Structuring Cyber-Physical Systems for Distributed Control with IEC 61499 Standard

E. Monroy Cruz, L. R. García Carrillo, *Member, IEEE,* and L. A. Cruz Salazar, *Student member, IEEE*

***Abstract*—With the constant growth in size and complexity of modern industrial applications, it is not suitable to use a unique and centralized controller device for the whole process. Thus, automation solutions need to be designed by distributing control execution across different devices. Cyber-Physical Systems (CPSs) are designed to solve networked embedded systems' complexities. The IEC 61499 standard fosters developing distributed control applications by defining a platform-independent application model that can be mapped to numerous hardware components. Therefore, a relevant research topic is how to solve the design complexities of industrial CPSs by applying modeling techniques. In that context, this paper proposes an IEC 61499-based model for CPSs to distribute the complexity of control software over numerous small devices. The approach enables the creation of a comprehensive structure that can combine design, simulation, and distributed deployment of automation software. The proposed scheme is validated through a packet sorting system implemented in a commercial IDE platform.**

***Index Terms*—Cyber-Physical System, IEC 61499 Standard, Distributed Control, Industrial Automation, Function Block**

I. Introduction

Traditional industrial manufacturing typically focuses on producing large numbers of uniform and homogeneous products (mass production). These production systems are rigidly made up of interconnected elements through both hierarchical and centralized control architectures [1]. Thus, those architectures contradict the predominant trend of smart factories framed by the *Industry 4.0 (I4.0)* [2]–[4]. I4.0 enables the increasing demand for high-quality products with specific customer requirements (mass customization) to be addressed. Against these manufacturing changes, new challenges have been created to ensure the competitiveness of manufacturing companies. Nowadays, production plants must show different capabilities such as modularity, adaptability, reusability, and agility [5], [6].

E. Monroy Cruz is with Posgrado CIATEQ A.C. (Centro de Tecnología Avanzada), TecNM Campus Atitalaquia, and CETIS No. 026, Hidalgo, México, ernesto.mz@atitalaquia.tecnm.mx (Corresponding Author).

L.R. García Carrillo is with Klipsch School of Electrical and Computer Engineering, New Mexico State University, Las Cruces, NM 88011, USA, luisillo@nmsu.edu

L. A. Cruz Salazar is with Universidad Antonio Nariño, Cartagena, Colombia, and Technical University of Munich, Munich, Germany, luicruz@uan.edu.co

Such requirements frame flexible production, which has generated a growing need for developing modern automation systems with smart technologies such as *Cyber-Physical Systems* (CPSs), and *Digital Twins* for taking a central position in the new generation of intelligent manufacturing, i.e., smart manufacturing [7].

Smart factories [2], bridging the gap between automation and personalization, have many components that must be increasingly autonomous and operate in parallel. Increased parallelism improves performance while complicating system development due to a more significant number of interactions between components. This complexity necessitates the use of software engineering techniques for developing intelligent machines with integrated control devices, which in turn implies a decentralized control architecture with real-time communication between the individual components [8]. These production scenarios with distributed automation replace a large and expensive control with several small controllers connected through networks [9]. Thus, evolved programming approaches allow the implementation of flexible and distributed control systems, resulting in so-called CPSs [7], [10]. CPS integration is an essential prerequisite for enable smart manufacturing, furthermore, it is considered its core [7]. Collaborative CPSs enable concepts for the I4.0 paradigm and are usually named *Cyber-Physical Production Systems*[11].

A CPS is defined by the combination of an embedded system, that is, a physical part equipped with sensors and actuators, with a software information technology component, that is, a cyber part [12]. CPSs allow mechanisms to be controlled and/or monitored by computer algorithms. Those sequential instrutions can add new capabilities to physical systems using computation and communication. CPSs can interact, control processes, and have decision-making capabilities in problem-solving. The digital twin is another concept associated with CPS integration. Specifically, a digital twin can create high-fidelity virtual models of physical objects in virtual space to simulate their behaviors in the real world and provide feedback [13]. Therefore, CPSs are a key enabler for smart manufacturing [14]. The need for a distributed architecture and the requirement for flexibility has revealed a gap in modularity standards for the I4.0 [4], [13], [15]. Therefore, implementing CPSs requires using controllers that can also manage data be communicated through standardized protocols based on information technologies and that can also manage data in a manufacturing environment. Currently, most automation systems have a centralized control architecture, mainly supported by the IEC 61131-3 standard [16]. This norm has been widely used and accepted by the industry given its ease of transferring proprietary software models to an open model that allows the portability of applications. However, this standard has limitations for compliance with intelligent manufacturing requirements, not allowing the development of a high degree of distribution in control [15]. In that sense, the IEC 61499 standard [17] extends the IEC 61131-3 standard with decentralized control architecture. It allows portability, interoperability, configurability, reconfigurability, and distribution for the software design [18]. At the core of the IEC 61499 standard, we can find the function block (FB), which supports modularity, hence reusability. A FB can be considered as the representation of a component (encapsulating data and algorithms) and has a well-defined interface that allows it to communicate easily with other FBs (components) [19], [20]. With this standard, an application can be mapped to any number of devices. Thus, this domain-specific modeling language separates the application model from the hardware configuration, facilitating the design of distributed control applications.

A. Main Contribution and Research Question

This paper aims to define an approach that allows engineers and programmers to integrate systems with software (cyber) and hardware (physical) to design distributed automation applications based on the potential benefits of the IEC 61499 standard. Our goal is to help automation developers and practitioners to improve their design and therefore increase the development of distributed automation systems. These novel systems can be modeled by applying IEC 61499 FBs, in which control, communication, and CPS integration are covered in a graphical modeling language using industrial controllers' technologies, e.g., Programmable Logic Controller (PLC). Our research scope focuses in industrial automation areas such as flexible material handling systems, reconfigurable industrial automation, and intelligent distribution networks. For that reason, the offered solution is implemented in the specific domain of programming languages for field level control IEC 61499/ IEC 61131-3 standards. It is worth mentioning relevant related I4.0 standards such as DIN SPEC 91345 (Reference Architectural Model for I4.0), IEC 62264-1 (ISA 95), and IEC 62541 (OPC UA). However, such standards are beyond the scope of the manuscript, and the interested reader is directed to [11] for additional information.

Even though the features have made IEC 61499 increasingly recognized as an efficient way of CPSs modeling in the industrial automation domain [21], nowadays, the standard does not include guidelines on the design of such applications based on FBs. The implications of the development process are not always clear, in addition to the fact that existing models and methods cannot be easily combined. Formulating a design that implies creating a versatile approach regarding CPS is a big challenge. For this reason, this work addresses the following research question: *How do IEC 61499 models develop into a CPS for distributed control aligned with the I4.0 requirements?*

The outline of this paper is as follows. Sec. II presents related work on distributed control modeling techniques in intelligent automation based on IEC 61499. In Sec. III, the CPSs approach adopted with IEC 61499 is proposed. The application methodology for the CPS approach is described in Sec. IV in a case study using the platform nxtSTUDIO. Finally, the Sec. V provides the conclusions and future work.

II. Related Works

Software design and implementation methodologies enable the building of modular, reusable, and adaptable systems. Software accounts for more than 50% of the effort in building industrial automation systems [1]. Therefore, in the last decade, software design with IEC 61499 paradigms has been a notorious topic for the scientific community [22]. Collaborative CPS and digital twin efforts have focused on providing feasible methods, techniques, and guidelines to design distributed and intelligent automation systems. CPSs trends for I4.0 include control design of physical components, their digital twins, as well as the communication between them, applied in diverse areas such as medicine, smart buildings, industrial robots, process plant, and prognostics and health management systems, among others [23].

For control in industrial automation, it is possible to highlight some categories which belong to a design approach: Model-Driven Engineering, Component Based Design (CBD), and Design Patterns. These development paradigms are considered as the most potential means to reduce software complexity because they closely parallel the FB architecture in IEC 61499 and the notions of modularity and reusability [22]. Several works related to these techniques have been reported in the literature. Dai et al. propose in [24] a modeling language based on the IEC 61499 for industrial CPSs that is based on the use of Model-View-Control (MVC) object-oriented design. Cruz and Vogel-Heuser [15] analyze how both IEC 61131 and IEC 61499 address object-oriented programming. Black and Vyatkin [25] adopt a CBD that uses modular and accessible components with well-defined functions and predefined communication interfaces to implement distributed and intelligent control integrated with IEC 61499. The key parts of the design are using smart software components encapsulated in FBs. The proposed architecture is scalable and reconfigurable [22].

Regarding the last type of approach, Patil et al. [26] propose corresponding design patterns for IEC 61499: 1) structural (for FB types) and 2) architecture, composition, and behavior (for FB instances). These design patterns were applied in more than one case study to refactor FB applications in IEC 61499 over a distribution station, resulting in low complexity and high design efficiency [27]. The work presented in [3], [26], [28] show the concept of a simulation-in-the-loop environment, which reproduces the working behavior of the actual plant as a digital twin. This idea is then applied to control modeling with IEC 61499 standard, which helps to visually identify the behavior of the systems before deploying the control strategy in the actual hardware. Our work relies on these particular references in order to compare, from a conceptual point of view, our methodology with respect to state-of-the-art solutions.

Some other development patterns are oriented to function block application architecture and do not address approaches for designing applications with basic and composite types of FBs. The main idea behind this reasoning is how traditionally, automation engineers currently design and implement control software[1]. Several use cases of industrial CPS are presented in the scope of Industry 4.0 in [7]. The authors note that diverse development paradigms, requirements, and approaches to technological aspects such as digital twins and communications represent a great challenge for CPS providers in the industry. Such is the case of the CPS-based open architecture in [29] where vertical integration in flexible manufacturing in the oil and gas industry is presented. Table I summarizes the works referenced in this section, addressed by their main contribution.

From these designs, two aspects are observed in common: the first one is the modeling through IEC 61499 FBs with mapping, creation, composition, and execution of FBs as objects/components/services at different design levels. The second one corresponds to the interest in pursuing reconfiguration, reuse, and flexibility in manufacturing systems. In theory, IEC 61499 adopts programming for distributed systems architecture modeling with a higher level of abstraction. It is commonly applied in practice to incorporate system design, simulation, and validation supported by computing and networking technologies. Such situation, as a result, requires an emphasis on the development of standard adoption methodologies. According to [22], the industrial adoption of IEC 61499 in the context of the three phases of the "S" shaped curve, the standard was in the first phase of "Launch" when it was promoted by innovators before the 2nd Ed. of IEC 61499 published in 2012. Nowadays, it is transitioning to the second "Takeoff" phase associated with early adopters. Therefore, there is still a long way to go to reach the third "Maturity" phase until the main challenges are entirely solved. One of them is the generation of feasible methods, techniques, and guidelines to design systems based on IEC 61499, focused on the integration of I4.0 enabling technologies such as CPSs. Thus, it is of utmost importance in modeling next-generation industrial control for distributed and intelligent automation.

TABLE I
SUMMARY OF THE SELECTED LITERATURE REVIEW

References	Main contribution I	Main contribution F	Main contribution R	IEC 61499 application's sco
Black and Vyatkin [25]	++	++	-	E/V/R
Cruz and Vogel-Heuser [15]	+	+	+	E/V/R
Dai et al. [24]	+	-	+	M/A/D
Drozdov et al. [1]	++	++	-	M/A/D
García et al. [29]	+	-	+	M/A/D
Lyu and Brennan [22]	++	-	++	M/A/D
Patil et al. (a) [26]	++	-	+	E/V/R
Patil et al. (b) [27]	++	-	++	E/V/R
Xavier et al. [28]	++	-	+	E/V/R
This authors contribution	++	+	++	E/V/R

I: Intelligent distribution networks; F: Flexible material handling systems; R: Reconfigurable industrial automation; E/V/R: Execution, verification, or reconfiguration of IEC 61499. Modeling, architecture, or design issues in IEC 61499. ++: High; +: Medium; – Low.

III. ADOPTION OF IEC 61499 FOR CPS MODELING

CPSs integrate sensing, computation, control and networking into physical objects and infrastructure, connecting them to the Internet and to each other [7]. Recently, a CPS-based digital twin is introduced in the 5C architecture for I4.0, providing self-awareness and self- prediction [23]. Embedded controllers monitor and control manufacturing processes, usually with feedback loops, where industrial operations affect the computations and vice versa [30]. To achieve the control distribution goals from a CPS point of view, the IEC 61499 standard is adopted as the design paradigm, due to its orientation towards deploying modular and distributed control solutions [31]. A FB, representing a software functional unit (cyber part), associated with a hardware resource (physical part) of the control system, properly connected with more FBs can define an application, and the distribution of this can be configured among several devices of the control system, see Fig. 1. Thus, a device uses the relationship specified by the application to determine appropriate responses to events.

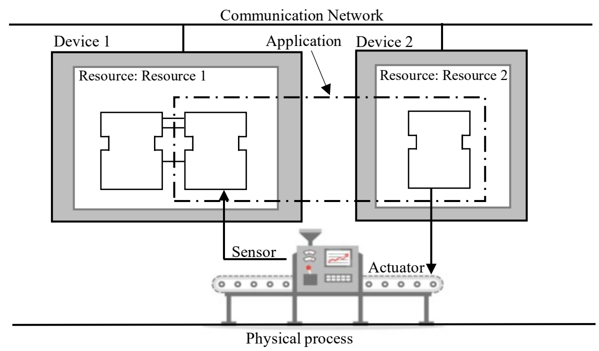

Fig. 1. System model in IEC 61499 standard. An application can be distributed through different devices communicated through a network and associated with physical components.

Furthermore, a resource is considered as a logical subdivision within the software structure of each device, which has independent control of its operations, as described by the IEC 61499 models in Fig. 2. Finally, each FB instance is associated with a single resource [21].

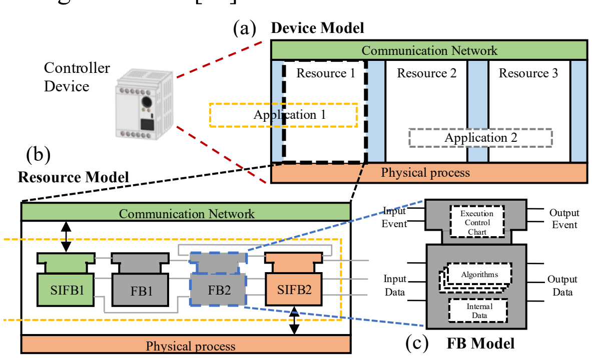

Fig. 2. IEC 61499 models within a system: (a) Device. (b) Resource. (c) Function Block.

With the above description, the automation architecture of a manufacturing system is modeled as a collection of devices, divided into resources, interconnected, and communicating with each other through one or more communication networks, which refers to the constitution of a CPS through the IEC 61499 standard. Fig. 3 shows the synergy of the simple structure of the CPS concept, adapted from [27], with IEC 61499 standard models. Here, two controllers interact with a physical process through sensors and actuators. The interaction between controllers and a digital twin is through a communication network, which enables the development of industrial applications for distributed control.

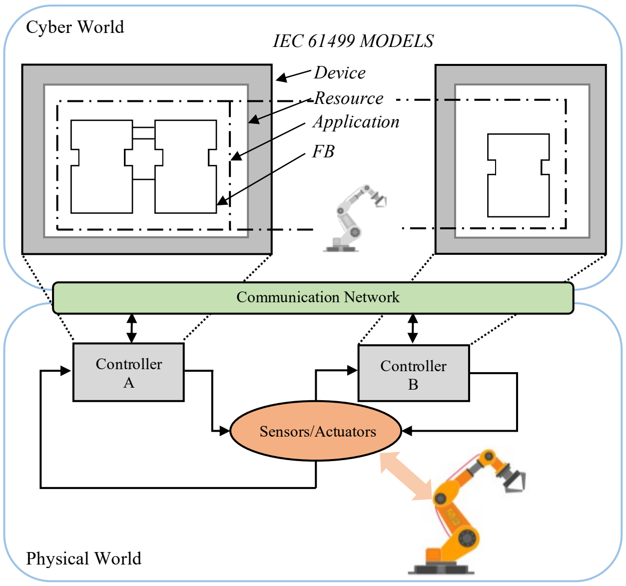

Fig. 3. CPS architecture for distributed control through IEC 61499 and their relations to the concepts of FB models.

IV. Application Methodology for CPS Approach

Five steps shown in Fig. 4 are envisaged to implement the proposed approach.

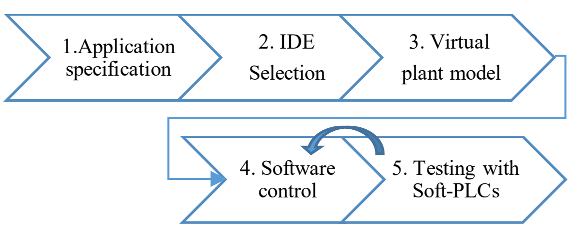

Fig. 4. Applied methodology steps for implementing the proposed approach.

When an industrial application has been specified (Step 1), we can select an integrated development environment (IDE) (Step 2) to virtually develop the process (Step 3). Then, software control must be set and distributed by FBs (Step 4). Finally, the system is deployed to several devices, e.g., Programmable Logic Controllers (PLCs), connected via a network. In this case, each PLC is considered using the so-called “soft PLC” device models, compliant with IEC 61499, which runs the RTE (Run-Time Environment), which is necessary to execute the part of the application deployed to the PLC. The RTE is also supporting the communication of the PLC with IDE. Therefore, CPS testing could be done (Step 5), and improved by the previous step.

A. System Specifications

A packet classification system by destination (local, national and foreign), as shown in Fig. 5, has been taken as an application case. The packages arrive at the system through the conveyor belt In_Conv, each must be transferred to the corresponding belt: L_Conv for packages with the local destination; N_Conv for national, and F_Conv for foreign. Each package has an external bar-coded label to identify its destination. In addition to the conveyor belts, the system consists of three double-acting pneumatic actuators: two horizontal (**L_Cyl**, **F_Cyl**) and one vertical (**N_Cyl**) that will serve as pushers to move the packages from one belt to another. There are also 15 sensors: seven of them are position detectors to report the location of the packages (P_In, P_L, P_N, P_F, PL_O, PN_O, PF_O) at the different points of the process belts; six sensors are associated to the positions of the ends of the three cylinders (XC_Retracted and XC_Extended for each actuator); two sensors are barcode readers (Loc_Sens and NoF_Sens) to determine the destination of the packages. Each reader has two possible output signals. Therefore, one reader determines whether or not it is Local and the other reader reports whether it is National or Foreign.

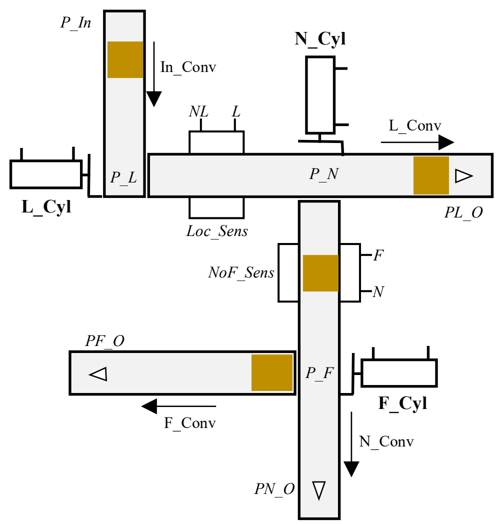

Fig. 5. Package classifier system that is comprised of three cylinders. Sensors are integrated through the four conveyor belts. Packages within the system are represented in brown squares.

Each actuator, together with its associated sensors, is considered a single system unit. Regarding the conveyor belts, it is considered that they are always active. Therefore, they are not considered in control, which means that the system to be controlled is made up of three components. The following is a list of requirements and conditions associated with the operation of the units:

1) Each cylinder has two input signals to indicate the type of movement to perform: X_Extend and X_Retract.
2) The cylinder **L_Cyl** is responsible for pushing the packet that reaches the end of the belt In_Conv (detected by P_In)

towards the L_Conv conveyor, where the Loc_Sens barcode reading system detects the boxes with Local destination.

3) The function of the **N_Cyl** cylinder is to transfer each packet that has been classified as non-local, and that is in front of it (detected by P_N), to the N_Conv belt, where the NoF_Sens barcode reader determines whether the destination is Domestic or Foreign.
4) Cylinder **F_Cyl** passes the non-domestic boxes to the F_Conv gang when they are in front of it (detected by P_F).
5) Each conveyor belt has a sensor at its end to detect when a package is at that point: P_L, PL_F, PN_F, PE_F are the respective sensors for each belt In_Conv, L_Conv, N_Conv and F_Conv.

B. *IDE Selection*

There are several IDEs for implementing the distributed control standard [32], among which the most reported works are: FBDK, 4DIAC, Profinet-CBA, ISaGRAF, and nxtSTUDIO [33]. All these platforms have served as programming tools for the design of control systems using the IEC 61499 standard and consist of a code editor, a compiler, a debugger, and some human-machine interface (HMI). Schneider Electric's nxtSTUDIO commercial platform seems to be nowadays the most IEC 61499-compliant reference for developing industrial applications, demonstrating that it is possible to design, generate the control, and create the graphical environment for the systems [22]. In addition, the control can be tested and debugged before running it on a real device, emulating a PLC. This means that it allows and is prepared to directly configure industrial equipment such as PLCs that comply with the IEC 61499 standard. It includes many libraries with prefabricated FBs designed to be used directly in typical control situations, allowing to generate and import FBs. Due to all aforementioned features, nxtSTUDIO has been taken as a development environment to perform the implementations in this work.

C. *Virtual Model*

Inspired by the preliminary work in [34], the suggested framework for developing distributed automation systems is the MVC object-oriented design. It is largely based on the closed-loop model architecture, where the plant is explicitly represented in the overall system model. This architecture allows easy integration with a simulation model for virtual commissioning, suitable for verification using models, which can subsequently be seamlessly converted to the real implementation configuration. The packet classification system proposed in Sec. IV.A is simulated by running a nxtSTUDIO Run-time on a PC, so that access to hardware inputs and outputs can be performed by using dedicated FBs to abstract the device level under the MVC design context. Fig. 6 shows the internal schematic of the Composite Automation Type (CAT) developed for each pneumatic cylinder. This CAT combines original function blocks (FBs) of the following types: four basic type, three composite, and three service interface type. We adopted two basic FBs: E_MERGE and E_CYCLE FBs, which are already available in the platform standard library. All these FBs with their respective linkages represent the operating model of a pneumatic component including its position sensors as well as their visualization.

By adding in the main application all CATs of the components involved (sensors and actuators), the complete model behavior of the package sorting system is internally implemented. On the left side of Fig. 7, all CAT instances developed that make up the application are shown, such as three cylinders together with their limit switches, the position detectors of the packages on the conveyor belts, as well as the target sensors. A short part of the interactive system visualization employing CATs is shown in the lower part. The CAT function block that represents the whole plant is shown on top portion. Once the model of the system to be controlled is assembled using CAT instances, it is possible to implement the control parts.

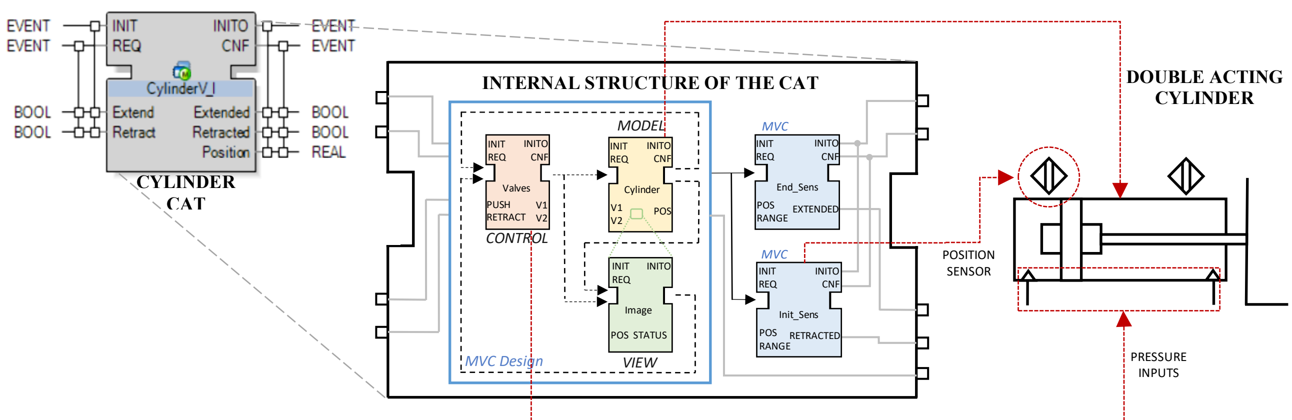

Fig. 6. *Model-View-Control* pattern used for developing the CAT structure of each pneumatic cylinder in the system classificator. There are other CATs inside the cylinder CAT that trepresent each part of the object: position sensors, pressure inputs, and the constitution of the double acting cylinder.

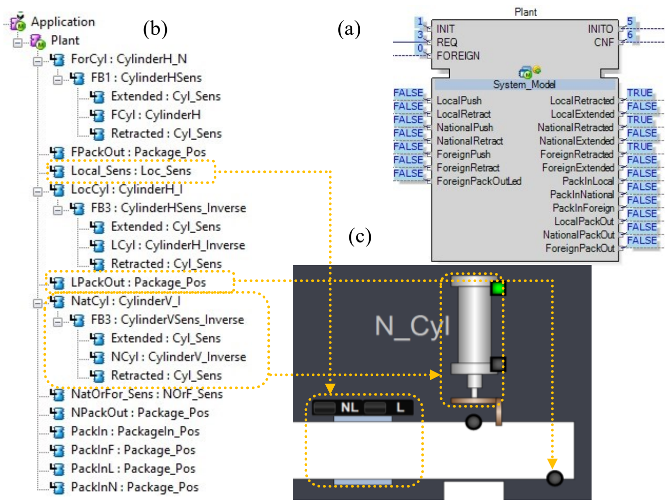

Fig. 7. Virtual model of the study case: (a) *System Model* FB that internally represents the package classificatory system. (b) CAT instances developed. (c) A short part of the resulting HMI.

D. Distributed Control

The CPS approach proposed in this paper for distributed systems contemplates the individual functionality of the pneumatic actuators, enabled by employing FBs instances included in a network. This represent a novel contribution in the domain of CPS's under consideration. In particular, by adopting this methodology individual components could be engineered as independent processes to suit different applications, reflecting the concepts of reusability, modularity, and adaptability. Moreover, their interaction is modeled in an abstract way by means of transferring events and data. Consequently, control signals can be sent from one component to another.

In this sense, within a basic FB in the IDE we developed and implemented an individual control algorithm for each cylinder according to its functionality. The three control FBs created were named as LocalCylinderControl, NationalCylinderControl, and ForeignCylinderControl. The second and third control FBs were generated from the first one by reusing its software functionalities. The control flow is adapted by changing event connections and the interface elements in the ECC.

With IEC 61499, each FB above is a control software unit containing algorithms and an execution control chart (ECC), with data and event inputs and outputs. When triggered by event inputs, the algorithms embedded in the FBs are invoked to perform the tasks and control commands. Subsequently, each one of these control FBs will be mapped and deployed into one of the emulate PLCs in the runtime environment available in the IDE. Thus, each device is responsible for managing the behavior of the mechatronic component associated with it. Fig. 8 the control FBs interface and the ECCs with their respective algorithms for Local and National cylinders. Therefore, for each cylinder to perform its respective task, it needs to communicate with the corresponding positioning sensors, as well as with the packages on the belts' position sensors, continuously requesting their status. After completing a task, an output event is generated from the FB, which serves as a notification/transition step so that a subsequent task/state can be executed in another FB. Control FBs interaction also occurs in such a way that LOCAL, NATIONAL, and FOREIGN events act as feedback to the command of another FB. This relationship between states and transitions is defined in the FBs internal finite state machine of ECC. Consequently, this link conditions the enabling or disabling of the adjacent FB.

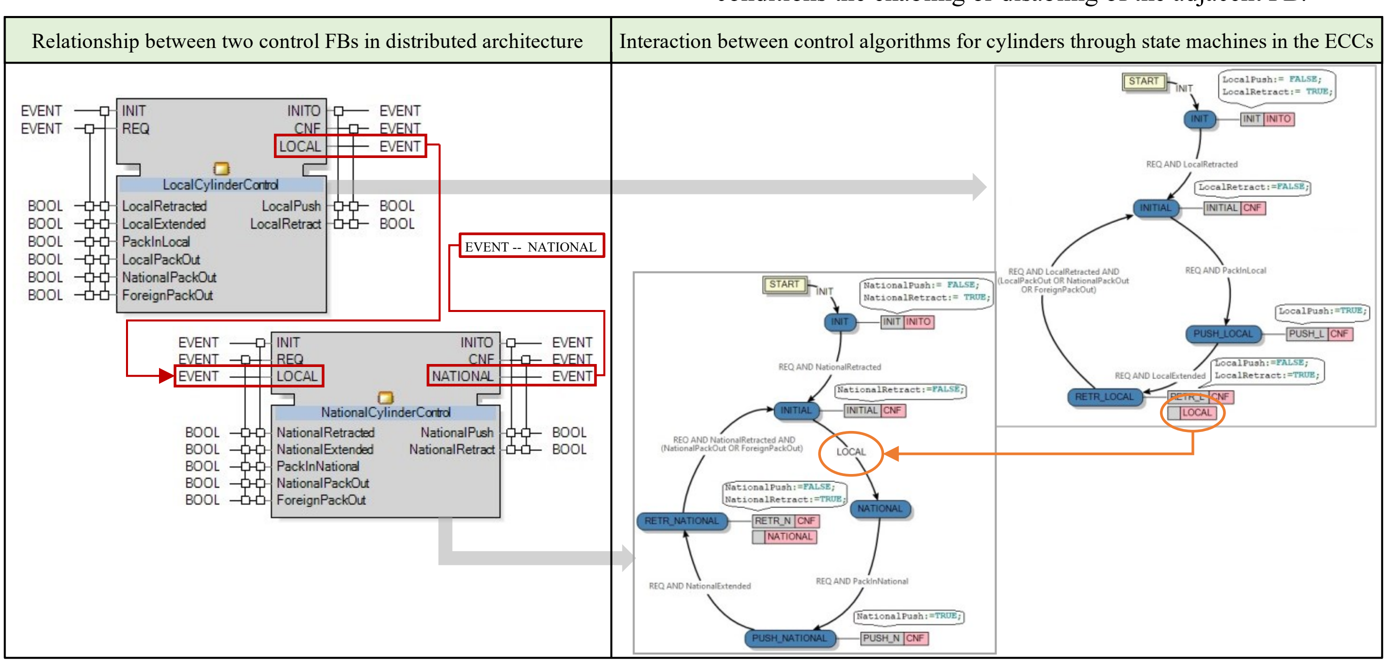

Fig. 8. Interaction between control FBs of the Local and National cylinders. LOCAL event output at LocalCylinderControl enables NationalCylinderControl through LOCAL event input. The ECCs for each controller and their respective algorithms are also shown. The LOCAL transition in the state machines reflects the interaction above mentioned.

For example, the operation of national cylinder **N_Cyl** is restricted to the transfer of a packet from In_Conv to L_Conv through local cylinder **L_Cyl**; this situation activates the LOCAL output event in the *LocalCylinderControl,* and this event is connected to the LOCAL input event of *NationalCylinderControl*. Such a relationship means that control units are connected side by side as a network, allowing pneumatic cylinders to communicate and interoperate with each other. In this context, a FB can be seen as an enabler to encapsulate the activation control of the in/out movements of the cylinders and monitor the packet classification during the execution of the whole system.

The proposed approach is general in the sense that it allows the reusability of FBs due to the fact that they can be interconnected into a network using event and data connections to specify the entire control application. Indeed, executing a personal FB-defined event-data interface and encapsulating local data and control algorithms make each FB a reusable software unit.

Furthermore, although the case study has been simplified for improving the clarity of the presentation, the proposed approach can be easily scaled to diverse processes with more components. This scaling holds considering that the method is applied for discrete manufacturing systems control.

E. Hardware Deployment Through Soft-PLCs

In order to properly run the resulting CPS approach with distributed control based on the IEC 61499 standard, run-times running IEC 61499 control FBs need to be appropriately mapped and deployed to soft-PLCs according to [35]. This deployment allows for emulation of real hardware applications, and in particular for our approach, the DCSmini PLCs from Schneider Electric were configured in the IDE. In nxtSTUDIO, the IP of the devices and runtime port numbers have to be configured correctly, since each IP port and number can only correspond to a specifically defined runtime. For our CPS solution, see Fig. 9, we run four nxtSTUDIO Run-times on a PC; one for the System Model presented in Subsection IV.C - which represents the actual process and is visualized as HMI in the PC- and three for the emulation of DCSmini controllers with IP address 127.0.0.1. The port numbers of these runtimes are 61499, 61503 61507, and 61511. In this way, each PLC can control the operation of its associated cylinder according to the corresponding mapping: Dev1_PC for System Model CAT, Dev2_DCS_mini1 for LocalCyliderControl FB, Dev3_DCS_mini2 for NationalCyliderControl FB, and Dev4_DCS_mini3 for ForeignCyliderControl FB.

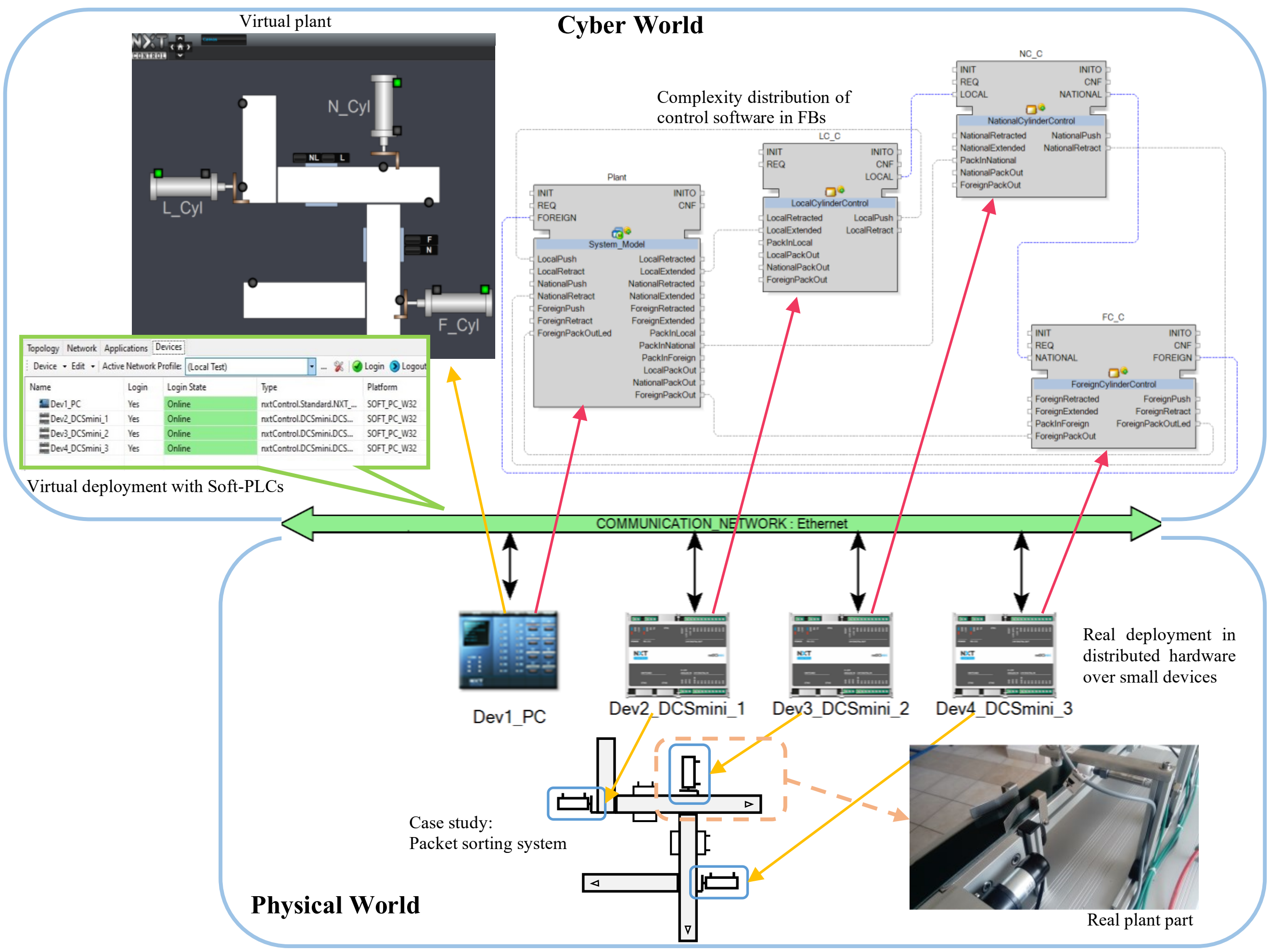

Fig. 9. Resulting CPS approach with distributed control of the package classifier through IEC 61499 system configuration. Each cylinder is controlled by an individual PLC, for which a control FB is associated (some connections in FBs distributed control are emitted for simplicity).

Since the HMI and control nodes communicate peer-to-peer, once executed the built application of the system using CATs and the FBs that make up the distributed control directly provide a complete interaction of the simulation model. Thus, data and event tracking of the complete application created in nxtSTUDIO IDE adequately reflect the performance of the implemented system, and the control functionality for the classifier was validated.

Fig. 9 also depicts the integration of the main elements in the proposed architecture. On one hand the cyber part is shown, where the complexity distribution of the control software is implemented with associated FBs. These Control FBs have connected adequately to System Model CAT FB that internally contains the virtual model of the plant. In the middle part of the CPS scheme, the Ethernet segment models the communication channel for all device instances. On the other hand, the physical part comprises one PC station, which shows the process HMI and three DCSmini controllers linked with their respective physical components. The physical connection of control hardware with sensors and actuators is not presented with great details. However, this configuration is easily enabled via the software libraries form nxtSTUDIO IDE such as NXT_RMTDEV_V1 that allows managing the input and output signals.

V. Conclusions and Future Work

This paper presents the close relationship of CPSs with the IEC 61499 standard, which enables a distributed way to design control applications in industrial automation. The proposed CPS modeling approach is better suited to the actual requirements of intelligent manufacturing systems and the I4.0 such as modularity, reusability, and adaptability. This is possible due to the capacity in our model for generating new –even more specialized– components from those already created. Specifically, this characteristic is enabled by reusing software functionalities by simply extending or adapting them with minimal modifications. The encapsulation of logic in the IEC 61499-based FBs reflects modularity in the system and is related to the definition of reusable models. Lastly, it is also important to highlight that this paper adopts nxtSTUDIO IDE as a programming tool for IEC 61499.

Since a CPS combines software control and communication with physical processes, the application shows that distributed automation systems can be successfully modeled in CPSs. By applying FBs of the IEC 61499 standard in conjunction with MVC design, models of automation systems in which control, communication, and physical processes coexist are covered in a graphical modeling language.

Most of the existing control hardware in manufacturing processes is PLC based. Thus, it is adequate to continue approaching the new requirements for smart manufacturing in line with the I4.0 concepts, such as modularity, flexibility, reusability, agility, adaptability, object-oriented programming, and others. As a future work, and taking into account the CPS perspective, the ubiquity of PLC technology in manufacturing, using OPC UA protocol to connect them to the cloud will be considered.

Acknowledgment

E. Monroy author thanks CONACYT-Mexico for the scholarship granted towards pursuing his PhD studies.

References

[1] D. Drozdov *et al.*, "Utilizing Software Design Patterns in Product-Driven Manufacturing System: A Case Study," in *Studies in Computational Intelligence*, vol. 853, 2020, pp. 301–312. doi: 10.1007/978-3-030-27477-1_23.

[2] M. de M. Fernandes, J. A. Bigheti, R. P. Pontarolli, and E. P. Godoy, "Industrial Automation as a Service: A New Application to Industry 4.0," *IEEE Latin America Transactions*, vol. 19, no. 12, pp. 2046–2053, Dec. 2021, doi: 10.1109/TLA.2021.9480146.

[3] T. Lyu, U. Dwi Atmojo, and V. Vyatkin, "Towards cloud-based virtual commissioning of distributed automation applications with IEC 61499 and containerization technology," in *IECON 2021 – 47th Annual Conference of the IEEE Industrial Electronics Society*, Oct. 2021, vol. 2021-Octob, pp. 1–7. doi: 10.1109/IECON48115.2021.9589945.

[4] R. Dintén, P. López Martínez, and M. Zorrilla, "Reference architecture for the design and development of applications for Industry 4.0," *RIAI - Revista Iberoamericana de Automatica e Informatica Industrial*, vol. 18, no. 3, 2021, doi: 10.4995/RIAI.2021.14532.

[5] B. Marschall, M. Schleicher, A. Sollich, T. Becker, and T. Voigt, "Design and Installation of an Agent-Controlled Cyber-Physical Production System Using the Example of a Beverage Bottling Plant," *IEEE Journal of Emerging and Selected Topics in Industrial Electronics*, vol. 3, no. 1, 2021, doi: 10.1109/jestie.2021.3097941.

[6] S. Cavalieri and M. G. Salafia, "Asset Administration Shell for PLC Representation Based on IEC 61131-3," *IEEE Access*, vol. 8, 2020, doi: 10.1109/ACCESS.2020.3013890.

[7] M. W. Hoffmann *et al.*, "Developing industrial cps: A multi-disciplinary challenge," *Sensors*, vol. 21, no. 6, 2021, doi: 10.3390/s21061991.

[8] B. Wiesmayr, L. Sonnleithner, and A. Zoitl, "Structuring Distributed Control Applications for Adaptability," 2020. doi: 10.1109/ICPS48405.2020.9274744.

[9] P. Jhunjhunwala and V. Vyatkin, "Proposing and Prototyping an Extension to the Adapter Concept in the IEC 61499 Standard," *IEEE Access*, vol. 10, pp. 2564–2577, 2022, doi: 10.1109/ACCESS.2021.3137642.

[10] G. Wan and P. Zeng, "An event-based programming model with geometric spatial semantics for cyber-physical production systems," *Applied Sciences (Switzerland)*, vol. 10, no. 21, 2020, doi: 10.3390/app10217651.

[11] L. A. Cruz S., D. Ryashentseva, A. Lüder, and B. Vogel-Heuser, "Cyber-physical production systems architecture based on multi-agent's design pattern—comparison of selected approaches mapping four agent patterns," *International Journal of Advanced Manufacturing Technology*, vol. 105, no. 9, pp. 4005–4034, Jul. 2019, doi: 10.1007/s00170-019-03800-4.

[12] B. Marschall, D. Ochsenkuehn, and T. Voigt, "Design and Implementation of a Smart, Product-Led Production Control Using Industrial Agents," *IEEE Journal of Emerging and Selected Topics in Industrial Electronics*, vol. 3, no. 1, pp. 48–56, Jan. 2022, doi: 10.1109/JESTIE.2021.3117121.

[13] F. Tao, Q. Qi, L. Wang, and A. Y. C. Nee, "Digital Twins and Cyber–Physical Systems toward Smart Manufacturing and Industry 4.0: Correlation and Comparison," *Engineering*, vol. 5, no. 4, pp. 653–661, Aug. 2019, doi: 10.1016/j.eng.2019.01.014.

[14] V. Vyatkin, "IEC 61499 as Enabler of Distributed and Intelligent Automation: State-of-the-Art Review," *IEEE Transactions on Industrial Informatics*, vol. 7, no. 4, pp. 768–781, Nov. 2011, doi: 10.1109/TII.2011.2166785.

[15] L. A. Cruz S. and B. Vogel-Heuser, "Applying Core Features of the Object-Oriented Programming Paradigm by Function Blocks Based on the IEC 61131 and IEC 61499 Industrial Automation Norms," in *Service Oriented, Holonic and Multi-agent Manufacturing Systems for Industry of the Future*, 1st ed., T. Borangiu, D. Trentesaux, P. Leitao, A. Giret Boggino, and V. Botti Navarro, Eds. Cham: Springer International Publishing, 2020, pp. 273–289. doi: 10.1007/978-3-030-27477-1_21.

[16] "Programmable Logic Controllers - Part 3: Programming Languages, Standard IEC 61131-3," Int. Electrotech. Commission, Geneva, Switzerland, 2013.

[17] "Function blocks for industrial-process measurement and control systems, Standard IEC 61499" Int. Electrotech. Commission, Geneva, Switzerland, 2012.

[18] G. Adamson, L. Wang, and P. Moore, "Feature-based function block control framework for manufacturing equipment in cloud environments," *International Journal of Production Research*, vol. 57, no. 12, 2019, doi: 10.1080/00207543.2018.1542178.

[19] L. A. Cruz S, J. H. Carvajal, O. A. Rojas, and E. Chacon, "Cyber-Physical System for Industrial Control Automation Based on the Holonic Approach and the IEC 61499 Standard," in *2018 Forum on Specification & Design Languages (FDL)*, Sep. 2018, vol. 2018-Septe, pp. 5–16. doi: 10.1109/FDL.2018.8524082.

[20] E. Monroy, L. R. García, L. A. Cruz, and B. D. Hernández, "Comparación de estándares IEC 61131-3 e IEC 61499 para implementar sistemas de control distribuido," in XXVIII Congreso Internacional Anual de la SOMIM, Bogotá, Colombia, 2022.

[21] D. Drozdov, V. Dubinin, S. Patil, and V. Vyatkin, "A formal model of IEC 61499-based industrial automation architecture supporting time-aware computations," *IEEE Open Journal of the Industrial Electronics Society*, vol. 2, 2021, doi: 10.1109/OJIES.2021.3056400.

[22] G. Lyu and R. W. Brennan, "Towards IEC 61499-Based Distributed Intelligent Automation: A Literature Review," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 4, pp. 2295–2306, Apr. 2021, doi: 10.1109/TII.2020.3016990.

[23] J. Lee, B. Bagheri, and H.-A. Kao, "A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems," *Manufacturing Letters*, vol. 3, pp. 18–23, Jan. 2015, doi: 10.1016/j.mfglet.2014.12.001.

[24] W. Dai, V. Vyatkin, C. Chen, and X. Guan, "Modeling distributed automation systems in cyber-physical view," in *2015 IEEE 10th Conference on Industrial Electronics and Applications (ICIEA)*, Jun. 2015, pp. 984–989. doi: 10.1109/ICIEA.2015.7334251.

[25] G. Black and V. Vyatkin, "Intelligent Component-Based Automation of Baggage Handling Systems With IEC 61499," *IEEE Transactions on Automation Science and Engineering*, vol. 7, no. 2, pp. 337–351, Apr. 2010, doi: 10.1109/TASE.2008.2007216.

[26] S. Patil, D. Drozdov, and V. Vyatkin, "Adapting Software Design Patterns To Develop Reusable IEC 61499 Function Block Applications," in *2018 IEEE 16th International Conference on Industrial Informatics (INDIN)*, Jul. 2018, pp. 725–732. doi: 10.1109/INDIN.2018.8472071.

[27] S. Patil, D. Drozdov, G. Zhabelova, and V. Vyatkin, "Refactoring of IEC 61499 function block application - A case study," 2018. doi: 10.1109/ICPHYS.2018.8390797.

[28] M. Xavier, S. Patil, and V. Vyatkin, "Cyber-physical automation systems modelling with IEC 61499 for their formal verification," in *2021 IEEE 19th International Conference on Industrial Informatics (INDIN)*, Jul. 2021, vol. 2021-July, pp. 1–6. doi: 10.1109/INDIN45523.2021.9557416.

[29] M. v. García, E. Irisarri, F. Pérez, E. Estévez, and M. Marcos, "Arquitectura de Automatización basada en Sistemas Ciberfísicos para la Fabricación Flexible en la Industria de Petróleo y Gas," *Revista Iberoamericana de Automática e Informática industrial*, vol. 15, no. 2, p. 156, Mar. 2018, doi: 10.4995/riai.2017.8823.

[30] E. A. Lee, "The past, present and future of cyber-physical systems: A focus on models," *Sensors (Switzerland)*, vol. 15, no. 3. 2015. doi: 10.3390/s150304837.

[31] A. A. Santos and A. F. da Silva, "Simulation and Control of a Cyber-Physical System under IEC 61499 Standard," *Procedia Manufacturing*, vol. 55, no. C, pp. 72–79, 2021, doi: 10.1016/j.promfg.2021.10.011.

[32] A. Hopsu, U. D. Atmojo, and V. Vyatkin, "On Portability of IEC 61499 Compliant Structures and Systems," in *IEEE International Symposium on Industrial Electronics*, 2019, vol. 2019-June. doi: 10.1109/ISIE.2019.8781290.

[33] nxtControl, "Schneider Electric - nxtSTUDIO," 2007. http://www.nxtcontrol.com/en/engineering/ (accessed May 15, 2022).

[34] E. Monroy Cruz, L. R. García Carrillo, S. Patil, and N. C. Hellman, "Adaptability in IEC 61499-Based distributed control for automation applications," presented at the IEEE 9th Int. Conf. on Industrial Engineering and Applications (ICIEA), Sanya, China, Apr. 15-18, 2022.

[35] A. Zoitl and T. Strasser, *Distributed Control Applications: Guidelines, Design Patterns, and Application Examples with the IEC 61499*. London, New York: CRC Press, 2016. [Online]. Available: https://books.google.de/books?id=vTo0CwAAQBAJ

Ernesto Monroy Cruz received the Master degree in Mechatronic from Polytechnic University of Pachuca, Mexico. From January to April 2014, he was a visiting scholar at the Unmanned Systems Laboratory from the Texas A&M University - Corpus Christi, USA. He currently is an assistant professor at the National Technological Institute of Mexico Campus Atitalaquia, and Ph.D student at Advanced Technology Center (CIATEQ), Mxico.

Luis Rodolfo García Carrillo received the PhD. degree in Control Systems from the University of Technology of Compiègne, France. He was a Postdoctoral Researcher at the Center of Control, Dynamical systems and Computation at UC Santa Barbara, USA. He currently holds an Assistant Professor position with the Klipsch School of Electrical and Computer Engineering at New Mexico State University, USA.

Luis Alberto Cruz Salazar received the Master degree in Electronic Engineering from Universidad del Cauca, Colombia. He is a Ph.D. candidate at the Institute of Automation and Information Systems at the Technical University of Munich. His main research interests are the design patterns for holons' and agents' development, as well as the Industry 4.0 by intelligent control software in *Cyber-Physical Production Systems.*