Fostering the creation of a Digital Ecosystem by a distributed IEC-61499 based automation platform

Andrea Barni*, Alessandro Brusaferri†, Franco A. Cavadini‡, Giuseppe Landolfi*,
Sandeep Patil§, Dario Piga*, Stefano Spinelli† and Valeriy Vyatkin§
* Scuola Universitaria Professionale della Svizzera Italiana - SUPSI, Manno, Switzerland
† Consiglio Nazionale delle Ricerche - CNR-STIIMA, Milan, Italy
‡ SYNESIS s.c.a.r.l., Lomazzo, Italy
§ Luleå University of Technology, Luleå, Sweden
Email: {name.surname}@{supsi.ch*, stiima.cnr.it†, synesis-consortium.eu‡, ltu.se§}

***Abstract*—Daedalus is conceived to enable the full exploitation of the CPS' virtualized intelligence concept, through the adoption of a completely distributed automation platform based on IEC-61499 standard, fostering the creation of a Digital Ecosystem that could go beyond the current limits of manufacturing control systems and propose an ever-growing market of innovative solutions for the design, engineering, production and maintenance of plants' automation.**

***Index Terms*—Distributed Control Automation, IEC-61499, Model Predictive Control, Digital Marketplace**

I. Context and motivation

Current worldwide landscape is seeing continuous growing value creation from digitization, with digital technologies increasingly playing the central role in value creation for the entire economy. More and more types of products are seeing a progressive transition to the "digital inside" model, where innovation is mostly related to the extension of the product-model to the service-model, through a deeper integration of digital representations. This means, concretely, that even in very "classical" domains the dissipation of borders between what is a product and what are the services that it enables is fostering a widespread "Business Model Innovation" need.

Within this context, the future of Europe's industry must be digital, as highlighted by Comm. Oettinger EU-wide strategy to "ensure that all industrial sectors make the best use of new technologies and manage their transition towards higher value digitised products and processes" through the "Leadership in next generation open and interoperable digital platforms" [1], opening incredible opportunities for high-growth of vertical markets, especially for currently "non-digital" industries.

Overall, the objective (and, thus, the motivation for Daedalus existence) is to bring Europe to a global leadership in "Digital Platforms" for industry. In fact, while on-line platforms are nowadays dominated by non-EU companies (Google, Apple, Amazon, etc.), the position of relevance of Europe in the manufacturing domains must be exploited to bring the next step in the evolution of those markets.

Considering the Commission high-level strategy but then looking at how global competition in the manufacturing sector is becoming fiercer and fiercer, with fast evolving requirements that must now take into account several concurrent factors, it is clear that European manufacturing companies have to focus the efforts on new automation solutions that could grant to the shop floor systems the flexibility and re-configurability required to optimize their manufacturing processes.

In this context, the question taken as premise to the conception of Daedalus framework is how the technological needs for the industrial automation sectors could be brought to convergence into a Platform capable of sustaining an Ecosystem and the corresponding Digital Market for the automation stakeholders. The core motivation for Daedalus was therefore born by the awareness that purely technological advancements in themselves are not enough to satisfy the need of innovation of the industrial automation market. New methodologies for the sector's main stakeholders to solve the new manufacturing needs of end-users must be conceived and supported by the creation of a technological and economic ecosystem built on top of a multi-sided platform.

In developing this concept, Daedalus has taken into account a certain number of fundamental "non-functional" requirements:

- The CPS forming interoperable devices must be "released" on the market together with their digital counterpart, both in terms of behavioural models and with the software "apps" that allows their simple integration and orchestration in complex system-of-systems architectures;
- The development of coordination (orchestration) intelligence by system integrators (more in general, by aggregators of CPS) should rely on existing libraries of basic functions, developed and provided in an easy-to-access way by experts of specific algorithmic domains;
- Systemic performance improvement at automation level should rely on well-maintained SDKs that mask the complexity of behind-the-scenes optimization approaches;
- Large-scale adoption of simulation as a tool to accelerate development and deployment of complex automation solutions should be obtained by shifting the implementation effort of models to device/system producers;

This translates into an explicit involvement of all main stakeholders of the automation development domain, brought together in a multi-sided market by Daedalus. Such "Automation Ecosystem" must rely on a technological platform that,

leveraging on standardization and interoperability, can mask the complexity of interconnecting these Complementors.
The following section therefore explores how the Deadalus Initiative has been tackling these high-level requirements through specific technological innovations. A new implementation model of CPS's automation based on the IEC-61499 international standard is presented in Sec. II, which paves the way for the deployment of more advanced real-time orchestration techniques, based on Model Predictive Control (Sec. III). This to finally converge into the concept of a Digital Marketplace for automation, together with the corresponding restructuring of the sector's business models (Sec. IV).

II. IEC-61499 IMPLEMENTATION OF CPS FOR AN IMPROVED MODEL OF DISTRIBUTED AUTOMATION

The Daedalus proposal of an Automation Ecosystem is based on a technological foundation: the extension of the international standard IEC-61499 with advanced functionalities for the deployment of a new generation of distributed intelligent devices (CPS). The functional model of Daedalus CPS, that blends coherently real-time coordination of its automation tasks with the "anytime" provision of services to other elements of the automation pyramid, enables the deployment of real-time CPS in production environment.
CPS embedding intelligence into distributed computational systems can provide both horizontal links with other components and vertical connection with business processes defying the rigid hierarchical levels of the automation pyramid towards its full virtualization.
CPS systems result in several challenges, such as interaction between the nodes, re-configuration, flexibility and distributed intelligence. In order to design such cyber-physical systems we need a good system engineering approach, enabled by use of IEC-61499 Function Block (FB) standard [2].

A. Cyber-Physical Components (CPC) Architecture

Modular design applies to the majority of automation systems mechanical structure and software, which is composed of components. In the context of the Daedalus project, software components have the following properties:

- *off-the-shelf:* reusable CPC, available either by buying from vendors, or as part of the standard library provided along with hardware equipment, or reused from another currently deployed system;
- *no modifications:* possible modifications implemented by deriving a new CPC;
- *self-contained:* emphasis on separation of concerns;
- *well-defined interface:* easily integrated with other CPCs;
- *supports hierarchy:* can be composed of other CPCs.

The concept of CPC, introduced in [3], and its implementation with FB encapsulate in a reusable component the control, simulation, and other functional elements related to a structurally distinguishable part of the system.

1) *Intelligent mechatronic component (IMC):*

introduced in [4], the IMC concept implies that machines or mechatronic components come with pre-programmed software, including components of model of the plant, components of control programs and any necessary network interfaces. Figure 1 shows an illustrative example of a pick-and-place manipulator, composed of three cylinders represented as IMCs. Its software is implemented as FBs of IEC 61499.
The CPC architecture refers to the software side of IMCs

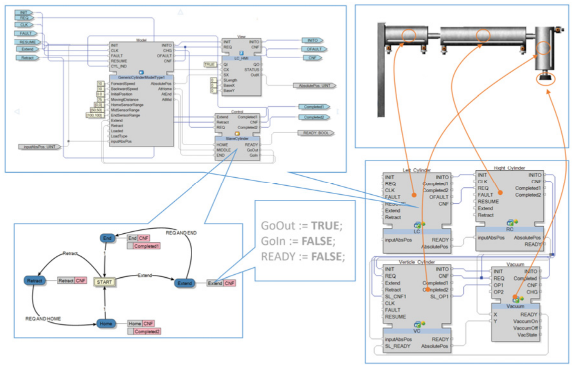

Fig. 1. (a) A pick and place robot, (b) Component based application, (c) Component hierarchy, (d) A single component - Basic function block ECC, (e) Algorithm inside basic function block. (HQ figure available online)

adapted in [5] to the domain of industrial automation and integrated with the IEC 61499 standard architecture. The building blocks of the architecture are CPCs integrating controllers and simulation models.
Daedalus' model of IEC-61499 CPS is not only an intelligent real-time executor of automation tasks: the full adoption of the CPS concept relies on the extension into the cyber-world and its integration with external modelling and simulation tools.
The plant is composed of a set of mechatronic components, denoted by M. Each component $\gamma \in M$ has corresponding software subcomponents $S(\gamma) = \{ctl, sim, hmi, view\}$ following the taxonomy of the CPC architecture [4]:

- $\gamma(ctl')$ is the CPC controller: it implements basic operations or services, invoked by a higher level controller (as in Sec. III), or a more sophisticated software agent capable of self-organization with other such components;
- $\gamma(sim')$ is a simulation model of the CPC to be used in closed-loop simulation with the control. The actual simulation model can be functionality implemented in an "external simulation environment", as shown in Figure 2, where each component $\gamma_i(sim)$ is represented by the corresponding simulation component $e(\gamma_i)$;
- $\gamma(hmi)$ implements the human-machine-interface (HMI) that provides interfacing to the component. The HMI is connected in closed-loop with $\gamma(ctl)$ and facilitates user interaction with the control application.;
- $\gamma(view)$ is a visualization of the dynamics of the simulation model, executing in closed-loop with the controller.

This methodology will mainly focus on the physical relationships between components. Topology of the plant configura-

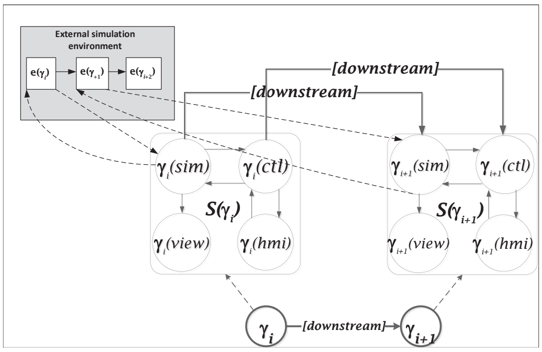

Fig. 2. Communication between two components with a downstream topological relationship.

tion can be expressed in the form of an attributed directed graph. Representing this information as a graph opens up the possibility of using a variety of graph transformation techniques [6] to assist in automatic software generation.

2) Composition of Components: Specific inter-component and intra-component communication is defined by the tuple $C = \langle R, Q \rangle$, where $R \subseteq S(\gamma) \times S(\gamma)$ specifies the intra-component communication between software sub-components $S(\gamma)$ within a single component γ. Figure 2 shows an example of two components in the downstream relation, including their internal architecture. For a single component, the set of communication defining relations is expressed as follows:

- $R(\gamma(sim), \gamma(ctl))$ and $R(\gamma(ctl), \gamma(sim))$: Bidirectional communication between simulation and control;
- $R(\gamma(sim), \gamma(view))$: Unidirectional communication between simulation and view representing state data from the simulation passed to the view for rendering;
- $R(\gamma(ctl), \gamma(hmi)$ and $R(\gamma(hmi), \gamma(ctl))$: Bidirectional communication between control and HMI representing control panel interfacing with the controller application.

Communication between two components γ_i and γ_{i+1} is shown in Figure 2, an arc connects these two components and is assigned with the downstream relation attribute. As a result of applying this pattern, the model of the system's behaviour can be designed with nearly complete reuse of the component models. A detailed works on applying many other design patters using IEC-61499 is presented in [7] and work [8] presents how to refactor existing FB application using the design patterns described in [7].

III. SDK FOR DEPLOYMENT OF MPC-BASED ORCHESTRATOR

The real time orchestration of CPC architecture requires advanced control solutions. Daedalus platform extends the capabilities of current IEC-61499 FBs to enable the deployment of an orchestrating intelligence implementing multi-disciplinary optimization algorithms. The de-facto standard approach for real-time optimal control of constrained highly complex systems is the Model Predictive Control (MPC) [9]. MPC cyclically solves a constrained optimization problem to compute the future inputs by predicting the behavior of the system, within a finite horizon, based on a dynamical model. Only the first control move is applied to the plant and then the procedure is repeated to provide close-loop capabilities. The design of MPC solutions requires advanced mathematical and control engineering skills for the identification or development of the system dynamical model, the definition of the optimization problem and its solution within the cycle time. Despite the existence on the market of MPC solutions, e.g. Mathworks MPC ToolBox™, and industrial products, e.g. Siemens SIMATIC PCS 7 APC® or Rockwell PlantPAx®, no solution exists within the IEC-61499 framework.

Daedalus has conceived the Software Development Kit (SDK), Fig. 3, to provide a set of tools and utilities to help automation system engineers to synthesize a MPC solution for aggregated CPS and its integration in IEC-61499 framework. The SDK packages are constructed based on object-oriented programming paradigm. This enables the simple re-usability of tool and modules, the creation of custom libraries for verticalization on specific sectors/applications as well as further extensions. The SDK is composed of three main fundamental elements, defined as the Online System Identification (OSI), the Online Control Modeller (OCM) and the Online Control Solver (OCS).

A. Online System Identification

The OSI block of the SDK is dedicated to infer a model of aggregated CPS from input and output data. It absolves two main functions:

- *batch identification*, where the model of the system is estimated offline using a set of historical input-output data generated by the system and gathered from dedicated experiments or based on past activities of the system;
- *adaptive identification*, where the model of the system is updated recursively and in real time based on current data. The adaptive identification block thus requires no data storage, and it allows the model to be constantly updated and adapted to follow a possibly time-varying behaviour.As for the initial model, it can be either specified by the user, or it can come from batch identification.

The OSI block is based on *PieceWise Affine autoRegressive with eXogenous inputs* (PWARX) model structures. PWARX models belong to the more general class of hybrid dynamical models, and they are defined by partitioning the space of regressor (defined by past input and output measurements)

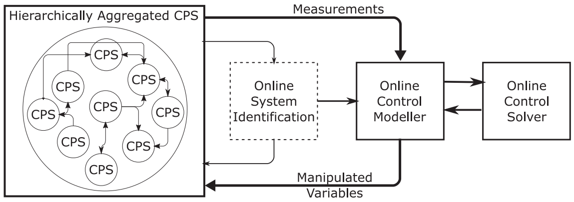

Fig. 3. Daedalus SDK components: OSI for system identification, OCM for the definition and development of optimization problem and OCS for the solution of the optimal control problem.

into a finite number of polyhedral regions, and by considering an affine dynamical submodel on each polyhedron. Thanks to their universal approximation property [10], PWA maps can accurately approximate any nonlinear function, preserving a good compromise between flexibility and simplicity.

PWARX models are also particularly suited for the MPC design implemented in the OCM block of the SDK. Indeed, using PWARX as predictive models allows us to formulate the MPC problem in term of convex *quadratic programming* (QP), provided that the discrete-value sequence defining the affine sub-models which are active within the prediction horizon is known, and thus it is not optimized. Otherwise, the solution of the MPC problem can be computed using numerical solvers for *mixed-integer quadratic programming* (MIQP) solvers.

The OSI module implements the two-stage PWA regression algorithm recently proposed in [11] and based on multi-model recursive least-squares (which are used to estimate the parameters of the local affine models) and multi-category discrimination methods (which are used to compute a polyhedral partition of the regressor space). The implemented identification approach processes data in a recursive way, and thus it can be used both for batch and real-time identification.

B. Tools for Optimal Control Problem

Daedalus SDK includes libraries to design and implement different categories of predictive solutions covering a wide range of system classes: from linear to nonlinear and PWA systems. The OCM module contains a set of classes for model-based and optimization-based control, ranging from linear to time-varying MPC.
The `OCM_MPC` class is the backbone of the OCM: it contains the definition of the linear MPC controller object and a collection of methods for the design, development and deployment of MPC solutions. The OCM object provides all the features and attributes that need to be configured: default values are imposed for basic users, but all the fields are accessible for complete customization by advanced users, e.g. definition of time-varying affine constraints.
Methods encapsulate functions and routines that can be easily called and integrated during the development of a control solution, e.g. for the implementation of MPC in velocity form for tracking problems, the configuration of a Kalman filter estimator. SDK methods hides the complexity of the algorithm, requiring just the configuration of the object and function parameters, thus lowering the knowledge entry-barriers for advanced control applications. Once the MPC object is configured, the control loop code reduces to call the `OCM_MPC` method to evaluate the next control move.

Linear MPC is the most diffuse approach for MPC, in which the prediction is based on a Linear Time Invariant (LTI) model; however for more complex systems, i.e. nonlinear systems or PWA models, a unique LTI model can be not enough.
The OCM implements more advanced schemes, denoted `OCM_AdaptMPC` and `OCM_LTVMPC`: these are child classes that inherit attributes and methods from the linear MPC class and extend them to implement the Adaptive and Linear Time-Varying (LTV) controllers [12]. Both schemes cyclically linearise the system to obtain a good approximation of system dynamics. These MPC formulations lead to a Quadratic Programming (QP) problem to be solved at each iteration, Fig. 4. Besides, leveraging on extensibility of the OCM functionalities, hybrid MPC problems can be formulated, e.g. importing system dynamic matrices generated by external tools.

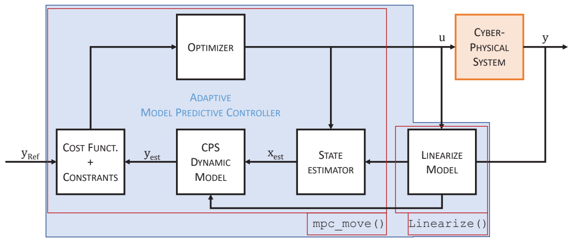

Fig. 4. Schematic of the Adaptive MPC block of the OCM

The Online Control Solver contains the routines that are executed online to translate the MPC problem in the corresponding QP and to compute its solution. The OCS provides mathematical solvers both for QP and for NonLinear Programming (NLP), to solve more general OCP. A QP solver [13] and NLP line search algorithms [14], based on Levenberg-Marquardt method and BFGS, are implemented. These internal solvers give the user a comprehensive MPC tool within the SDK platform and the possibility for the advanced user to have access to the subroutines for extensive customization. On the other hand, OCS implements methods enabling the usage of external solvers (open-source, e.g. `quadprog`, or commercial solvers, e.g. Gurobi®) to empower the end-user to address different types of optimization problems (e.g. MIQP problems for hybrid systems).

C. IEC-61499 function block for MPC applications

For the integration of advanced control solutions within a distributed automation platform, Daedalus SDK introduces a MPC-based CPS Orchestrator Function Block that provides optimal control capabilities. The Orchestrator FB is configured through SDK methods, while in execution such function block calculates in real-time the sequences of automation tasks to be performed by exchanging synchronization signals (as events and data) required for task execution. Daedalus approach exploits the characteristic composability of the FB architecture to conceive aggregation of equipment and devices to generate more complex systems supported by a progressive orchestration of their behaviour.

An application of the IEC 61499 MPC-based FB for coordination of CPSs is presented in [15] on a lab-scaled industrial plant for the end of life treatment of electronic boards.

IV. Digital Marketplace for the Creation of an Interdependent Ecosystem of Automation Solutions Providers

Digital platforms have been widely adopted as instruments to support the diffusion of product-related services, reducing transaction costs and facilitating exchanges that otherwise would not have occurred [16]. The main value that the platforms create is the reduction of the barriers of use for their customers and suppliers. As for suppliers and producers, platforms create significant value also for consumers, providing ways to access to products and services that they have not even been imagined before. The platform developed within the Daedalus project follows this trend by extending platform logics to the automation domain. This is done exploiting as a foundation the new evolution of the IEC-61499 standard that envisages the technology on the top of which additional value drivers for the automation complementors can be set up.

CPC components and architectures, distributed applications, optimal intelligent algorithms, virtual avatars of behavioral models and runtime services within the cyber-world find their aggregation point in the Daedalus platform, where most of the complementors of the automation ecosystem can host hardware related software modules in an integrated and coherent platform ecosystem [17]. To support the adoption of these components in the deployment of real applications in the shop floors of the future, Daedalus envisions the creation of a Digital Marketplace (MP) where, thanks to the adoption of a common, standardized and interoperable Automation Language, developers and users can match offer and demand of IEC-61499 based hardware and software solutions in a simple way.

The technological solution, a digital MP, is conceived with the aim of creating the ecosystem for the concurrent integration and market deployment of mechatronic systems, CPS and digital twins. While developing this technical architecture solves many issues by creating a seamless user experience, on the other hand, managing the content of Daedalus Automation App Store will still pose a certain number of challenges. In particular, the aspects related to the organization and description of Software applications within the MP, and how to make them available and discoverable, represent the main technological issue to make the MP an effective tool supporting the matchmaking among demand and offer.To address this need, a data model aimed to cover, at one side, an exhaustive description for the involved products and, at the other, all the economic aspects of the products in term of prices, contracts, etc., has been designed. The here proposed data model fully characterize an app of the ecosystem in order to be accurately searched and identified. For that reason, such data model has to consider aspects like provided functionalities, compatibilities, specifications, meaning of the application I/O, application extensibility, description of the logics that the automation application wants to provide and the openness of its source code. The diagram, designed by means of UML class diagram specifications (Fig. 5), is structured into five different sections: (i) *User Characterization*: data entities for user description and characterization; (ii) *Product Description*: data objects needed to describe the products (hardware, application and services) hosted by the MP; (iii) *Contract Definition*: entities needed to formalize all possible configuration options of the contract that regulates the economic aspects between the MP and the customers about the use of the products; (iv) *Validation and Certification*: to formalize those entities meant to support the mandatory validation of the product and the optional certification. The main processes managed by the MP base their own logics onto the here described data model.

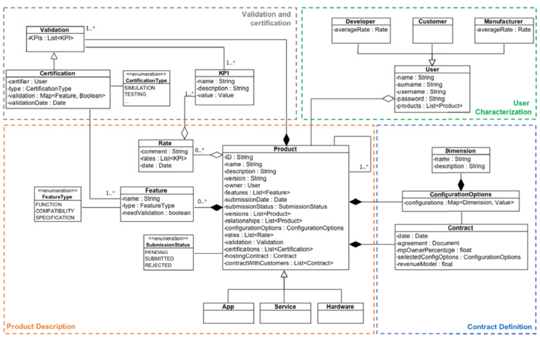

Fig. 5. Daedalus Marketplace UML class diagram. (HQ figure online)

A. Product submission process

This process is meant to manage the entire submission process of a new product made by a developer/manufacturer, starting from the submission request, passing through the product validation, till to make the product available. The process starts with the submission request of the product sent by the developer or manufacturer. The submission request needs to be accompanied by the selection of the configurable hosting contract, specifying the options of the hosting service and the kind of validation/certification. The nature of the contract establishes the rules that regulate the economic relationship between the developer/manufacturer and the MP in managing the product hosting. The product submission also includes the pricing strategies definition which allows the submitter to specify how the MP has to manage contracts of the products usage between the MP itself and the customers.

The MP integrates a set of validation services, provided by third-parties actors, meant to test the products and validate their declared functionalities. Once the testing phase has been completed, the product certification can be performed: the certifier is an accredited body that uses the test results in order to certify a certain product. The tester not necessarily assumes the responsibility for certifying the product.

The MP provides a search engine mechanism based on a set of algorithms meant to result the best possible answer to a search query. The search, based on a semantic engine, aims to improve search accuracy by understanding the customer's intent and the contextual meaning of terms. Semantic search systems consider various points including context of search, intent, variation of words, synonyms, meaning, generalized and specialized queries, concept matching and natural language queries to provide relevant search results.

B. Business Model and Pricing Strategies

During the submission process, the submitter needs to specify how the MP will manage potential contracts of the products usage between the MP itself and the end-users. The submitter can select a set of dimensions on which the strategy template can be based on (i.e. license-based, fee-based, number of updates based, etc.). In order to better understand possible contract strategies, some examples are here proposed:

Case A: A machine (and its related IEC-61499 application control) is host on the MP. A plant owner who needs to integrate it into an existing plant can buy it directly through the Daedalus MP and download the software app from the MP. To receive continuous application updates or access to additional value-added functionalities provided by the application, the MP becomes its contact point;

Case B: A manufacturer needs an existing IEC-61499 application which provides a certain functionality. This application, available on the MP, can be bought by the manufacturer, integrated and sold with his own control solution. Here, the manufacturers pay for using the application. The developer may provide the application source code, or the compiled one;

Case C: A manufacturer wants to develop a system composed by a set of sub-systems, where each of them is controlled by a single, already existing, control application. The manufacturer needs to develop a IEC-61499 supervising controller application able to orchestrate the overall system. The SDK enables manufacturer to access a set of optimization functionalities and to define, for each sub-system, the mathematical models and develop the supervising controller application. The usage of the SDK may require expert intervention for developing the application on manufacturer behalf. In case, he can require an ad-hoc development directly to the MP, which is in charge to regulate the transactions also for consulting activities, as the controller development.

The reference use cases described above create a value/monetary flow that interests the MP and enables it to be economically profitable. In particular, since the MP is the matchmaker among subjects that offer a product/service and subject that require it, two main revenue models are considered: (i) application of a percentage fee on product/service transactions billed on product/service supplier; (ii) application of a fixed hosting fee for the products/services hosted on the marketplace.

V. Conclusions

Daedalus initiative has proposed a multi-sided "Automation Ecosystem" to satisfy the need of innovation of the industrial automation market. Leveraging on standardization and interoperability, new methodologies and technologies are provided and embraced to realize such vision. The cornerstone is the adoption and the extension of the IEC-61499 industrial standard for the deployment of distributed intelligent devices. The functional model of Daedalus IEC-61499 CPS permits to fully exploit the digital counterpart of CPS, enabling additional functionalities, like multi-level orchestration intelligence or the access to non-real-time services. IEC-61499 interoperable CPCs can be so designed and combined in more complex reconfigurable mechatronic systems, both in the real and cyber (simulated) world. To address the design of hierarchically distributed optimal control applications, the Daedalus SDK is presented. This provides open and reusable system identification libraries to construct and learn the behavioral model of the CPS and MPC classes to simplify the implementation of optimal control applications. To guarantee and sustain the development of this Ecosystem, a Digital Marketplace is envisioned where Automation Complementors can jointly contribute to the creation of added value for the end-user, within the context of sustainable business models.

Acknowledgment

DAEDALUS has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement N° 723248.

References

[1] European Commission, "Digitising european industry." available on: https://ec.europa.eu/digital-agenda/en/digitising-european-industry.

[2] 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, 2011.

[3] G. Zhabelova, C.-W. Yang, S. Patil, C. Pang, J. Yan, A. Shalyto, and V. Vyatkin, "Cyber-physical components for heterogeneous modelling, validation and implementation of smart grid intelligence," in *IEEE International Conference on Industrial Informatics (INDIN)*, pp. 411–417, 2014.

[4] V. Vyatkin, "Intelligent mechatronic components: control system engineering using an open distributed architecture," in *IEEE Conference on Emerging Technologies and Factory Automation (ETFA)*, pp. 277–284, 2003.

[5] J. H. Christensen, "Design patterns for systems engineering with IEC 61499," 2009.

[6] U. Ryssel, J. Ploennigs, and K. Kabitzsch, "Generative function block design and composition," in *2006 IEEE International Workshop on Factory Communication Systems*, p. 253, 2006.

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

[8] S. Patil, D. Drozdov, G. Zhabelova, and V. Vyatkin, "Refactoring of IEC 61499 function block application - a case study," in *IEEE International Conference on Industrial Cyber-Physical Systems (ICPS)*, 2018.

[9] J. B. Rawlings and D. Q. Mayne, *Model predictive control: Theory and design*. Nob Hill Pub. Madison, Wisconsin, 2009.

[10] L. Breiman, "Hinging hyperplanes for regression, classification, and function approximation," *IEEE Transactions on Information Theory*, vol. 39, no. 3, pp. 999–1013, 1993.

[11] V. Breschi, D. Piga, and A. Bemporad, "Piecewise affine regression via recursive multiple least squares and multicategory discrimination," *Automatica*, vol. 73, pp. 155–162, 2016.

[12] S. Gros, M. Zanon, R. Quirynen, A. Bemporad, and M. Diehl, "From linear to nonlinear mpc: bridging the gap via the real-time iteration," *International Journal of Control*, pp. 1–19, 2016.

[13] C. Hildreth, "A quadratic programming procedure," *Naval Research Logistic Quarterly*, vol. 4, pp. 79–85, 1957.

[14] J. Nocedal and S. Wright, *Numerical Optimization*. Springer, 2006.

[15] S. Spinelli, A. Cataldo, G. Pallucca, and A. Brusaferri, "A distributed control architecture for a reconfigurable manufacturing plant," in *IEEE International Conference on Industrial Cyber-Physical Systems (ICPS)*, pp. 673–678, 2018.

[16] P. Muñoz and B. Cohen, "Mapping out the sharing economy: A configurational approach to sharing business modeling," *Technol. Forecast. Soc.Change*, 2017.

[17] G. Landolfi, A. Barni, S. Menato, F. A. Cavadini, D. Rovere, and G. Dal Maso, "Design of a multi-sided platform supporting cps deployment in the automation market," in *IEEE International Conference on Industrial Cyber-Physical Systems (ICPS)*, p. 684–689, 2018.