HAL
open science

A holonic-based method for design process of cyber-physical reconfigurable systems

Abdelmonaam Abid, Moncef Hammadi, Jean-Yves Choley, Alain Riviere, Maher Barkallah, Jamel Louati, Mohamed Haddar

To cite this version:

Abdelmonaam Abid, Moncef Hammadi, Jean-Yves Choley, Alain Riviere, Maher Barkallah, et al.. A holonic-based method for design process of cyber-physical reconfigurable systems. Systems Engineering (ISSE), 2016 IEEE International Symposium on, Oct 2016, EDIMBURGH, United Kingdom. pp.1-5, 10.1109/SysEng.2016.7753180 . hal-01589567

HAL Id: hal-01589567

https://hal.science/hal-01589567

Submitted on 6 Jun 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A holonic-based method for design process of cyber-physical reconfigurable systems

Abdelmonaam Abid, Moncef Hammadi, Jean-Yves Choley and Alain Rivière
QUARTZ EA 7393
SUPMECA-Paris
Saint-Ouen, France
{abdelmonaam.abid & moncef.hammadi}@supmeca.fr

Maher Barkallah, Jamel Louati and Mohamed Haddar
Mechanics, Modeling and Production Research Laboratory
National School of Engineers of Sfax
University of Sfax
Sfax, Tunisia
bark_maher@yahoo.fr

***Abstract*— Recent advances in manufacturing industry has led the way for the most frequently discussed topic among practitioners and academics which is cyber physical systems (CPSs). Design process of cyber physical reconfigurable systems (CPRSs) operating in a dynamic environment is a significant challenge when a meaningful coordination of tasks between human to machine and machine to machine is required. This work presents a method based on system engineering "V" model and the holonic concept to deal with design process of CPRSs, as well as, we propose complete steps to explain how to adopt configuration from the customization to the software implementation in the operational resources. To address this method, an example of CPRSs is provided for showing the interactions of autonomous holonic agents and to give the impact of coordinated tasks in the holonic multi-agent platform in terms of code reusability and modularity. This methodology offers substantive advantages over traditional process-design methods, which are often nonintegrated and require large quantities of iteration and long development times.**

Keywords—holonic agent; cyber physical reconfigurable systems; coordination; Janus

I. INTRODUCTION

The synergy between reconfigurable manufacturing systems and cyber-physical systems (CPSs) holds promise for the resolution of the challenge posed by the complex coordination tasks between the software components and the physical environment. Both internal and external changes of manufacturing systems should be solved in real-time by adopting a computational intelligence to process the information and provide appropriate control settings in an effective manner.

These Cyber-Physical Reconfigurable Systems (CPRSs) are required to operate correctly while ensuring an appropriate design process of physical and software entities either for expected or unexpected events.

The huge significance of modeling and representation of CPSs has long been recognized by researchers and considered to be a grand challenge. Hence, the main key concerns CPSs lie in developing a complete method with high flexibility. To conduct this challenge, a limited related works examine reconfigurability in CPSs taking into account the complex interactions between computational and physical entities.

To address the emerging behavior design, [1] presented this challenge to define a methodology to achieve emerging behavior by coordinating controlling entities online capability.

Also, in [2], a fully modular tool S-TALIRO is presented in order to perform formal monitoring, testing and verification of CPSs.

Several works do not design the complexity of the CPSs in a clear and understandable way. For this reason, another challenge included was presented in [3] as a testing framework for the goal to detect system operating conditions that cause the system to exhibit the worst expected specification robustness.

The purpose of this work is to present a design process method to coordinate tasks in CPRSs. The method describes how to accomplish the coordination tasks from the product specification to the implementation of reconfigurability in the manufacturing systems.

As CPSs are articulated on the communication between physical systems, we used the web service between the application, which is designed for the customer in order to choose the appropriate product, and the manufacturing environment which is controlled by the designer. Therefore, we adopt the decentralized system concept via the holonic multi-agent platform JANUS [4, 5] for the reason to support the dynamic structure reconfiguration. We chose the Janus platform because it provides a full set of facilities for lunching, displaying, developing and monitoring holons, roles and organization.

Furthermore, this method is based both on The "V" Systems Engineering Model and the holonic concept that facilitate the designer task to monitor, control, and verify in real time the execution of the CPRSs. Even better, we deal with the challenge of CPRSs that is based on the interaction and coordination between tasks.

The rest of this paper is organized as follow. In section 2, we present the details of the proposed method. Section 3 gives an example of the proposed method with a case study of a welding cell. Finally, section 4 reports conclusions and future works.

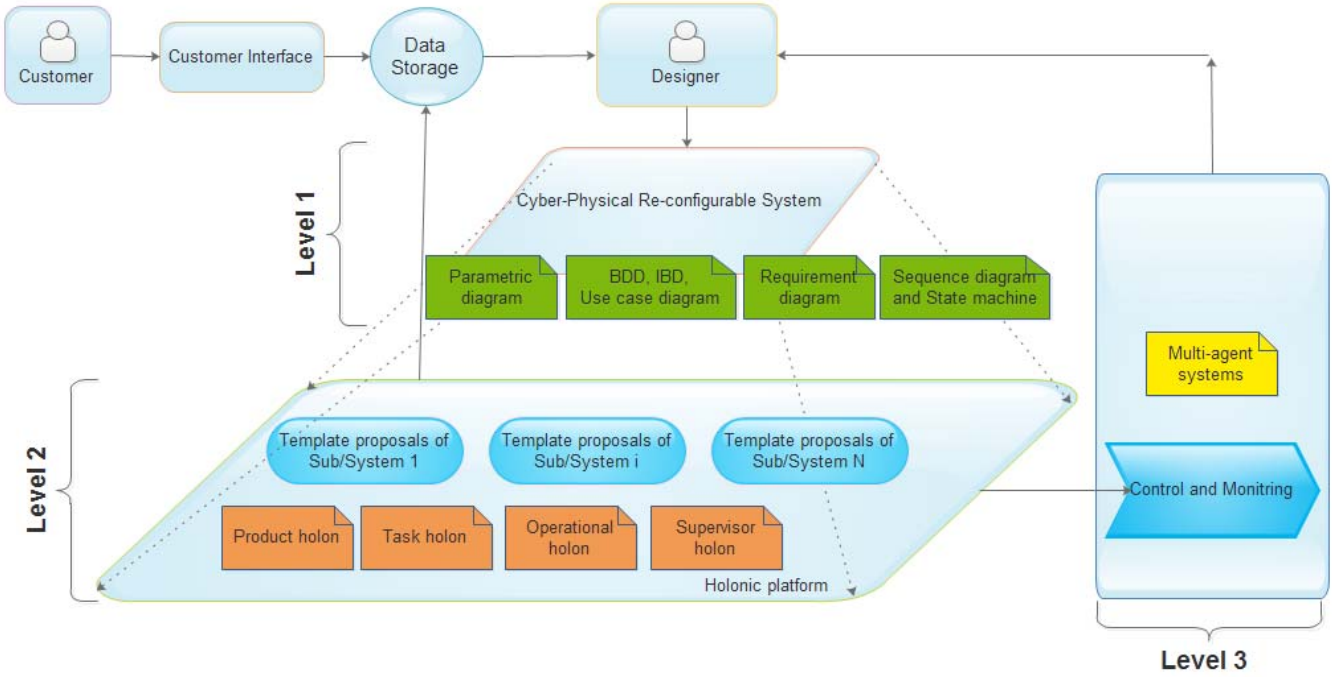

Fig. 1. The proposed method : Tasks of CPRSs process design

II. METHOD : CORDINATED TASKS IN CPRSS

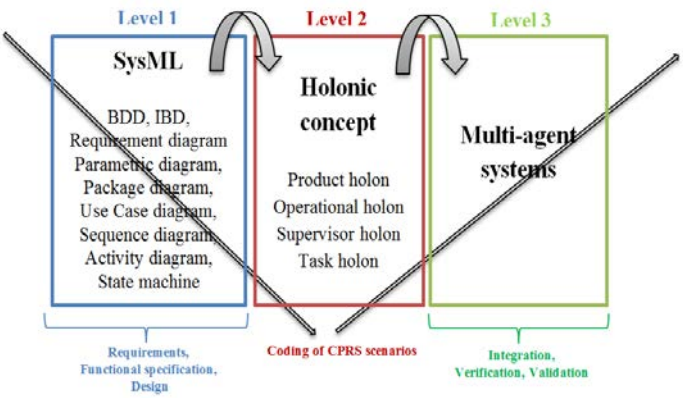

Fig. 2. The proposed model to achieve the design process of CPRS

This section introduces a design process method *(Fig. 1)* to achieve the coordination mechanism of tasks in a CPS.

A proper coordination of CPSs is a challenging problem for design, testing, verification, and validation of CPSs. Therefore, we suggest a model (*Fig. 2)* composed of three levels based on the "V" Systems Engineering Model which involves building a logical V shape sequence. The "V" model is merging as the standard way to represent systems engineering.

The following levels in the "V" model are conceived to provide a design process for CPRSs. The left side shows the requirement, the functional specification, and design of CPRS using SysML language. The central core of the "V" model shows the coding of the different parts in the CPRSs while applying the holonic concept. The right side shows the integration, the verification, and the validation steps of CPRSs as multi-agent systems.

The first level provides a good starting point for CPRSs analyses that are performed using the SysML language. It provides detailed information that can be used and expanded in design process development. In order to reduce excessive time invested in the analysis of complex and large-scale problems, the use of the system modelling language (SysML) [6] gives a starting point for the design engineer. Using SysML as a basis for designing CPRSs provides the opportunity to view all the requirements and the functional specifications in the context of surrounding systems. It also prompts the designer to think about how to design CPRS within the overall parametric, behavioral, and structure diagrams definition. Finally, it identifies the integration opportunities that should be considered and provides a head start for the CPRSs analysis.

The system design level is the first level where we focus on the solution of CPRSs. This is an important transitional step that links the system requirements that were defined in the previous level with system implementation that will be performed in the next level.

According to the second level, the developed SysML diagrams are the inputs for coding using the holonic concept. A holon is an autonomous, co-operative and intelligent entity able to collaborate with other holons to process the tasks [7]. Autonomy and cooperation are two important characteristics of holons. A holon consists of a physical component and an

informational component. The former corresponds to the holon's physical entity such as a product or a machine. The latter contains information, decision-making capabilities, and mechanisms for managing the physical component and for cooperating with other holons. Although hardware and software development may be the first task that comes to mind when thinking about designing process of CPRSs. This is a highly iterative process, particularly for software, where key features may be incrementally implemented, tested, and incorporated into the baseline over time.

In the third level, the implementation of the holonic manufacturing concepts can be done using the agent technology which is appropriate to implement modularity, decentralization, reuse and complex structures characteristics. Furthermore, CPSs integrate computing and communication with monitoring and control of physical systems. This complexity arises mostly from the complex interactions with the physical world.

The multi-agent technology provides a systematic process for integration, verification, and validation that addresses the challenges and complexity of CPRSs.

In this method, we illustrate the importance of reconfigurability and the dynamic behaviour not only in software engineering but also in the costomization level.

Meanwhile, the proposed method consists of filling the gap between customer requierement and the manufacturing environment by coordinating the corresponding tasks in a flexible way.

The conceptualization of CPRSs is shown in *Fig. 1*. Here, one or more computing entities distributed in physical environment are shown as a template proposal of systems or subsystems in the agent platform such as JANUS in order to share fonctionality and network connectivity. Moreover, it provides full sets of facilities for lunching, displaying, developing, and monitoring holons, roles and organization.

This method is based on concurrent application that performs multiple tasks at the same time and handles the interaction between these tasks. Examples of concurrent applications include rich user interfaces, servers that expose functionality via network, cloud-based applications or trading systems.

Complex product configuration design requires rapid and accurate response to customer's product needs. This is why, we propose a customer interface that links the customer interface to the manufacturing environment. The different scenarios were presented by the use case diagram (*Fig. 3*) in order to capture the interaction of the customer, as an actor, and the use cases involved in customer interface.

After the selection of the product and its caracteristics, the tabbed product will be stored in database to the designer via the web service [8].

Web service contributes in providing a standard means of interoperating between software applications running on a variety of platforms and frameworks.

Moreover, Web services are characterized by their great interoperability and extensibility, as well as their machine-processable descriptions, thanks to the use of XML.

Before the execution of the costumer command in the CPRSs, the designer received the command either as a predefined product configuration or as a new product configuration.

Technically, the agent platform can be accessible for the designer to make the appropriate changes in the templates proposals of each system in order to adapt the whole system to the given configuration.

As shown in *Fig. 4*, the designer has two choices. The first is to reuse the template proposals of the corresponding systems and sub/systems.

In this case, the configuration is proceeded automatically via the network without wasting time. The second is to revise the templates proposals of each systems and subsystems, and in this case, he just modifies the code in the agent platform. Lastly, the implementation is proceeded while the designer is controlling and verifying the coordinated tasks through the sequence diagram and/or the state machine.

This diagram shows object interactions arranged in time sequence [9] and is generated by the Janus platform by means of Sniffer agent [10].

This method enables designer to reduce time by minimizing software engineering tasks. The procedure of coordination tasks consists of (i) creating a suitable customer interface for product selection; (ii) communicating the characteristics of product with the designer; (iii) negotiating an adaptable plan to proceed the execution tasks and (iv) providing a feedback to all concerned holonic agent in order to control and verify the interaction.

The contribution of this paper is based on the design of this method in order to implement reconfigurable tasks in the agent platform according to the product characteristics. The latter allows to (i) select the best configuration mode according to the product types through code reusability and modularity; (ii) maintain a social network and (iii) facilitate the simulation and control tasks.

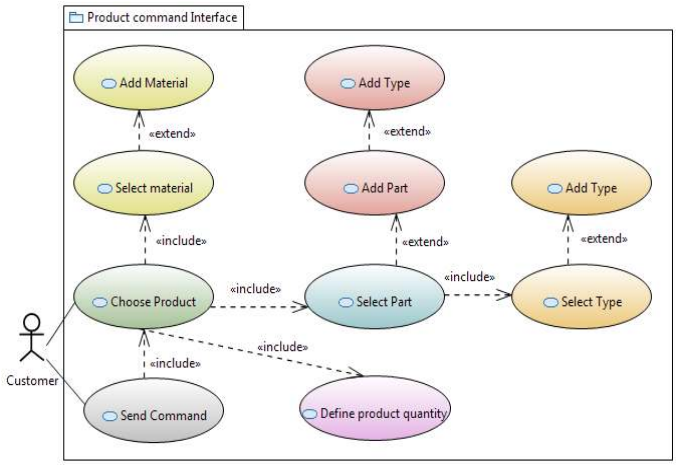

Fig. 3. Use case Diagram of the customer/product command interface

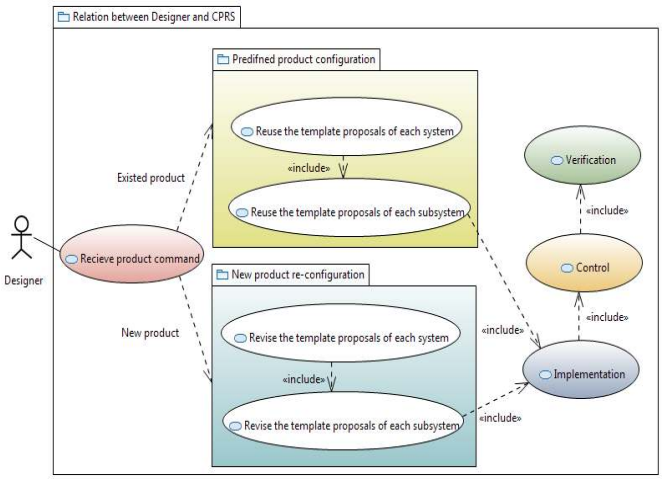

Fig.4 . Use case Diagram of the disgner with CPRSs

III. CASE STUDY OF CPRSS : WELDING CALL

This section introduces a case study based on a welding cell (*Fig. 6*). The latter is composed of one manufacturing system (Welding machine), one conveyor and one Robot (used to load/unload raw materials and products).

Initially, the system produces subparts of a scaffolding tour used for the civil engineering works. The system inputs are three raw materials: part 1, part 2, and part 3. The flow starts in the Conveyor, then passes to Robot, after that the execution of the welding operation in the Welding machine, and finally in the Conveyor.

The output of the welding machine is a subpart of a scaffolding tour welded to be passed to conveyor whereby Robot.

A new customer command is introduced (*Fig. 5*). We propose that the welding cell will be reconfigured to satisfy a new requirement, which is the production of a new kind of product adapted to the civil engineering tasks.

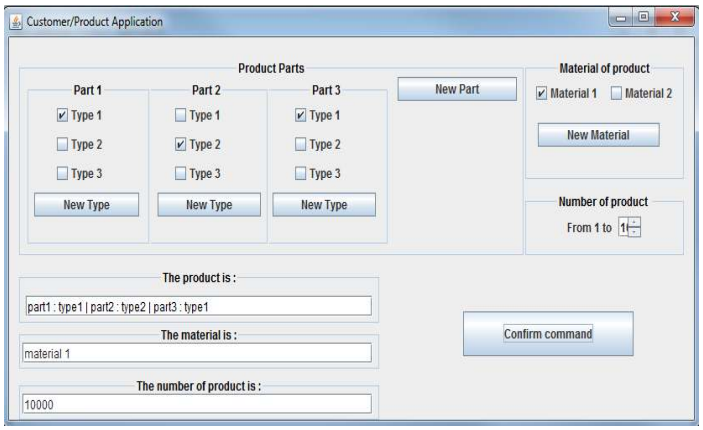

Fig. 5. Example of Customer interface

The system will produce a new kind of products according to customer requirements which is the starting point of the designing phase. Introducing a new product variety in the manufacturing process (*Fig. 8*) should be classified as an unpredicted event due to the modifications in the hardware components and the software codes.

By applying our proposed method described in section.2, the procedure begins with introducing the new requirements of the customer by setting the characteristic of the product via the customer interface. These characteristics will be stored in a database as an XML file. The first step in the manufacturing environment is done by the designer. He loads the XML file, and then he makes a comparison in order to retrieve the similar case while respecting the functional and non-functional requirements and respecting the structure of the welding cell (*Fig. 9*)

We suppose in this example that there is a new type of product that does not exist in the template proposals of JANUS platform. After that, he implements the revised java code either by adding new roles or capacities in the system templates.

Janus can generate sequence diagram of the given systems in order to allow designer to make the control and verification of the corresponding system templates.

Fig. 7 illustrates the sequence diagram of the welding cell systems and their coordination by means of exchanged messages.

As a result the designer is satisfied by the revised template proposals which respond to the customer needs.

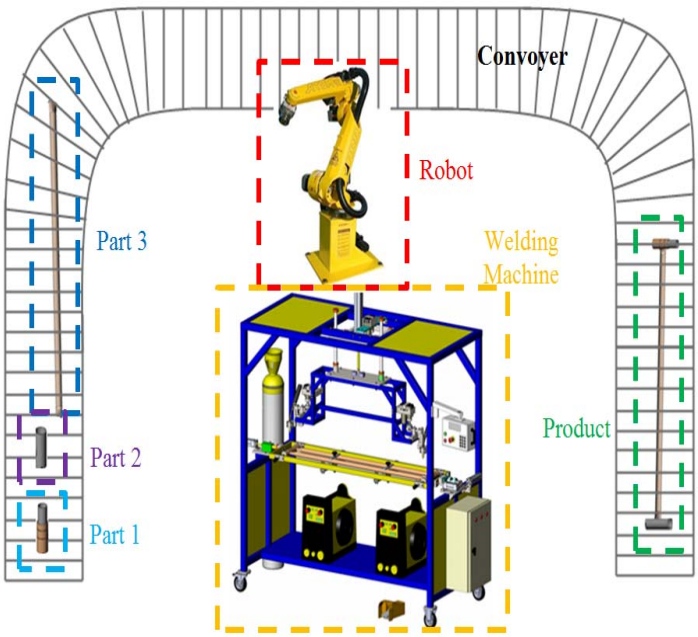

Fig. 6. CPRSs : Welding Cell

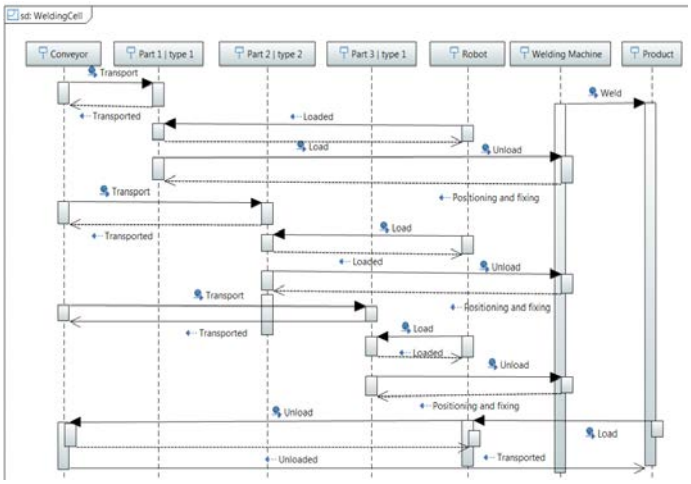

Fig. 7. Sequence Diagram of the Welding Cell (Control)

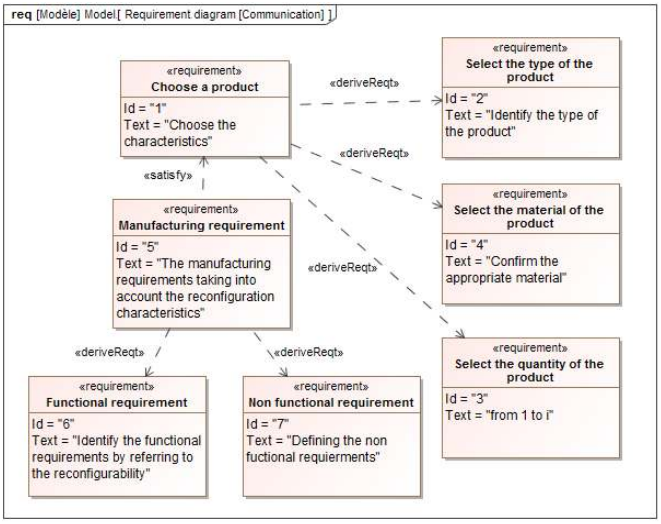

Fig. 8. Requierment Diagram of the Welding Cell (Communication)

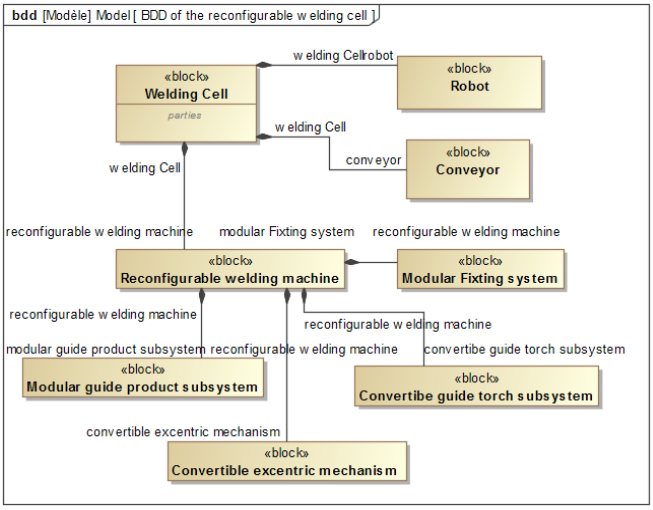

Fig. 9. BDD of the Welding Cell (Structure)

IV. Conclusion and Futur Work

This paper describes a novel methodology based on fundamental engineering models that ensure process design tasks in CPRSs

The aim of this method is to provide usable tasks for designer to ensure reconfigurability in the CPSs while allowing customer to be able to set the product characteristics in a flexible way. The outcome of the process design methodology allows formulation and process definition using the holonic concept and the multi-agent technology. The methodology, which is based on fundamental engineering "V" model, can be used as an alternative to traditional process design methods, resulting in streamlined and robust process development. This method can be used as guidance for designing CPRSs.

We presented an example of a new customer command. As a consequence, the welding cell is reconfigured to satisfy a new requirement of external disturbances which consisted of an unpredicted customer order in a welding cell. The reaction of this event that uses our proposed method responds to the changeable customer requirement in a modest period of design time.

As a future works, we will carry out a validation study based on the presented method.

References

[1] PJ. Mosterman, J. Zander, "Software & Systems Modeling, 2016Industry 4.0 as a cyber-physical system study," Softw. Syst. Model, vol. 15, 2016.

[2] B. Hoxha, H. Bach, H. Abbas, A. Dokhanchi, Y. Kobayashi and G. Fainekos, "Towards formal specification visualization for testing and monitoring of cyber-physical systems," Int. Workshop on Design and Implementation of Formal Tools and Systems.

[3] H. Abbas, B. Hoxha, G. Fainekos and K. Ueda, "Robustness-guided temporal logic testing and verification for stochastic cyber-physical systems," Proc. of IEEE International Conference on CYBER Technology in Automation, Control, and Intelligent Systems.

[4] N. Gaud, S. Galland, V. Hilaire, A. Koukam, "An organizational platform for holonic and multiagent systems," in: the Sixth International Workshop on Programming Multi-Agent Systems (ProMAS08), of the Seventh International Conference on Autonomous agents and Multiagent Systems (AAMAS), Estoril, Portugal, 2008, pp. 111-126.

[5] S. Galland, N. Gaud, S. Rodriguez, V. Hilaire, "Janus: Another yet general-purpose multiagent platform," in: the 7th Agent- Oriented Software Engineering Technical Forum (TFGAOSE-10), Agent Technical Fora, Agent Technical Fora, Paris, France, 2010.

[6] J. Estefan. (2008, Jun.). Survey of Model-Based Systems Engineering (MBSE) methodologies, INCOSE-TD-2007-003-01.

[7] A. Koestler, "The Ghost in the Machine, " Hutchinson, London, 1967.

[8] E. Uhlmann, C. Gabriel, N. Raue, "An Automation Approach Based on Workflows and Software Agents for Industrial Product-Service Systems," Procedia CIRP, 2015 – Elsevier.

[9] U. Manzoor, B. Zafar Manzoor, "Multi-agent modeling toolkit-mamt," Simul. Model. Pract. Theory, vol 49, 2014, pp. 215–227.

[10] F. L. Bellifemine , G. Caire and D. Greenwood, "Developing Multi-Agent Systems With JADE," 2007.