

## Research Article

# On the Design Complexity of Cyberphysical Production Systems

Luis Ribeiro  and Martin Hochwallner 

*Department of Management and Engineering, Division of Manufacturing Engineering, Linköping University, 58183 Linköping, Sweden*

Correspondence should be addressed to Luis Ribeiro; [luis.ribeiro@liu.se](mailto:luis.ribeiro@liu.se)

Received 23 February 2018; Accepted 9 May 2018; Published 10 June 2018

Academic Editor: Dimitris Mourtzis

Copyright © 2018 Luis Ribeiro and Martin Hochwallner. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Establishing mass-customization practices, in a sustainable way, at a time of increased market uncertainty, is a pressing challenge for modern producing companies and one that traditional automation solutions cannot cope with. Industry 4.0 seeks to mitigate current practice's limitations. It promotes a vision of a fully interconnected ecosystem of systems, machines, products, and many different stakeholders. In this environment, dynamically interconnected autonomous systems support humans in multifaceted decision-making. Industrial Internet of Things and cyberphysical systems (CPSs) are just two of the emerging concepts that embody the design and behavioral principles of these highly complex technical systems. The research within multiagent systems in manufacturing, by embodying most of the defining principles of industrial CPSs (ICPSs), is often regarded as a precursor for many of today's emerging ICPS architectures. However, the domain has been fuzzy in specifying clear-cut design objectives and rules. Designs have been proposed with different positioning, creating confusion in concepts and supporting technologies. This paper contributes by providing clear definitions and interpretations of the main functional traits spread across the literature. A characterization of the defining functional requirements of ICPSs follows, in the form of a scale, rating systems according to the degree of implementation of the different functions.

## 1. Introduction

The need and pursuit for highly adaptable systems are not an exclusive endeavour of Industry 4.0 and more generally of what is understood as the 4th Industrial Revolution. The motivation can be at least traced back to the aftermath of the 1970s oil shock when mainly European and American companies strongly invested in technology and particularly in the so-called flexible manufacturing systems (FMS) [1], at the same time Japanese companies were developing lean manufacturing principles. Even if the early incarnations of FMS were relatively unsuccessful, the need went on as mass customization became the excellence paradigm in production [2].

Mass customization works generally in economies of scale with relative stable markets. However, at a time when unpredictable market demands are accompanied by almost continuous and fast-paced innovation processes and production sustainability is the new excellence paradigm, the existing automation practices that had for some

years supported mass customization processes are starting to subside.

For more than two decades now, researchers have turned to distributed computation practices and artificial intelligence in the quest for new metaphors for developing, designing, and implementing more adaptive production systems [3, 4]. The idea of reconfigurable manufacturing systems (RMS) [5], as a manufacturing system that is designed at the outset for rapid change in structure, as well as in hardware and software components, in order to quickly adjust production capacity and functionality within a part family in sudden changes in market or in regulatory requirements, emerged almost in parallel. Along also came the concepts of holonic manufacturing systems (HMS) [6] and bionic manufacturing systems [7].

Collectively, these paradigms share quite a few important design principles and ideology. Among them is the notion that systems should be modular. Modularity is interpreted therein in both physical and logical terms. Standardized interfaces between the system's modules are of paramount

importance to ensure functional composability and scalability. The modules themselves should denote a higher degree of autonomy, and their functionalities are generally self-contained, a principle that in some branches of robotics became known as embodiment [8] and that has very close ties to the current cyberphysical conceptualization.

The quick expansion of information technologies (IT), especially those related to computer networks, in the late 90s to early 2000s has paved the way for the Industry 4.0 (I4.0) as web service-related technologies and service-oriented architecture started to penetrate and dominate industrial IT infrastructures [9, 10]. Even if they did not fundamentally change the ongoing enterprise processes, and most notably left nearly untouched the field level control layers of production facilities, they have opened up for a more interoperable use of information.

From there, several relatively competing paradigms have developed, namely, industrial Internet of Things (IIoT), cloud automation/manufacturing (CA), cyberphysical production systems (CPPS), and industrial cyberphysical systems (ICPS). It is difficult to pinpoint exactly which properties pertain exclusively to which. It is also unclear if some are manifestations of others. The so-called 4th Industrial Revolution, or Industry 4.0, seems to be the overarching umbrella for a lot of scientific and technical activities that are generally anticipated to lead to the next generation of automation systems. So much so that this anticipation has motivated several research efforts globally [11]: the H2020 Framework Programme for Research and Innovation under the Factories of the Future Public-Private Partnership (EU); the Industrial Internet Consortium, created by AT&T, Cisco Systems, General Electric, IBM, and Intel in 2014 (US); the Made in China 2025 initiative (CN); and globally many other initiatives.

The authors position this paper within the CPPS/ICPS thematic, detailed in greater extent in the subsequent section, which they perceive as the involving paradigm for most of the activities occurring within the factory of the future. As with all the other paradigms, there is resounding fuzziness in concepts and technologies and their relative roles in the creation of next-generation automation systems. There are various ideas regarding the functions and reference architectures of such infrastructures, and these are spread and presented in different contexts.

In the previous context, the present paper develops along several contributing lines. Drawing from reference literature in the fields of multiagent systems (MAS) applied to manufacturing, CPPS/ICPS, and to a lesser extent from cloud automation and more general literature in CPSs, the paper synthesizes and functionally formulates the zero-tier requirements that have been attributed to CPPSs in the context of I4.0.

The requirement formulation is accompanied by a corresponding structural model positioned at the same abstraction level.

Both the requirements and the structural model are then discussed in greater detail and are the base for developing a scale that enables the classification of CPPSs. Such scale ranges from a system not fulfilling the minimum requirements to

be convertible to a minimal CPPS implementation up to how a full featured CPPS would fulfil the synthesized requirements.

The authors trust that such a scale can be used as an indicator of preparedness for tackling the challenges that are believed to be workable for a system implemented under the premises of I4.0. Simultaneously, the proposed scale can be used as a tool to inform subsequent system design and implementation directions towards target levels or functional goals.

In the remainder of this paper, the methodology considered during the research presented herein is detailed in the next section, followed by the main results and discussion. The paper finishes with the main concluding thoughts and pertinent research questions worth of a follow-up.

*1.1. Materials and Methods.* This work's methodological approach is based on traditional requirements engineering (RE), sometimes also called requirement management (RM). RE/RM is a systematic and disciplined approach to the specification and management of requirements [12] and in part a discipline of systems engineering [12, 13].

A requirement is therefore understood as [12]

- (i) a condition or capability needed by a user to solve a problem or achieve an objective;
- (ii) a condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed documents;
- (iii) a documented representation of a condition or capability as in (i) or (ii).

Requirements can be elicited from many different sources [13]: documents, systems in operation, and stakeholders. Stakeholders are here understood as a person or organization that has a direct or indirect influence on a system's requirement [12].

The stakeholders are here addressed by their roles representing relevant concerns. General stakeholders for CPPSs are the law, the product user, the plant customer including the sales channel, the procurement channel for parts/material (material inflow), the equipment operator/worker, the production engineer (process, quality), the plant operator, the plant owner, the equipment maintenance/installation, and the equipment manufacturer.

Another source of requirements is documents. In the context of this paper, relevant documents are laws, standards, and publications. This work contains a literature review and consolidation based on publications in the area. The selected literature was obtained from a systematic analysis of publication arising mainly from members of several technical communities, namely, the following IEEE technical committees: industrial agents and industrial cyberphysical systems. The selection was guided by a tool developed to analyse vast networks of coauthorships partially described in [14]. The authors assume that to an important extent, the surveyed publications are also reflecting the interests of the many different stakeholders. The authors also accept that this vision is

debatable; however, given the novelty of the domain and the degree of speculation around it, previously peer-validated information entails an important level of consensus around the views expressed that otherwise would be harder to attain.

At the same time and on the very same line, this contribution is intended to serve as source of requirements upon which other contributions can be developed which will eventually consider more matured views from other stockholders.

One particular important differentiating factor from conventional RE/RM lies in the fact that the present paper does not concentrate on one specific system but rather on an entire class of systems. Scope is normally defined as [12] the range of things that can be shaped and designed when developing a system. As such, it should be evident that the present contribution has a metalevel positioning due to its wider scope.

Further on this, potential CPPSs inherit the requirements for existing systems in operation, and any particular design derived from it will have to comply with, at least, similar requirements applicable to its predecessor.

This contribution assumes these previous requirements but focuses on the novel aspects that are perceived to pertain particularly to the CPPS domain. When there was an alignment of requirements but the CPPS formulation supposed an important reinterpretation, then the requirement was included in the discussion.

The starting point for the subsequent discussion is the adapted CPPS structural models presented in [15, 16]. Of particular interest is the definition of CPPS (Figure 1) understood here as an aggregation of human resources, cyberphysical production modules (CPPM) (defined next), subsystems, and aggregated products towards which it establishes one or several cyberphysically formulated interaction interfaces. These interfaces are used for monitoring and control of the CPPS operations as well as to tap into the knowledge generated both by the human resources, the CPPMs, and the subsystems during the production process as well as knowledge generated by its aggregated products throughout their life cycle. This internal knowledge is used in different time scales to continuously improve operations and to inform the strategic consumptions of capital, raw materials and energy in their many forms. Cyberphysically formulated products will also generate value for external systems, as part of networks of things and services, towards which they maintain interaction interfaces. The outcome of such interactions is external knowledge. Access to it may be offered as a value-adding service that can potentially help to further improve CPPS operations.

The subsequent CPPM definition adheres to the notion of module as detailed in [17] which states that a module is tightly coupled within and loosely connected to the rest of the system.

The CPPM definition (Figure 2) is adapted from and envisions a CPPM as [16] a module consisting of three logically aggregated entities: an equipment, a controller or computing platform, and a cyberrepresentation of this whole. The computing platform may be shared between several cyberrepresentations if it provides access to the equipment that they represent. The cyberrepresentation contains both

the interface and the algorithms that enable the module to interact with other modules, human resources, or subsystems without the need for reprogramming it and implements a hardware abstraction layer that decouples the interaction and execution logic from the equipment details.

The authors will consider these starting structural formulations throughout the paper to support the discussion of the design complexity of CPPS across the many different step-wise implementations.

## 2. Results and Discussion

*2.1. Engineering Requirements from the Specialized Literature.* The specialized literature offers many important insights into the characteristics that belong to a CPPS. However, these many different characteristics have been defined at equally many abstraction levels. In general and as previously discussed in [16, 18, 19], scientific contributions have therefore ranged from hardware in the loop implementations up to reference architectures. This means that the expectations that are posed upon system realization also vary substantially. There is also normally a disconnection between reference architectures, where desirable characteristics are embedded in a conceptual design, and their subsequent realization that often does not adhere to the conceptual principles.

The distance between conceptual design and concrete implementations is also aggravated by the fact that many different characteristics are normally specified without a degree of obligation other than a mention that the characteristic is desired.

In the following analysis, the authors, sustained by a survey of selected literature, proceed with formally specifying requirements accordingly to [13] and also, but to a much lesser degree, following a top-down decomposition approach proper of axiomatic design processes [20].

The practical implications of this approach is that all the synthesized CPPS characteristics are expressed in the form object + degree of obligation + function and that each identified characteristic is at least expanded by one level in depth. They are also formulated, to best effort degree, according to independence axiom that postulates that an optimal design always maintains the independence of functional requirements [20]. The objects of interest are drawn from the CPPS/CPPM definitions presented before. The degree of obligation is an enumeration with three possible values with the following semantics [13]:

- (i) Shall: the object must implement the function attributed to it.
- (ii) Should: the object will, if possible, implement the function attributed to it.
- (iii) Will: the object may come to implement the function attributed to it at some point in the future.

The selected literature reflects an extremely wide set of “desirable” characteristics of a CPPS. A considerable number of these are using different nomenclatures for similar

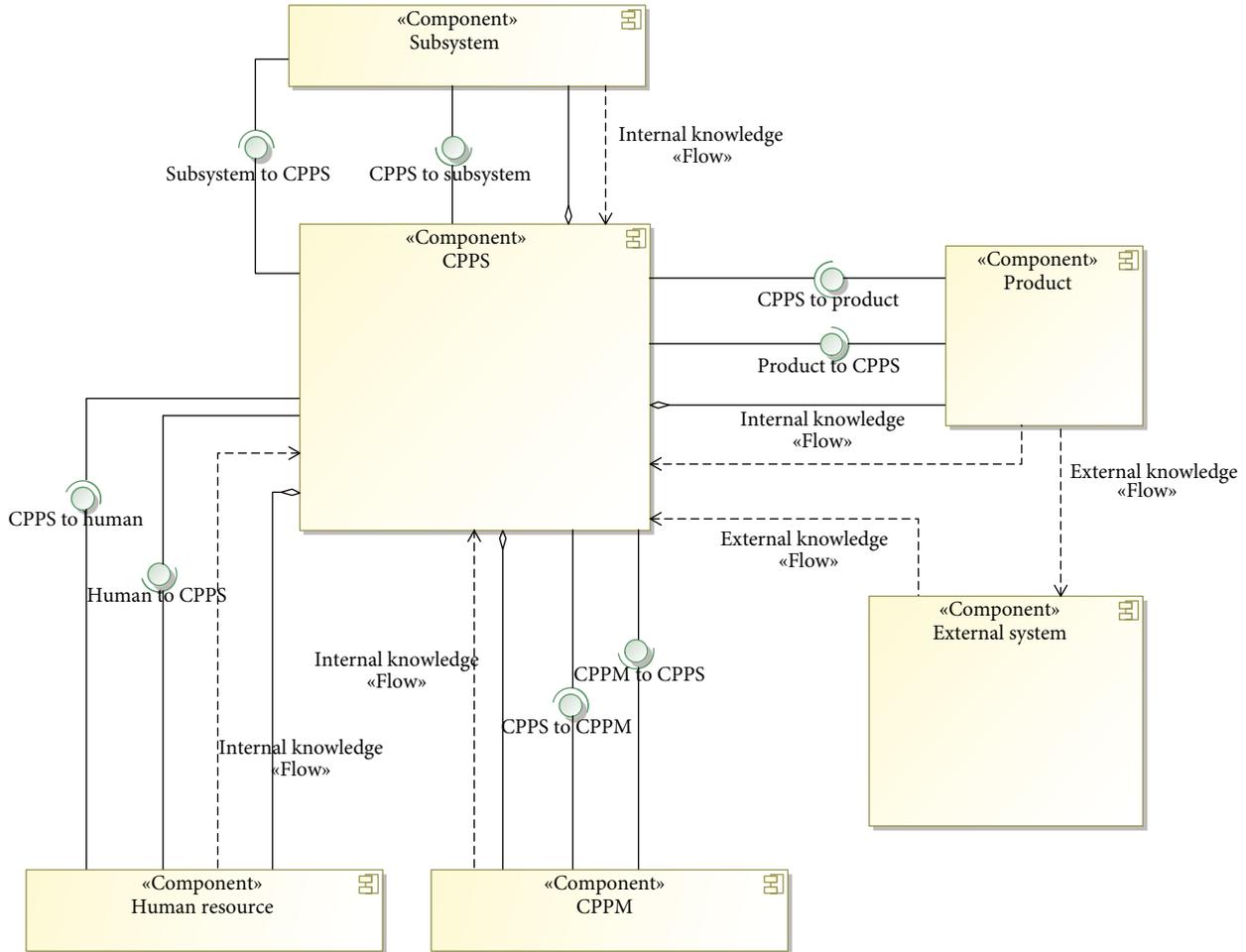


FIGURE 1: CPPS structural model (adapted from [15]).

functions. In the scope of this work, the authors have clustered sufficiently close attributes into the following top-level requirements: adaptability, convertibility, integrability, predictability, usability, diagnosability, safety, and security. Their top-level definition (level 1) provides typically a (re)formulation that is closer to the normal interpretation of similar concepts but is already functionally specified. The subrequirements (level 1.X) provide further details that conduce to and inform about the top level. A few requirements are very close or even taken from other authors. These are properly identified. Some of the nomenclatures were also taken from previous works on reconfigurable manufacturing systems, in particular [5, 21, 22], but unless otherwise stated has been reinterpreted and formulated in a functional way.

From the top-level requirements, the authors would like to discuss first the three that they believe are the main differentiating requirements within a CPPS: adaptability, convertibility, and integrability (Figure 3). At the core of almost all research agendas in I4.0 lies the idea of constantly adapting systems. Adaptation can happen in structure, function, or both. As such, adaptation can only be implemented if the system components can be integrated with each other (integrability). It additionally requires a relative modular physical

structure (convertibility) to support a wide scope of adaptive solutions beyond simple functional adaptation.

The extent to which different adaptability, convertibility, and integrability functions are implemented is a direct measurement of the degree of how cyberphysical, at the light of I4.0, a production system really is.

The tables below (Tables 1–3) reflect the results of the literature survey and the reformulation of many of the spread and sparsely defined desired characteristics for these three main requirements. The related literature sources include only the ones that have more strongly stated or otherwise contributed to the proposed RE/RM exercise. Similar concepts and definitions will of course be found in other papers in the specialized literature but they are either contained by the cited sources or otherwise had generally weaker requirements.

The adaptability requirements (Table 1) focus mainly on the expectations on system behavior. It has been accepted for many years now that the ability to adapt to changing conditions is of paramount importance for production systems. The notion of CPPS seems to encompass also the possibility of structural adaptation whereby mobile equipment can even change, in a more or less autonomous way, the factory layout. The authors set a strong emphasis in this ability to adapt.

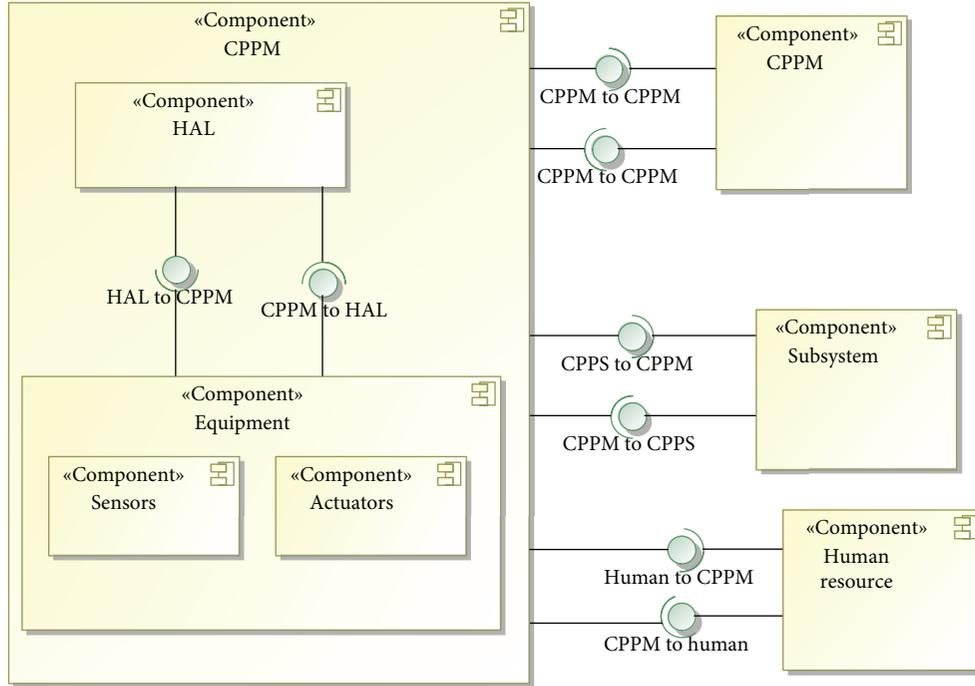


FIGURE 2: CPPM structural model (adapted from [16]).

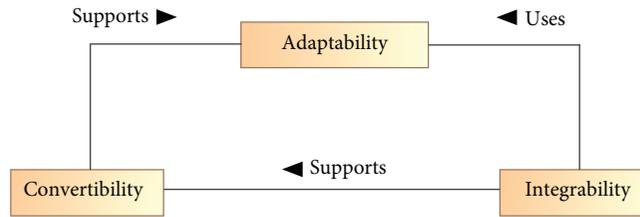


FIGURE 3: Associations between the core requirements in CPPSs.

TABLE 1: Synthesis and reformulation of requirements on adaptability.

Requirement/ subrequirement	(Re)formulation	Related literature sources
Adaptability 1	The CPPS shall adapt its behavior and structure to changing conditions.	[24–33]
Adaptability 1.1	The CPPS shall have the perception of environmental elements and events with respect to time or space, the comprehension of their meaning, and the projection of their status after some variables have changed, such as time, or some other variables, such as a predetermined event [34].	[24, 34]
Adaptability 1.2	The CPPS should learn its future behavior from past experiences.	[24]
Adaptability 1.3	The CPPS shall adapt in the most optimal way possible.	[35]
Adaptability 1.4	The CPPMs and subsystems should collaborate with other CPPMs, subsystems, and human resources in the same system during operation.	[24, 36, 37]
Adaptability 1.5	The CPPS shall autonomously attempt to overcome faults or otherwise support users in this process.	[26, 29, 38]

Conditions that would cause a system to adapt could vary in their emergency and implementation time frame and naturally include faults and failures, surges, or drops in production as well as the ramp up of new products. The literature suggests a high degree of obligation in the implementation of such functions. This reason has justified the use of the obligation operator “shall” in most requirements in Table 1. The

two exceptions would be adaptability 1.2 where learning is mentioned and adaptability 1.4 where collaboration is stated.

The justification for the lower degree of obligation in adaptability 1.2 relates to the fact that this is something that the specialized literature mentions quite frequently; however, learning implies the generation and availability of a substantial amount of data when applied at a system level. On

TABLE 2: Synthesis and reformulation of requirements on convertibility.

Requirement/ subrequirement	(Re)formulation	Related literature sources
Convertibility 1	The CPPS shall have a modular physical structure that enables the rapid adjustment of its production functionalities.	[21, 22]
Convertibility 1.1	The CPPM shall provide reusable physical interfaces that enable it to participate in distinct production contexts.	[21, 22, 33, 35, 37]
Convertibility 1.2	The CPPS should provide interfaces that enable the physical composition of its CPPMs.	[6]

TABLE 3: Synthesis and reformulation of requirements on integrability.

Requirement/ subrequirement	(Re)formulation	Related literature sources
Integrability 1	The CPPS shall provide mechanisms to integrate new heterogeneous components and systems that are unknown to it.	[24, 36, 37]
Integrability 1.1	The CPPS will connect to other CPSs and extract information from them that is used to support its operations.	[24, 26, 29, 36]
Integrability 1.2	The CPPM and subsystems shall provide reusable logical interfaces that enable the integration of new equipment.	[6, 24, 29, 31, 38]
Integrability 1.3	The CPPS shall use as the basis of its operation a domain specific language that in a uniform way enables CPPMs, subsystems, and human resources to describe their capabilities, operational requirements, and status to the CPPS.	

systems designed on the onset for frequent change (both physical and logic), acquiring significant data may prove to be a challenge. In this context, adaptability 1.2 is in principle important but may not be implementable in the general case.

Collaboration (adaptability 1.4) is frequently cited as the preferred mechanism to support adaptation. However, many other forms of interaction [23] are possible, and collaboration may not be desirable at all times. In some cases, competitive interaction strategies may, for example, attain better results. This is normally context dependent.

Table 2 shows the convertibility requirements. If adaptation is reflected on how the system behaves, then convertibility is about the physical characteristics of the system that ultimately allow it to make use of its adaptive behavior. Modularity is greatly recognized as the prevailing characteristic, and a CPPS should ensure that its components can be combined in different ways to adapt and generate new functions when required. It therefore requires from its CPPMs a minimal level of mechanical interfacing and compatibility.

Convertibility alone is only providing guarantees of mechanical conformity. Surfaces, cabling, power lines, and so on should match. Integrability covers therefore the logical interoperability of the system. Cyberphysical system must be put together along these two dimensions. The degree of obligation is very high and supposes the existence of logical abstractions that enable a CPPS to interpret and use the functions of equipment that were previously unknown to it (integrability 1). These include the usage of machine interpretable languages and compatible logical interfaces (integrability 1.2 and integrability 1.3).

In the future, if the vision of I4.0 fully consolidates and becomes a reality, the scope of action of CPPSs will extend

beyond just a cluster of CPPSs and may come to encompass other CPSs that relate to it (integrability 1.1).

As mentioned before, this contribution assumes that the existing requirements applicable to today's production system will also to a certain extent apply to CPPSs, and the authors do not inspect or address them in their full range. However, from these, there are a few that may require a fundamentally different approach or otherwise are challenges, in particular predictability, usability, diagnosability, safety, and security.

Predictability (Table 4) has been traditionally interpreted in two ways: the behavior of the system is transparent and explainable and the system behaves in a time-predictable way. The authors believe that the first should be inherent to any adaptive process in the industry and therefore interpret predictability more towards the second direction (time predictably as in predictability). The obligation degree here is maximum, and the general contract is that the CPPS delivers a performance that is compatible with the processes under its control (predictability 1.1). This does not necessarily mean hard real-time performance but means a behavioral delivery that enables planning and asserts production quality (predictability 1.2 and predictability 1.3).

Predictability is generally related to usability (Table 5). No matter how adaptable the system is, without a proper set of control tools that enable the straightforward operation of the system (usability 1), CPPSs cannot be accepted. The novelty in CPPS tools lies in that they must exploit and support the adaptive capabilities of the system and hence enable the design of products along the entire range of possibilities covered by the convertibility and integrability possibilities of a given setup (usability 1.1).

TABLE 4: Synthesis and reformulation of requirements on predictability.

Requirement/ subrequirement	(Re)formulation	Related literature sources
Predictability 1	The CPPS shall operate in a time-predictable way.	
Predictability 1.1	The CPPS shall operate at a time frame that is compatible to that of the physical processes it seeks to control.	[24, 36, 37]
Predictability 1.2	The CPPS shall operate in a way that enables short-, medium-, and long-term planning of production activities.	
Predictability 1.3	The CPPS shall produce with predictable quality.	[38]

TABLE 5: Synthesis and reformulation of requirements on usability.

Requirement/ subrequirement	(Re)formulation	Related literature sources
Usability 1	The CPPS shall provide tools for efficient development of diverse production processes.	[29, 35]
Usability 1.1	The CPPS shall provide tools that enable the production of the full number of variants that could be enacted considering the available resources and their potential configurations.	[21, 22]
Usability 1.2	The CPPS shall provide information that is tailored to the needs of specific users' roles.	[24, 36, 39]

In the scope of supporting operations, CPPS must also ensure that operators are being informed at the right time about the relevant events taking place in the system. But the amount of events in a CPPS is, due to its adaptive behavior, overwhelming. As such, the system must have the ability to filter and direct only the relevant information required to support the different decision-making processes (usability 1.2).

The ability to understand what a complex system is doing has ramifications not only to its normal operation but also, at least as importantly if not more, to its behavior under disturbances. CPPS shall attempt to detect and diagnose faults and failures within. Given their complex and compositional structure, this may be a challenge depending on the size and the intricacy of the underlying interactions. The obligation here is to do a best effort in diagnosability (Table 6) and while operating in a fully traceable way. In a CPPS context, traceability means to a great extent make a set of complex causality flows understandable.

Safety (Table 7) is one of the emerging concerns within the CPPS-related literature. It is an absolute obligation that has so many different dimensions at the moment that make it very difficult to characterize. Today's safety regulations apply mainly to fixed installations and conventional automation solutions. The regulations are scarcer when considering autonomous machines and even more when considering mobile autonomous machines that may interact with humans. Most human-machine interactions are, in an industrial context, assessed on an individual case.

The success of CPPS and I4.0 in general requires that safety regulations are extended and come to include, in the general case, situations where autonomous systems, performing structural adjustments in a production floor, can be accepted and validated.

Security (Table 8) is a more tangible endeavour in CPPSs. The obligation is to execute in a way that protects the CPPS from harmful influence from its components or external systems. Most of the readily available supporting

technologies are generally poor when it comes to fulfilling security requirements. While this problem has already been proven a challenge in running systems, it is aggravated in the context of CPPSs due to their general low maturity.

*2.2. Grading CPPSs according to Their Degree of Implementation.* Different system designs will adhere to the requirements specified before to a different extent. Some systems may not adhere at all. This creates a range of possibilities that collectively inform about the ability of evolving a specific system or system design in order to tackle the emerging challenges envisioned by I4.0.

The authors therefore propose a scale that distinguishes between the different stages at which a system or system design may be. The proposed scale ranges from  $-1$  to  $5$ , where  $-1$  would apply to a system that, for different reasons, cannot be transformed to reach a CPPS level of functional delivery and a  $5$  would apply to a full-featured CPPS. Such scales are relatively common in the software development domain, for example, the Capability Maturity Model Integration (CMMI) rates software development organizations and guides their development process in respect to the way that software development process is conducted.

While the authors think that such a fully developed model would be extremely complex to develop in the scope of CPPSs due to the level of maturity of the entire field, they believe that similar ideas can be usefully developed considering the requirements earlier detailed. The discussion of the different levels is therefore presented next. A conceptual exemplary system is discussed along each level to better illustrate evolve-ability challenges and potential.

Level 0—at this level, substantial engineering effort would be required to evolve the system to CPPS-grade functionalities.

- (i) Adaptability: the production system does not support adaptation. There are practical ways to revamp the system to become adaptable but substantial engineering effort is necessary.

TABLE 6: Synthesis and reformulation of requirements on diagnosability.

Requirement/ subrequirement	(Re)formulation	Related literature sources
Diagnosability 1	The CPPS shall whenever possible autonomously detect and diagnose the root causes of product defects [21] or otherwise actively support users in their identification.	[21, 22]
Diagnosability 1.1	The CPPS shall operate in a traceable way.	[35]

TABLE 7: Synthesis and reformulation of requirements on safety.

Requirement/subrequirement	(Re)formulation	Related literature sources
Safety 1	The CPPS shall operate in a safe way.	[24, 40]

TABLE 8: Synthesis and reformulation of requirements on security.

Requirement/ subrequirement	(Re)formulation	Related literature sources
Security 1	The CPPS shall operate in a secure way.	[24, 36, 38, 40]
Security 1.1	The CPPS shall not allow harmful interference from other CPSs it interacts with.	
Security 1.2	The CPPS shall provide secure access to its functions and modules.	

- (ii) Convertibility: the equipment is generally not modular and/or follows essentially a mechanical monolithic structure. Parts within the equipment can be replaced, maintained, or serviced but replacing does not lead to change in function.
- (iii) Integrability: the equipment is not connected through standardized interfaces in a consistent way and many different interfaces and interaction protocols are in place. Integration is handled in a case by case basis.
- (iv) Other requirements: the system fulfils the minimum level of requirements for operating according to the production objectives and applicable regulations.

A semiautomated workshop with several workstations and/or machine tools is a typical scenario for a level 0 system. In such a scenario, equipment is generally not integrated with each other. Production is moved from the workstation/machine to workstation/machine by a manual process or otherwise by a fixed and indexed automated transportation solution. The same production processes are always used at the same workstations/machines or otherwise reconfigured by an operator that manually introduces or selects new programs already available from the workstations/machines. More advanced machines may also automatically act upon the presence of workpieces if they are electronically tagged in a way that enables a specific machine to execute a program for that specific workpiece. The immediate steps that should be taken to improve such a system, bringing it to the next level, include improving its integrability. As legacy equipment is frequently found in such environments, it is generally not straightforwardly possible to use more advanced communication protocols, for example, web service-based protocols. Adapter devices will have to be considered in these cases

that create a harmonizing layer over legacy components and act as command interpreters and translators. These adapter devices are also usually a prerequisite for improved adaptability. Convertibility issues will need to be addressed over time by replacing the physical equipment with more modular equipment where applicable.

Level -1—at this level, it is not practically possible to evolve the system.

- (i) Level -1 is defined after level zero since its starting point is the same; however, there are either legal restrictions (e.g., regulatory or IP protection) that block any further system integration effort. Equipment or systems exist in the form of self-contained isolated islands.

Level 1—at this level, the system is in principle modular and some of these modularity considerations have been incorporated at design time to cater for potential long-term changes.

- (i) Adaptability: the production system is programmed to cover all the foreseeable cases where decisions may need to be taken and interactions between components are static and permanent.
- (ii) Convertibility: the equipment is generally modular and it is possible to replace equipment with some with similar capabilities. These changes usually require a nonnegligible system stoppage at least in the area where the new equipment is being integrated.
- (iii) Integrability: the equipment is in principle connected using standard interfaces and/or communication protocols and it is controllable through standard automation languages.

- (iv) Other requirements: the system fulfils the minimum level of requirements for operating according to the production objectives and applicable regulations.

A system at level 1 generally denotes a relatively high level of automation whereby processes and machines are generally connected and influence each other. Communication occurs over standard industrial bus systems, and standard data formats are considered making it possible to replace similar components. Generally, more than one bus communication system is supported which improves the opportunities for integrating new equipment. However, there are different system islands that, although integrated, use information flow and formats that are generally specifically defined in the context of that island and cannot be automatically interpreted or used elsewhere. The main challenge in evolving such system lies in harmonizing the information semantics and ensuring there is a minimal level of equipment semantic interfacing that allows global commands to be issued. These may come to include selectively (de)activating the equipment, selecting specific programs within controllers or machines, influencing and controlling workpiece planning, and scheduling and routing strategies.

Level 2—similar to level 1.

- (i) Adaptability: same as the previous level. However, the system is prepared so that similar modules may be activated or deactivated as part of its operations.
- (ii) Convertibility: the equipment is generally modular and was designed in a way that with a minimal stoppage can be integrated into the production system.
- (iii) Integrability: the equipment is in principle connected using standard interfaces; additional effort has been put in ensuring that modules with the same function and same interfaces can be integrated with a minimal or no reprogramming effort.
- (iv) Other requirements: the system fulfils the minimum level of requirements for operating according to the production objectives and applicable regulations.

A level 2 system in a fairly tuned level 1 system where great care has been put at design phase to enable the quick replacement of equipment, a typical scenario is a production line where different products are produced during different shifts and the line requires the addition and/or removal of equipment, or tools, to support the different production processes (i.e., there is a system preparation process during shift changeovers). The system still features static logical links to the different pieces of equipment which is the main improvement point to reach a level 3 system. Here, the strategy is to improve the logical interfaces rendering them interpretable and discoverable to enable dynamic orchestration of components. The adoption of emerging standards such as OPC-UA and other web service-based formats is a proper step in this direction. Such interfaces will enable a substantial reduction of the preparation time as more processes may be easily designed and

implemented. At the same time, the integration of new machines becomes easier.

Level 3—at this level, the system's structure and behavior are similar to level 2; however, equipment can be discovered dynamically and functions and processes can be activated and designed by orchestration of the different components.

- (i) Adaptability: the production system is programmed to cover all the main cases where decisions may need to be taken; however, new cases can be easily introduced by orchestrating the system's components. The system is reactive to the introduction of new components and can use them as long as there is an orchestration routine including them.
- (ii) Convertibility: the equipment is modular and was designed in a way that with a minimal stoppage can be integrated into the production system.
- (iii) Integrability: the equipment is in principle connected using standard interfaces described in machine-interpretable formats that support the recognition and operation as previously described.
- (iv) Other requirements: the system fulfils the minimum level of requirements for operating according to the production objectives and applicable regulations. Its more dynamic and complex structure may also enable the usage of advanced data collection and processing practices for diagnosability.

A level 3 system is what would be attained if a factory would be created at the present time with the best possible available technology. It assumes the seamless integration and harmonization of equipment functions. This means that within the convertibility limits of the system, many different production processes may be enacted in a practical way. This requires the work of experts for designing these processes and system orchestration routines. This includes both the design for normal operations and recovery actions. These routines are of varying complexity depending on the system's and products' nature and size but once developed, the system is able to follow and react to changes on them seamlessly. Level 3 is particularly aligned with the objectives of the RAMI 4.0 model [41]. However, at this level, system autonomy is still constrained. The system is still executing what it is being told to do. Multilevel autonomous decision-making mechanism is not by default included in the design which focuses mainly in integrability aspects.

Level 4—at this level, the system has the ability to interpret its structure and from it infer and make available possible orchestration routines. The system self-orchestrates within its autonomy boundaries and calls upon human resources to intervene for taking actions beyond its autonomy scope.

- (i) Adaptability: the system has a high degree of autonomy which it uses to dynamically allocate different resources during the production process. Newly introduced equipment is immediately recognized and brought into operation. The system can plan

in function of the available resources and the current production targets.

- (ii) Convertibility: the equipment is modular and was designed in a way it can be integrated without any stoppage.
- (iii) Integrability: the equipment is connected using standard interfaces described in machine-interpretable formats that support the recognition and operation as previously described.
- (iv) Other requirements: the system fulfils the minimum level of requirements for operating according to the production objectives and applicable regulations. Its more dynamic and complex structure enables the usage of advanced data collection and processing practices for diagnosability. In particular, the system has the ability of understanding and inferring from different production contexts which resources should be allocated to carry out the required tasks.

Level 4 systems have been demonstrated, with limitations, in scientific setups and prototypes. A fairly extensive revision can be found in [42]. They draw upon the interface discoverability and interpretability opportunities to take autonomous decisions concerning the workflow of workpieces under dynamic conditions. They require a very good cyberphysical alignment in respect to form and function that usually is not available in today's systems. The generation of self-orchestration routines requires the generalization of functions. Among the many demonstrated systems and proposed architectures, it is usual to consider high-level functions such as transformation, transportation, and coordination. They generally require a one-to-one mapping between the cybercontrolling entity and its physical counterpart with a high degree of embodiment that is generally complex to attain and partly motivates this paper. A complete discussion on these design issues can be found in [16] and the references therein. The steps to level five are a multidisciplinary open research question that will include a substantial increase of the systems' cyberphysical autonomy.

Level 5—this level reflects the ideal CPPS configuration denoting very high cyberphysical autonomy that enables it to even dynamically change its structure.

- (i) Adaptability: the system has the ability to autonomously self-adjust including its physical structure. These changes may come to include acquiring missing functionality of components from a digital marketplace. The system plans and adjusts at different time frames. The scope of self-adjustment includes catering not only for production disturbances but also for faults and failure recovery.
- (ii) Convertibility: components have a high mechanical interoperability which otherwise can be adjusted by the system itself. Components are usually mobile or transferable by other components.

- (iii) Integrability: the system has the ability to recognize any new component or system that is added to it and can generate new functionality by combining the functions provided by the newly added component or system with the ones existing.

- (iv) Other requirements: all to their full extent.

*2.3. Integrated Discussion.* Even if most expectations concerning what a CPPS should be are somehow dispersed in the literature and detailed in different contexts, there is a set of prevailing requirements that clearly suggest the dominating characteristics. These requirements are, themselves, prone to multiple interpretations. This makes it difficult to draw the line where a system goes from a conventionally automated system to a full-featured CPPS. The distinction is harder since most of today's systems already exhibit characteristics that rightfully belong to CPPS.

Current research directions set an important focus on interfacing and system integration aspects. These are very important since, as suggested by the structural models in Figures 1 and 2, they are the pillars of convertibility and integrability. However, a smaller focus has been set on the advanced behavioral aspects.

Previous research in MAS has partially covered some ground in this direction. The functional requirements uncovered as part of the RE/RM exercise in this contribution are reflecting just that. The grading scale previously presented shows that, on the more advanced levels, the key functions are of behavioral nature and although interfacing aspects are fundamental, they do not cater for the adaptability requirements alone.

Most of today's production systems can be positioned between levels -1 to 2 in the scale. Level three is approaching the research domain and reflects the best of what could nowadays be achieved with advanced technologies, for example, OPC-UA, coupled to mainstream AI. This possibility arises from the fact that, generally, web service stacks are relatively easy to integrate with current automation controllers. Such integration will typically not change the controller's native programming languages, in a practice that is well aligned with the traditional automation pyramid [43] also known as the ISA-95 model.

The ISA-95 model has however been recently deemed less adequate for the sound development for CPPSs, and new models such as the RAMI 4.0 (DIN SPEC 91345 [41]) have emerged. The RAMI 4.0 model provides a better coverage on the nature of the communication between the system's components and introduces the idea of an administrative shell acting as a single point of information retrieval. The administrative shell provides a broad coverage for collecting information related to some of the main requirements early specified but generally excludes the behavioral modeling of the components.

The authors believe that level 4 represents the current research front and it is already making important assumptions regarding the nature of the physical part of the system. The redesigning effort for transitioning from levels 3 to 4 is still substantial, and the maturity of the current supporting

technologies is generally low which currently undermines their adoption.

Finally, level 5 takes a look into the future's full-featured CPPSs. Here, the degree of system autonomy and integration is very high and the system complexity is maximal. While level 4 would be attainable with a considerable engineering effort in the direction of applied research, level 5 is still beyond reach without a more multidisciplinary and more fundamental research effort.

There is no doubt, therefore, that the systems of the future will entail a much higher complexity along the dimensions discussed in this paper. Physical complexity arising from improving the system's integrability and convertibility should be quantifiable at large (number of parts, material properties, interface types, etc.).

However, the quantification of behavioral complexity is a more elusive task. Today's production systems are already extremely complex entities combining physical and logical aspects. The fact that very few systems can be brought to an operational status seamlessly is a clear sign that there is already a certain degree of unknown behavior generating unexpected effects. This inherent uncertainty is important to understand the general complexity baseline of production systems. To better substantiate the discussion around increasing degrees of behavioral complexity and associated uncertainty, it is helpful to introduce the idea of "weak emergence" defined as [44] macrostate  $P$  of a system  $S$  with microdynamic  $D$  is weakly emergent if  $P$  can only be derived from  $D$  and  $S$ 's external conditions but only by simulation.

At the light of the previous definition and assuming that today's systems could be set up seamlessly without uncertainty, for a relatively small system and a reduced number of fixed component interactions, it could be feasible to fully formally verify the system behavior. A larger system would require the creation of black boxes, individually verified, and a subsequent verification of the whole based on a selection of inputs and outputs in and from these black boxes. In the current state of developments, this would be believable up until level 3 in the scale discussed before. Until level 3, the causality matrix of the system is extremely complex but to a certain extent still possibly manageable with some simplifications. At level 4, autonomous behavior is introduced. This dramatically complicates the causality matrix because it removes operational constraints from the system decision-making processes. Level 5 is the theoretical limit case.

The richer the adaptive process, the bigger the effort in creating traceable system information and the stronger the requirements on the noncore requirements of CPPSs, namely, diagnosability, predictability, safety, and security. In this context, while the fulfilment of the core requirements is what will ensure the anticipated sophistication in production activities, it is the fulfilment of that most likely will dictate their success. This makes them fundamental research topics that need to be understood less at the light of levels -1 to 2 but more at levels 3-5. Seamless operation of CPPS will only be attained by managing and understanding design complexity as a whole.

### 3. Conclusions

This paper proposes an RE/RM exercise that, drawing from reference literature in supporting concepts for what is now understood as I4.0, provides a synthesized interpretation of the main requirements pending upon CPPS development. Unlike most of the literature definitions, the set of requirements is functionally formulated to better pinpoint the behavioral expectations related to CPPSs. However, even this is prone to multiple interpretations. To partially overcome this challenge, this contribution proposes a scale for classifying CPPSs according to the degree to which they adhere to the different requirements. The paper further discusses the main implications of the different levels and positions of the existing developments.

The authors hope that such contribution will shed a better light on the science of engineering CPPSs by removing partially the conceptual fuzziness and by providing a sound basis for positioning existing systems and understanding the cost of transitioning between different levels.

The discussion is as generalized as possible and targets CPPSs and not their specific instances. The authors believe that this metalevel positioning is very important within the scope of I4.0 since it captures its system of systems nature in a tangible way. It also highlights that the main design challenges require a holistic perspective that promotes and respect the cyberphysical nature of the system components.

### Data Availability

The supporting data was collected from the literature cited within the article.

### Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

### Acknowledgments

This research was financed by Linköpings Universitet.

### References

- [1] J. A. B. d. Oliveira, *Coalition Based Approach for Shop Floor Agility – a Multiagent Approach*, FCT-UNL, 2003.
- [2] B. J. Pine, *Mass Customization: the New Frontier in Business Competition*, Harvard Business Press, 1999.
- [3] W. Shen, L. Wang, and Q. Hao, "Agent-based distributed manufacturing process planning and scheduling: a state-of-the-art survey," *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, vol. 36, no. 4, pp. 563–577, 2006.
- [4] W. Shen and D. H. Norrie, "Agent-based systems for intelligent manufacturing: a state-of-the-art survey," *Knowledge and Information Systems*, vol. 1, no. 2, pp. 129–156, 1999.
- [5] Y. Koren, U. Heisel, F. Jovane et al., "Reconfigurable manufacturing systems," *CIRP Annals*, vol. 48, no. 2, pp. 527–540, 1999.

- [6] H. Van Brussel, J. Wyns, P. Valckenaers, L. Bongaerts, and P. Peeters, "Reference architecture for holonic manufacturing systems: PROSA," *Computers in Industry*, vol. 37, no. 3, pp. 255–274, 1998.
- [7] K. Ueda, "A concept for bionic manufacturing systems based on DNA-type information," in *Human Aspects in Computer Integrated Manufacturing*, pp. 853–863, Elsevier, 1992.
- [8] R. Pfeifer, M. Lungarella, O. Sporns et al. M. Lungarella, F. Iida, J. Bongard et al., "On the information theoretic implications of embodiment – Principles and methods," in *50 Years of Artificial Intelligence: Essays Dedicated to the 50th Anniversary of Artificial Intelligence*, pp. 76–86, Springer, Berlin, Heidelberg, 2007.
- [9] A. W. Colombo, T. Bangemann, S. Karnouskos et al., *Industrial Cloud-Based Cyber-Physical Systems*, The IMC-AESOP Approach, 2014.
- [10] F. Jammes and H. Smit, "Service-oriented paradigms in industrial automation," *IEEE Transactions on Industrial Informatics*, vol. 1, no. 1, pp. 62–70, 2005.
- [11] P. Leitao, S. Karnouskos, L. Ribeiro, J. Lee, T. Strasser, and A. W. Colombo, "Smart agents in industrial cyber-physical systems," *Proceedings of the IEEE*, vol. 104, no. 5, pp. 1086–1101, 2016.
- [12] M. Glinz, "A glossary of requirements engineering terminology," in *Standard Glossary of the Certified Professional for Requirements Engineering (CPRE) Studies and Exam*, IREB - International Requirements Engineering Board, 2011.
- [13] C. Rupp, *Requirements-Engineering und-Management: Professionelle, Iterative Anforderungsanalyse für die Praxis*, Hanser Verlag, 2007.
- [14] L. Ribeiro, S. Karnouskos, P. Leitão, and T. I. Strasser, "A community analysis of the IEEE IES industrial agents technical committee," in *IECON 2017 - 43rd Annual Conference of the IEEE Industrial Electronics Society*, pp. 6139–6144, Beijing, China, 2017.
- [15] L. Ribeiro, "Cyber-physical production systems' design challenges," in *2017 IEEE 26th International Symposium on Industrial Electronics (ISIE)*, pp. 1189–1194, Edinburgh, UK, 2017.
- [16] L. Ribeiro and M. Björkman, "Transitioning from standard automation solutions to cyber-physical production systems: an assessment of critical conceptual and technical challenges," *IEEE Systems Journal*, pp. 1–13, 2017.
- [17] N. Chiriach, K. Hölltä-Otto, D. Lysy, and E. Suk Suh, "Level of modularity and different levels of system granularity," *Journal of Mechanical Design*, vol. 133, no. 10, article 101007, 2011.
- [18] A. M. Farid, "Measures of reconfigurability and Its key characteristics in intelligent manufacturing systems," *Journal of Intelligent Manufacturing*, vol. 28, no. 2, pp. 353–369, 2017.
- [19] A. M. Farid and L. Ribeiro, "An axiomatic design of a multi-agent reconfigurable mechatronic system architecture," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 5, pp. 1142–1155, 2015.
- [20] N. P. Suh, *The Principles of Design (No. 6)*, Oxford University Press on Demand, 1990.
- [21] Y. Koren and M. Shpitalni, "Design of reconfigurable manufacturing systems," *Journal of Manufacturing Systems*, vol. 29, no. 4, pp. 130–141, 2010.
- [22] M. G. Mehrabi, A. G. Ulsoy, and Y. Koren, "Reconfigurable manufacturing systems: key to future manufacturing," *Journal of Intelligent Manufacturing*, vol. 11, no. 4, pp. 403–419, 2000.
- [23] G. D. M. Serugendo, M.-P. Irit, and A. Karageorgos, "Self-organisation and emergence in MAS: an overview," *Informatica*, vol. 30, no. 1, pp. 45–54, 2006.
- [24] M. Broy, M. V. Cengarle, and E. Geisberger, "Cyber-physical systems: imminent challenges," in *Large-Scale Complex IT Systems. Development, Operation and Management. Monterey Workshop 2012*, pp. 1–28, Springer, Berlin, Heidelberg, 2012.
- [25] S. K. Khaitan and J. D. McCalley, "Design techniques and applications of cyberphysical systems: a survey," *IEEE Systems Journal*, vol. 9, no. 2, pp. 350–365, 2015.
- [26] S. Balasubramanian, R. W. Brennan, and D. H. Norrie, "An architecture for metamorphic control of holonic manufacturing systems," *Computers in Industry*, vol. 46, no. 1, pp. 13–31, 2001.
- [27] R. F. Babiceanu and F. F. Chen, "Development and applications of holonic manufacturing systems: a survey," *Journal of Intelligent Manufacturing*, vol. 17, no. 1, pp. 111–131, 2006.
- [28] C. Pach, T. Berger, T. Bonte, and D. Trentesaux, "ORCA-FMS: a dynamic architecture for the optimized and reactive control of flexible manufacturing scheduling," *Computers in Industry*, vol. 65, no. 4, pp. 706–720, 2014.
- [29] A. Giret and V. Botti, "Holons and agents," *Journal of Intelligent Manufacturing*, vol. 15, no. 5, pp. 645–659, 2004.
- [30] P. Leitão, A. W. Colombo, and F. Restivo, "A formal specification approach for holonic control systems: the ADACOR case," *International Journal of Manufacturing Technology and Management*, vol. 8, no. 1/2/3, pp. 37–57, 2006.
- [31] P. Leitão and F. Restivo, "ADACOR: a holonic architecture for agile and adaptive manufacturing control," *Computers in Industry*, vol. 57, no. 2, pp. 121–130, 2006.
- [32] P. Leitão, J. Barbosa, and D. Trentesaux, "Bio-inspired multi-agent systems for reconfigurable manufacturing systems," *Engineering Applications of Artificial Intelligence*, vol. 25, no. 5, pp. 934–944, 2012.
- [33] V. Marik and D. McFarlane, "Industrial adoption of agent-based technologies," *IEEE Intelligent Systems*, vol. 20, no. 1, pp. 27–35, 2005.
- [34] M. R. Endsley, *Designing for Situation Awareness: an Approach to User-Centered Design*, CRC press, 2016.
- [35] M. Riedl, H. Zipper, M. Meier, and C. Diedrich, "Cyber-physical systems alter automation architectures," *Annual Reviews in Control*, vol. 38, no. 1, pp. 123–133, 2014.
- [36] J.-S. Yoon, S.-J. Shin, and S.-H. Suh, "A conceptual framework for the ubiquitous factory," *International Journal of Production Research*, vol. 50, no. 8, pp. 2174–2189, 2012.
- [37] M. Moghaddam and S. Y. Nof, "The collaborative factory of the future," *International Journal of Computer Integrated Manufacturing*, vol. 30, no. 1, pp. 23–43, 2017.
- [38] X. Xu, "From cloud computing to cloud manufacturing," *Robotics and Computer-Integrated Manufacturing*, vol. 28, no. 1, pp. 75–86, 2012.
- [39] E. A. Lee, "Cyber-physical systems-are computing foundations adequate," in *Position Paper for NSF Workshop On Cyber-Physical Systems: Research Motivation, Techniques and Roadmap*, pp. 1–9, Austin, TX, USA, 2006.
- [40] A. Banerjee, K. K. Venkatasubramanian, T. Mukherjee, and S. K. S. Gupta, "Ensuring safety, security, and sustainability of mission-critical cyber-physical systems," *Proceedings of the IEEE*, vol. 100, no. 1, pp. 283–299, 2012.
- [41] D. SPEC, "91345: 2016-04," *Reference Architecture Model Industrie 4.0 (RAMI4.0)*, 2016.

- [42] P. Leitão and S. Karnouskos, *Industrial Agents: Emerging Applications of Software Agents in Industry*, Morgan Kaufmann, 2015.
- [43] A. ISA, “ISA-95.00. 01-2000: enterprise-control system integration part 1: models and terminology,” Technical report, ISA, The Instrumentation, Systems, and Automation Society, 2000.
- [44] M. A. Bedau, “Weak emergence,” *Noûs*, vol. 31, no. s11, pp. 375–399, 1997.

