is represented by a set of *validation models*, which are detailed models that accurately represent the real physical subsystems and their interconnections.

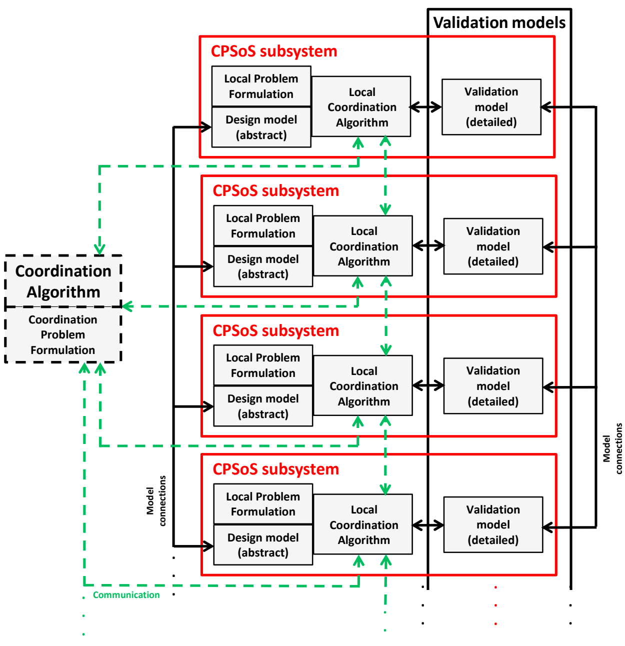

Fig. 1: Structure of a CPSoS model in the modelling and simulation framework.

The framework provides standardized interfaces for the connection of physical process model components (i.e. design and validation models) as well as automation components (i.e. local and global coordinators). In order to implement the interconnections of the model structure shown in Fig. 1, four interfaces must be specified, namely the interfaces between global and local coordination algorithms, between local coordination algorithms, between local coordination algorithms and validation models, and between validation model components. Here, the interface between two validation models is implemented using the standard *Modelica* interfaces while dedicated, generic interface definitions are developed for the connections within the distributed management system, and for the connection of the management system to the validation models. All interfaces support time-discrete as well as event-driven communication. In addition to these two types of communication, the interface between the validation models and the local coordinators also supports continuous communication. This option is considered for cases where the local coordinator contains a simple low-level controller, e.g. a PI controller.

The framework will enable a fully automatic generation of interconnections and of the communication structure of the CPSoS in *Modelica*. The information on the communication structure and the parameterization of the interfaces (i.e. numbers, names, types and dimensionalities of the variables that can be transmitted via the interfaces) are provided to the model generator via an XML configuration file. Based on this information and repositories of white-box and black-box model and automation components (the latter of which can be connected by co-simulation via the *Functional Mockup Interface/FMI*), the automatic model generator will first verify the structural correctness of the interconnections, generate the required communication structure of the CPSoS model, and integrate all model components into a consistent model of the complete CPSoS. The workflow of the generation process of the automated CPSoS model is shown in Fig 2. The interaction of all the model components takes place via a *Modelica-based model management engine* that allows management components to communicate (e.g. in the beginning of the simulation coordinators can define their parameters such as execution delay, sampling time, etc., via the interfaces). During the simulation, they can use the interfaces to send data requests to other coordinators or retrieve information about the model structure.

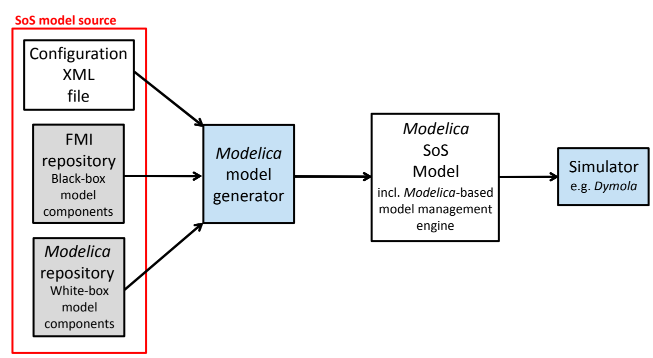

Fig. 2: Workflow for the generation and execution of automated CPSoS models.

This plug-and-play approach makes model development much more efficient and less error-prone, makes model components easily re-usable in different scenarios, allows to easily test and validate different management algorithms on an (existing) model of an industrial CPSoS, and simplifies the direct deployment of new management architectures to the automation hardware of the real-world CPSoS due to the availability of generic interfaces that can be directly connected to hardware systems.

The framework will be illustrated on an industrial application example, the CPSoS model of petrochemical production complex where different plants are interconnected by networks of mass and energy.

REFERENCES

[1] P. Fritzson and P. Bunus: Modelica – A General Object-Oriented Language for Continuous and Discrete-Event System Modeling and Simulation. In *Proc. Annual Simulation Symposium*, 2002, 365-380.

[2] Modelica and the Modelica Association, http://www.modelica.org.