A Model-based Approach for Designing Cyber-Physical Production Systems

Stefano Spellini
Univ. of Verona, Italy
stefano.spellini@univr.it

***Abstract*—"Industry 4.0" proposes to transition from "blind" mechatronics systems to Cyber-Physical Production Systems. The target of this thesis is to support such a transition, employing a model-based approach. In particular, this framework exploits the Systems Modeling Language (SysML), allowing the reuse of architectural and process models, and designing novel functionalities on top. Furthermore, this thesis proposes a formalization approach to SysML diagrams, based on the Assume-Guarantee reasoning through contract-based specifications. By exploiting reactive synthesis techniques, contracts can be used to prototype and verify the system's components.**

The proposed framework has been built and tested alongside the construction of the ICE Laboratory, a research facility consisting of a full-fledged production line.

I. Introduction

Market trends of the 21st Century are characterized by high product demands along with high degrees of customization. Manufacturing technologies must evolve to cope with the increasing unpredictability of modern society conditions while guaranteeing cost-efficiency. "Industry 4.0" [1] is meant to assist this transformation, proposing a set of *production systems development guidelines* to a wide range of engineering disciplines, from systems design to product development. Among the promises of the Industry 4.0 trend, the concept of *reconfigurability* in manufacturing systems stands out as a key factor in quickly reacting and adapting to market changes [2].

The ongoing transformation is particularly problematic for Small and Medium Enterprises (SMEs). While a large manufacturing corporation may consider redesigning their production plants from scratch to incorporate novel technologies, SMEs are often forced to gradually introduce intelligence in their already existing lines. Furthermore, companies must be able to evaluate in advance the impact of re-designing their production lines. As such, a methodology capable of representing different systems' viewpoints and, thus, enabling a holistic approach to the design, is still missing.

Furthermore, different standards and languages have been developed in the past to organize the knowledge within the manufacturing context. Nonetheless, multiple languages are extremely specialized to represent a single aspect, but an all-encompassing approach does not exist. When (re-)designing, configuring, or optimizing a production system, the complexity induced by the presence of such heterogeneous information is hardly bearable if tackled manually. Thus, design automation and model-based techniques become crucial.

For such reasons, this thesis proposes a design framework for Cyber-Physical Production Systems (CPPSs) based on models. This framework exploits the expressivity of the System Modeling Language (SysML) to guarantee a complete and familiar modeling environment. To enable reuse, an approach is also proposed to import models expressed using other languages (*i.e.*, Automation Markup Language (AML)). Furthermore, the formalization of the system's specifications into Assume-Guarantee (A/G) contracts allows designers to verify properties and produce components implementations.

II. Methodology

The overall conceptual framework proposed by this thesis, named MOOD4I, is depicted in Figure 1. The methodology flow starts from a set of standards and languages to describe various aspects of the production line. In particular, the framework takes as input the production recipe expressed through a graph representation named Resource Task Network (RTN). Furthermore, the structural view of the plant is defined by handling AML specifications and mapping such knowledge to SysML diagrams. The framework also allows importing production requirements structured using the International Society of Automation (ISA)-95 standard. Therefore, one of the objectives of the proposed framework is to allow designers reusing already available descriptions of parts of the system.

The core of MOOD4I is composed of the SysML language and Assume-Guarantee (A/G) reasoning through contracts. On the one hand, SysML provides an intuitive and complete modeling language for a broad range of systems. It enables the specification of multiple system's viewpoints, from the architecture to behaviors. Furthermore, according to the Service Oriented Manufacturing (SOM) principle, the framework proposes a hierarchical model of production recipes, designed to expand the knowledge regarding the production process compared to Resource Task Network (RTN). Therefore, the proposed design flow can map the system's features described by the input representations to SysML diagrams. On the other hand, MOOD4I exploits A/G contracts to decompose the system design problem, to verify properties (*i.e.*, *feasibility*) and to synthesize implementations using reactive synthesis algorithms. Such a set of techniques provides a connection between formal reasoning and model-based design, especially applied to production systems.

The outputs of the proposed framework are a set of artifacts and implementations exploiting the knowledge provided by the models. First, MOOD4I is capable of constructing virtual prototypes of the modeled system. This task requires a formalization step of models properties into A/G contracts, to synthesize a set of components' implementations. The prototype can be simulated to estimate key parameters (*i.e.*, timing and power consumption) and to verify its correctness. Secondly, in a similar manner, the set of synthesized implementations

Figure 1: Overview of the conceptual flow proposed by the Modeling, Formalization & Design for Industry (MOOD4I) framework.

can be conceptualized and constructed to be integrated into the actual system. Regarding production processes, the set of recipe models is exploited by the proposed scheduling algorithm, reacting to unforeseen events and optimizing the makespan according to the recipe decomposition.

A. Cyber-Physical Production Systems Modeling

The first point of the proposed framework is a modeling strategy for CPPS. To acquire and structure knowledge about the system, it describes modeling and design flows based on the Platform-Based Design (PBD) paradigm [3]. The methodology supports both the *top-down modeling of requirements and functionalities*, as well as the *bottom-up reuse of components* already existing in the system and available to designers.

To represent complex recipes, MOOD4I presents a multi-level, hierarchical modeling strategy for production processes. To carry out such a task, the modeling strategy assumes that the system is configured according to the principles of SOM [4]. The recipe model is structured over three levels, each of them representing a different abstraction of the production: (1) the *task level* consists of a task-resources graph. It allows describing the bones of the production process, which are the tasks, their dependencies and the machines on which such tasks can be allocated; (2) the *service level* refines the concept of "task", describing the sequence of steps required to be carried out to complete the task; (3) the *machine function level* describes the interactions that need to take place between the control software and the machine implementing a service.

B. Compositional Design and Virtual Prototyping using Assume-Guarantee Contracts

The MOOD4I framework enables compositional reasoning about the system's design. It allows decomposing the design problem and, thus, splitting responsibilities of achieving particular functionalities over the set of components composing the system. Such an approach is guaranteed by employing A/G contracts and contracts theory [5]. In fact, A/G contracts allow to decompose system design among the different components involved (*i.e.*, horizontal decomposition), and among different levels of abstraction (*i.e.*, vertical decomposition). MOOD4I present a *compositional approach to generate the control software* for CPPSs. The main innovation of our methodology is the structured decomposition, supported by the A/G contract formalism, of the design process. Furthermore, the set of implementations is constructed in a way that facilitates their integration into simulators and software architectures (*i.e.*, Robot Operating System (ROS)).

C. Service-oriented and Model-based Scheduling

The scheduling algorithm proposed by MOOD4I is a reactive-dynamic algorithm exploiting the proposed three-level modeling approach. Therefore, the scheduler assumes to be implemented into a manufacturing system integrating a Service Oriented Architecture (SOA). By leveraging on the concept of "service", the scheduling algorithm exploits the information represented in the model, aiming at minimizing the makespan of the production while maximizing the machine utilization. The algorithm exploits the increased granularity of the model to schedule the sub-tasks (*i.e.*, services) specified within the tasks. Furthermore, knowing the encapsulated services within a sub-task, it is possible to identify the required resources (*i.e.*, tools, materials *etc.*). This allows to schedule these sub-tasks taking into account the various delays, such as waiting times for the retrieval of missing resources.

III. Directions for Future Research

To automatically construct contract-based specifications from SysML models, it is necessary to integrate in MOOD4I a formalization framework that is built around A/G contracts. As an example, the Contract-based Heterogeneous Analysis and System Exploration (CHASE) framework [6] provides a library of constructs that, on the one hand, can be related to SysML diagrams elements and, on the other hand, have a direct mapping to logic formulas. We plan to exploit and integrate such a framework in MOOD4I in the future.

References

[1] H. Lasi, P. Fettke, H.-G. Kemper, T. Feld, and M. Hoffmann, "Industry 4.0," *Business & information systems engineering*, vol. 6, no. 4, pp. 239–242, 2014.

[2] Y. Koren, U. Heisel, F. Jovane, T. Moriwaki, G. Pritschow, G. Ulsoy, and H. Van Brussel, "Reconfigurable manufacturing systems," *CIRP Annals*, vol. 48, no. 2, pp. 527–540, 1999.

[3] K. Keutzer, A. Newton, J. Rabaey, and A. Sangiovanni-Vincentelli, "System-level design: orthogonalization of concerns and platform-based design," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 19, no. 12, pp. 1523–1543, 2000.

[4] T. Lojka, M. Bundzel, and I. Zolotová, "Service-oriented architecture and cloud manufacturing," *Acta polytechnica hungarica*, vol. 13, no. 6, pp. 25–44, 2016.

[5] A. Benveniste, B. Caillaud, D. Nickovic, R. Passerone, J.-B. Raclet, P. Reinkemeier, A. Sangiovanni-Vincentelli, W. Damm, T. A. Henzinger, and K. G. Larsen, "Contracts for system design," *Foundations and Trends® in Electronic Design Automation*, vol. 12, no. 2-3, pp. 124–400, 2018.

[6] P. Nuzzo, M. Lora, Y. A. Feldman, and A. Sangiovanni-Vincentelli, "CHASE: Contract-based requirement engineering for cyber-physical system design," in *Proceedings of IEEE/ACM Design Automation and Test in Europe (DATE) 2018*, 2018, pp. 839–844.