ISSN 2683-345X
journal homepage: http://ijiemjournal.uns.ac.rs/

International Journal of Industrial Engineering and Management

Volume 14 / No 1 / March 2023 / 63 - 93

Original research article

Softbots 4.0: Supporting Cyber-Physical Social Systems in Smart Production Management

R. J. Rabelo[a,*], S. P. Zambiasi[b], D. Romero[c]

[a] *UFSC - Federal University of Santa Catarina, Florianopolis, Brazil*;
[b] *UNISUL - University of Southern Santa Catarina, Florianopolis, Brazil*;
[c] *Tecnológico de Monterrey, Mexico City, Mexico*

ABSTRACT

This paper presents the results of a research work that aimed at investigating the usage of softbots – as digital intelligent software assistants – as a supporting technology to help in excelling smart production management composed of cyber-physical systems (CPS). This work also attempts to differentiate the many equivalent terms used in the literature to softbots, trying to demonstrate that all of them are just types of softbots. Softbots 4.0 acts as a smart human-machine interface, representing a digital virtual assistant to handle human-related issues when implementing the concept of Operator 4.0. Yet, the concept of Softbot 4.0 is framed against the RAMI 4.0 model in a way to show how it can also be taken as an (intelligent) manufacturing entity. The main goal of Softbots 4.0 is to support the Operator 4.0 in interfacing with smart machines, robots, and systems, aiding people in the automatic, planned or pro-active execution of different, repetitive or complex tasks, efficiently in a more symbiotic human-machine environment. Five different cases have been selected for the assessment, implemented as software prototypes in different production management and shop floor control scenarios close to real CPS. The high potential of the softbots approach could be observed, especially when combined with other enabling technologies for Industry 4.0. A global assessment and reflections on these experiments are discussed at the end.

ARTICLE INFO

Article history:

Received November 24, 2021
Revised March 7, 2023
Accepted March 8, 2023
Published online March 13, 2023

Keywords:
Industry 4.0;
Operator 4.0;
Softbots;
Production Management;
Shop Floor Control;
Social Cyber-Physical Systems;
Smart Factories

*Corresponding author:
Ricardo J. Rabelo
ricardo.rabelo@ufsc.br

1. Introduction

Industry 4.0 can be defined as a production model characterized by the intensive digitalization, use of enabling technologies, and flexible interconnection of smart production systems and value chains to attend customer needs more effectively, applying business models based on the Internet of Things, Services, and People (IoTSP) [1-3]. Manufacturing enterprises have gradually adopted Industry 4.0 to improve their general efficiency and sustainability while coping with the need for highly customized, shorter product lifecycles, and emerging business models [4, 5]. Benefiting from the advances in industrial automation, information and communication technologies (ICTs) as well as from control and management models, shop floor systems and equipment, have also turned into much more active entities within wider, intensively collaborative, and smarter production environments [5].

Production Management refers to planning, coordinating, and controlling the resources required for fabricating specified products by specified methods.

Published by the University of Novi Sad, Faculty of Technical Sciences, Novi Sad, Serbia.

http://doi.org/10.24867/IJIEM-2023-1-325

It handles activities like the selection of products, production processes and the right production capacity, production planning, inventory control, maintenance of machines, production control, and quality and cost control. In essence, this means ensuring that production is running as planned, that corrective actions are taken in the case of deviations, and that products are produced with the expected quality, using the right resources, at the lowest cost, with minimum delays, in a way that the company remains sustainable [6].

Smart Production Management involves handling those aspects helped by integrated sensors, cyber-physical systems (CPS), and computing platforms, including intensive data modelling and predictive engineering for high-quality decision-making [3].

In this *smart* context, one of the strategies being pursued by companies in their *digital transformation path* towards Industry 4.0 is the development of smarter environments, where workers are provided with integrated, easier, quicker, more systematic, and more accurate access to information related to diverse companies' areas, including the shop floor. Their ultimate goal is to support more agile, comprehensive, and confident (data-driven) decision-making, nowadays mostly supported by management and operational dashboards [7-10]. However, this is not simple. The challenges can be illustrated by a study close to more than 150 professionals from world-class large companies [11]:

- 45% of respondents say, *"their organization has not realized the full value of data"*.
- 63% believe that *"inaccurate, outdated or otherwise bad data has been used to fuel business decisions in their organizations"*.
- 64% say that this bad data brought bad repercussions, including *"having to restart a project (57%), missing a business opportunity (53%), wasting time on a project (43%), getting fined due to lack of compliance (37%), losing funding on projects (33%), and reversing business decisions (29%)"*.
- 83% say there is a *"prevalence of 'dark data', i.e., data that is collected but not used to glean insights for decision-making"*.
- (Only) 23% *"explore data for insights at their organizations"* and 39% "*believe only half or less of their team has the skills needed to make use of data"*.
- 34% say they *"spend 16 to 20+ hours per week manually finding, managing, and gleaning insights from data each week. [...] indicates that users do not have the tools they need"*.

It is also highlighted that much information is not digitalized and hence cannot be used for companies' real-time analyses. On the other hand, several existing data that is usually spread over many disparate and isolated repository silos and not properly integrated legacy systems, which are implemented using different technologies, formats, terminologies, and security schema, making the access, understanding, and usage of the right information sometimes challenging for the decision-makers at all company's levels [8, 12]. Besides that, despite the benefits of data-driven environments, they can bring more complexity to managers. The practice has been showing that workers have been increasingly exposed to massive amounts of data about several companies' processes and resources, suppliers, customers, and service providers, leading to many situations of cognitive stress. This, in turn, leads to potentially less comprehensive or even wrong analyses and decision-making due to the usual many checking, analyses, supervision actions, etc., that need to be more often and rapidly performed by shop floor operators and managers [13, 14].

Many Industry 4.0 projects have underestimated the impact on the workers caused by the deep changes in the organizations' processes and by the involved technologies [15, 16]. Several authors (e.g.: [10, 13, 15, 16]) have pointed out the need for a shift in the way workers, systems, and machines interact to enhance operational excellence and human satisfaction towards a cognitive, smart industry. Different concepts have arisen from this need, such as "Social Smart Factory", "Human Cyber-Physical Systems", and "Cyber-Physical Social Systems" [17, 18], which can be generally defined as environments where implicit and explicit knowledge related to activities, preferences, and other human elements are considered in the production system via smart technologies in the virtual, physical, and social worlds for smarter decision-making within smarter working manufacturing environments.

One direction to this emergent type of environment is embraced and represented by the concept of *Operator 4.0* [15]. It relies on a vision where *"smart and skilled operators should perform not only 'co-operative work' with robots, but also 'work aided' by machines as and if needed, by means of human CPS, advanced human-machine interaction technologies and adaptive automation towards 'human-automation symbiosis work systems"*. This concept is one of the pillars that the European Commission has taken in its vision for Industry 5.0 towards placing workers' well-being at the centre of the production process, including providing more symbiotic means of human-machine interactions [19]. However, the path towards its implementation is not clear at all [20], requiring larger research efforts [21, 22].

Several approaches can be adopted to support the implementation of the diverse types of Operator 4.0, as depicted in [20] and [23], being *software robots* (or just *softbots*) a very promising one [24-26]. A *softbot* can be generally defined as a computer program that interacts and acts on behalf of a user or another program [24]. Software robots bridge user needs and bot services to execute functions that are provided by other systems [27].

This kind of software assistance tends to become an integral part of how people live and work [28], helping humans in reasoning and decision-making, physical activities, communicating and interacting, information evaluation, and data processing [12]. Gartner predicts that 50% of knowledge workers will use some kind of virtual assistant daily by 2025, up from 2% in 2019 [29].

Looking at the *Operator 4.0 concept*, softbots can support the implementation of the *Smarter Operator 4.0 subtype* [16]: it can interface with smart machines and robots, computers, databases, and other information systems, aiding the worker(s) in the execution of different tasks in human-like interactions in many industrial scenarios. To this type of softbot, we refer as *"Softbot 4.0"*, which we define as *an intelligent social agent immersed in an industrial computing ambient that is capable of assisting workers in more intelligent management and automation of business processes, computer systems, cyber-physical systems, and production assets, within an Internet of Things, Service, and People environment. Relying on natural language processing technologies and smart text-voice-visual interfaces, it can realize the environment it is immersed in, collect data, understand, and learn from the interests and behaviour of its users; plan and execute tasks autonomously, proactively, or on users' behalf; respond according to users' requests, preferences, expertise level, and business contexts; and warn or prevent users from requesting wrong or unsuitable actions.* In a very simple way, we could say that a *Softbot 4.0* means acting as a workers' partner, helping them, in different manners, to handle their daily tasks, technical problems, and any other work-related topic, more effectively, cleverly, and in a friendly way.

Romero et al. [15] have identified eight categories of possible applications where human operators might be aided by *intelligent software* in Industry 4.0, as *softbots* represent. *Softbots* are also seen as a powerful approach to facilitate the introduction of modern *digital lean manufacturing* and *Jidoka* concepts to help humans in quality control [30, 31].

A *Softbot 4.0* intends to act as what Preece et al. [32] call a third generation of human-machine interfaces (HMIs), which are designed to support users with less mechanical, more intuitive, intelligent, adaptive, and emotional/social levels of interaction. In manufacturing, for example, *Softbots 4.0* can be implemented as a new communication channel with systems and machines, or be implemented as "intelligent interaction wrappers" on top of CPS or legacy systems/machines. The interaction of softbots with humans can be provided by different means, like web browsers and desktop computers, mobile devices, wearables, holography, augmented reality, natural language, haptics, etc. [33]. Korhan et al. [34] argue that such kind of HMI represents a means to implement the emergent concept of *cognitive ergonomics* as modernized shop floors need to help operators in doing tasks that require mental cognition (such as decision-making, planning, situational knowledge, etc.).

Given this potential, the goal of this paper is to show some real examples and assess the use of softbot technology in helping workers better manage production in social CPS. The main motivations for this research work are threefold: (i) the works on the application of softbots (or similar concepts) in Industry 4.0 scenarios are presented as trends and theoretical reflections; (ii) the implementation of softbots has not been devised to support the *Smarter Operator 4.0 concept*; and (iii) the implementation of industrial softbots has been mostly deployed as simple chatbots not directly linked to real CPS.

This paper is organized as follows. Sections 1 and 2 present the research motivations, aims, and methodology for this work. Section 3 provides a review of softbots and equivalent concepts. Section 4 summarizes related works on softbots in manufacturing. Section 5 shows the developed case prototypes and the achieved results. Section 6 presents the conclusions of this research.

2. Basic Research Methodology

Considering the paper's goal, this work has been fundamentally designed to evaluate the use of *softbots technology* in Industry 4.0 scenarios as a means to support the implementation of the *Smarter Operator 4.0 subtype* [15]. Given the wide scope of Industry 4.0, the intended assessment has focused on production management and shop floor control of real manufacturing CPS.

From this perspective, the research methodology was split into three main steps. Firstly, a general *survey* on softbots was carried out having as the essential goal of compiling and clarifying related concepts of softbots. Secondly, a search on the *ScienceDirect, ACM*, and *IEEExplorer* scientific databases was made looking for works on softbots and equivalent terms. However, the objective of this step was not to provide a comprehensive state-of-art literature review on softbots in general. Instead, it was finding out relevant works that have proposed contributions directly related to the use of softbots (or equivalent concepts) to help workers – as Operators 4.0 – in their interactions with manufacturing CPS to assist them at different levels of production management. Finally, five implemented cases were selected, described, and assessed.

Non-scientific materials, such as companies' blogs or their commercial advertisements and web pages (as the digital assistants of SAP's ERP [35] and Oracle DA [36]), were avoided as they do not use to disclose technical information about how their softbots (or equivalent) have been actually implemented, integrated, and further assessed, especially when real CPS is involved.

3. Softbots

3.1. Basic Concepts and Applications of Softbots

There are many definitions in the literature of what a softbot is. To the best of our knowledge, the first formal mention of *softbots* was given by Etzioni et al. [37], which used them as a high-level interface to help users in the search for WWW resources. Another example is provided in [38], as a *"virtual system deployed in a given computing environment that automates and helps humans in the execution of tasks by combining capabilities of conversation-like interactions, system intelligence, autonomy, proactivity, and process automation"*.

The search for less friction in human-system interaction is not a new subject. The first article about *chatbots* – a *conversational software bot* – was published almost 60 years ago, a software that allowed users to interact with computers via natural language [39]. Many years later, and possibly due to the emergence of the Internet and research boom in the Distributed Artificial Intelligence area from the mid-80s, many works started to propose models and implement intelligent softbots as "agents" (e.g.: [40-50]). They combined chatting capabilities with programmed actions to be performed via requests from or on behalf of users to grab useful information from the Internet in different domain areas. Several relevant development environments to implement *chatbots* have been developed since then by large software houses and research projects, such as *Cortana* [51], *Sandy* [52], *Siri* [53], *PAL* [54], *Narval* [55], *Watson* [56], and *Alexa* [57]. Very recently, *ChatGPT* [58] has arisen as a powerful AI-based chatbot that is capable to answer questions from very broad subjects. Areas of application are many, including travelling, health, banking, and government. However, very few works have been directed to production management and shop floor control in Industry 4.0 scenarios [59].

Hermans [40] has identified eight examples of application areas where softbots – as intelligent agents – can be useful:

i. *Systems and Network Management* – Intelligent softbots can be used to enhance systems management software. For example, they can help filter and take automatic actions at a higher level of abstraction; they can be used to detect and react to patterns in the system's behaviours; and they can be used to manage large configurations dynamically.

ii. *Mobile Access/Management* – Intelligent softbots can reside in the network rather than on the user's personal computers, performing user requests persistently despite network problems. In addition, they can process data locally and send only compressed answers to the user rather than overwhelming the network with large amounts of unprocessed data.

iii. *Mail and Messaging* – Intelligent softbots can facilitate these functions by allowing mail handling rules to be specified ahead of time, letting softbots operate on behalf of the user according to those rules and identifying the user's behaviour patterns.

iv. *Information Access and Management* – Intelligent softbots can help users not only with search information and filtering, but also with categorization, prioritization, selective dissemination, annotation, and (collaborative) sharing of information and documents.

v. *Collaboration* – Intelligent softbots can help users build and manage collaborative teams and manage their work products within groupware environments.

vi. *Workflow and Administrative Management* – Intelligent softbots can help in workflow process automation, making them more efficient while reducing costs and human intervention.

vii. *Electronic Commerce* – Intelligent softbots can "go shopping" for a user, taking specifications, and returning with purchase recommendations. Softbots can act as "salespeople" for sellers by providing product or service sales advice and helping troubleshoot customer problems.

viii. *Adaptive User Interfaces* – Intelligent softbots can monitor users' actions, develop models upon their abilities, and automatically help when problems arise. When combined with speech technology, they can enable computer interfaces to become more human or more "social".

Despite all these potentials, many projects have not reached the foreseen expectations. According to a survey on the use of conversational softbots (chatbots) [26] considering 529 respondents from North America and Europe from various companies from different sectors and sizes, the following result has been reached: for almost 50% of the users the chatbot gave useless responses; almost 40% said that the chatbot quite often redirected users to self-serve FAQs given its inability to provide some answer; almost 39% affirmed that chatbots provided bad quality suggestions; 59% stated that chatbots misunderstood users' requests as well as could not handle nuances of human dialogue; and for 30% the chatbot executed inaccurate commands. The survey also pointed out two main reasons for these drawbacks. The first one refers to the lack of deeper involvement of businesspeople in the project formulation. The second one refers to the lack of knowledge from softbot developers as the implementation of interactive interfaces – as required by this type of system – is very different from the development of web or business application interfaces.

3.2. Softbots Properties

Several definitions for softbots are influenced by the application domain they were applied to [27]. As an attempt to provide a generalized view of softbots from the system requirements point of view, some selected works have been analysed [14, 27, 40, 59-69], and the following properties for a "near-complete" softbot could be identified:

- (Some level of) *Knowledge* about the subjects it has to deal with, also considering that this knowledge can be continuously enriched throughout time, both by the softbot's designers and by the own softbot thanks to its autonomy and intelligence.
- (Some degree of) *Autonomy* to reason, evolve, plan, and deliberate about why, what, when, where, and how to do it, whom to communicate with, regarding how much the chosen actions will cost in terms of time, performance, resources, etc.
- (Some level of) *Intelligence* to learn from its interactions (with humans, systems, and sensors), and to evolve according to; to make inferences regarding imprecise or incomplete information present in conversations; and to filter and select correct, trustworthy, and less costly information sources, including trying to avoid peak-hours on the Internet when possible.
- *Coordination* of actions given the different execution paths (contexts) and sequences of (sync and async) function invocations to different systems for different and sometimes simultaneous users.
- (Some degree of) *Flexibility* to act and to interact in different scenarios in different contexts.
- (Some degree of) *Adaptability* to consistently act and interact with different actors regarding their preferences, emotions, and the knowledge it has about them.
- *Sociability* to sense, interact and interoperate seamlessly with different users and systems using proper low-level and high-level communication protocols and semantics.
- *Integrability* with other systems (including other softbots), computing environments and smart devices (such mobile devices, wearables, etc.), and *Interoperability* to guarantee seamless information exchange (control and data) and portability.
- (Some level of) *Security* to protect itself from external malicious attacks and from being accessed by non-authorized actors.
- (Some level of) *Self-management* to evaluate itself against its goals and performance metrics; supervise the execution of its actions; monitor the computing environment it is running in and its execution status; and be resilient to take proper measures to remain alive.
- *Performance* to execute its tasks promptly, correctly, completely, and reliably, regarding its resources and local goals.
- *Usability* in terms of e.g.: deployment, configurability, operability, modifiability, and accessibility.

Lebeuf et al. [27] argue that, however, several domain problems do not require all such softbots' properties or implementation complexities, pointing out that developers should design them and focus on what the intended softbot is going to be for.

When directed to manufacturing, and hence to *Softbot 4.0* concept, these properties should then be linked to production management, shop floor control, and CPS.

Softbots can be seen as just another technology with a very high potential to add value to smart industries. However, it is not able to solve "anything" alone. Real applications of softbots in Industry 4.0 scenarios represent a reasonably complex endeavour, requiring the combination and integration of other technologies, such as IoT and industrial networks, cloud computing, software engineering and integration/interoperability techniques, AI, planning strategies, data storing, HMI technologies, communication protocols, fault tolerance, etc. [69].

Being software, a softbot's architecture, implementation model, knowledge representation model, and internal programming paradigm can vary a lot from case to case. This means that a softbot can be very simple in terms of e.g., interactions and reasoning capabilities, be a small piece of software, monolithic-like and lightweight; or be a very complex and/or large software, services-based and distributed, security-embedded, supporting different and adaptive interaction models, using multiple communication protocols and APIs/other services and libraries, linked to physical equipment or IoT devices, responding to multiple QoS requirements, and AI-based (including complex machine learning algorithms, etc.).

From the implementation point of view, a softbot can be implemented using different patterns, depending on its application and execution environment. For example, it can be developed by adopting a *three-tier* architecture (where a program is physically split into a presentation, a process, and a data layer, especially in the case of being conceived as a web-based application) and its variations, like the MVC model (*Model-View-Controller*, especially in the case of being a mobile App for different types of devices, as smartphones, tablets and wearables); as a classical monolithic software, or as one developed under a service-oriented perspective. In terms of deployment, it can also vary, e.g., from being deployed in a local server, in a local cloud, in a private external cloud, and be accessed as an on-premises application or as SaaS (Software-as-a-Service). Internal system elements can communicate with each other via file transfer, database, remote invocation procedures, messaging, etc. [70].

3.3. Types of Softbots

Several concepts have some profound intersections with what a softbot is, and there is not a clear conceptual frontier established in the literature to differentiate them from each other. Given the broad notion of what a *softbot* can be, Lebeuf et al. [27] consider that such concepts or terminologies used in the literature are, actually, types of softbots, adopted by different authors depending on the type of interaction with users (which can comprise communications with other systems, services, and softbots), the softbot's purpose, its scope of actions, and level of intelligence and autonomy.

Lebeuf et al. [27] proposed a *softbots taxonomy*, which groups those beforementioned softbots properties into three general 'dimensions' (*environment*, *intrinsic*, and *interaction*). Moreover, they see some concepts as evolutions or just as different types of softbots from the intelligence and autonomy levels points of view. For example, they consider (software) 'daemons' as the most basic ones, 'chatbots' as an intermediate level of softbots, and 'agents' as the most advanced ones. However, this proposed organization, besides being very general, seems quite incomplete when looking at several other terms/concepts some authors have been adopting in the last decades.

Other authors have proposed typologies for specific types of softbots. Wellsandt et al. [71], for instance, presented a classification for digital personal assistants: (i) *Adaptive voice assistants* (viz.: speech, optical sensors, screen outputs, execute services upon request, general-purpose, adaptive, computer-generated human-like voice); (ii) *Chatbot assistants* (viz.: text, images, videos, screen interaction, task-oriented support, special purpose, present information to users, virtual characters); (iii) *Embodied virtual assistants* (viz.: human-like, speech, screen outputs, virtual characters, special purpose, adjust to user autonomously, anthropomorphism); (iv) *Passive pervasive assistants* (viz.: unobtrusive, collects data from sensors, initiates interaction with the user, observes user's tasks and context, autonomous, special purpose); and (v) *Natural conversation assistants* (viz.: speech, imitate human natural language interactions, execute services upon request, static behaviour, understands compound commands).

Based on this rationale and the readings done for this work, the descriptions below represent an initial attempt towards comprising and differentiating the main types of softbots and used terminologies presented in the literature trying to clarify to which extent they can be considered as types of software bots. One may see that some types have some overlapping with others or seem synonyms to others. The descriptions are based on some references, although the definition of the same concept is sometimes also blurry in the literature [27]. Our observation is that, after all, the different terminologies seem to just represent the implementation and focus on some of those different softbots' properties, previously mentioned:

- A softbot can be considered as a (software and conversational) *agent* if it implements (at least) the three mandatory agents' properties: (some) knowledge to solve some problem(s), sociability (some interaction with humans), and (some level of) autonomy [72]. This happens regardless of its internal architecture (e.g., BDI [62]) or if it can move itself (mobile agents) through the network to execute its tasks within the digital environment inside the same or different security domain [73].
- A softbot can be considered as an *intelligent system* (sometimes called a Knowledge- or AI-based system) if it can perceive its environment, sensors, and humans, reason and plan accordingly and adaptively, and execute appropriate actions considering its goals, plan requirements, and the environmental conditions in which it will act. This means that even very simple, fast, and straightforward actions or answers to users may be either a result of a deep reasoning process or be the very required action for a severe problem in a critical virtual or physical system humans are managing [72].
- A softbot can be considered as a (*virtual, intelligent,* or *digital*) *personal assistant* if its actions are devoted to coping with the very particular needs of one or a few users with similar goals within some domain areas [71].
- A softbot can be considered as a *chatbot* if it supports chatting (also called a "voice-enabled assistant") and (mostly natural) language processing related to a given subject [68, 74].
- A softbot can be considered as a *bot* or a *daemon* if it were just programmed for doing very specific and somehow repetitive actions automatically but provide some interaction with humans [27].
- A softbot can be considered as an *immobot* if it were specifically designed to monitor itself (e.g., when linked to a CPS) to keep it safe and running [75].
- A softbot can be considered as a *holon* if it were designed to actively represent physical entities of a factory, like parts, pallets, machines, and sensors as intelligent industrial objects [76].
- A softbot can be considered as an *avatar* if it were designed to represent, act, and interact with other users, systems, digital twins, or physical entities of a factory on behalf of the user [77].
- A softbot can be considered as an *intelligent tutor* if it were designed to teach humans adaptively and interactively, according to their current knowledge, current profiles, and answers' level (depth, number of hits, etc.), among other aspects.
- A softbot can be considered as a *devbot* if it were designed to help software developers during the different phases of software development, via reactive or proactive chatting, automation of some tasks on behalf of developers, etc. [78].

3.4. Softbots in Industry 4.0

In the context of Industry 4.0, there are important potential benefits of using softbots when dealing with production management, shop floor control, and industrial CPS. A compilation of this includes [12, 14, 37, 59, 66, 71, 79-86]:

- Automatic and/or autonomous execution of actions, from simple alarms to complex business processes, involving (also on-the-fly) interoperability with diverse systems and communication with diverse computing and hardware devices (e.g., exoskeletons, sensors, and AGVs).
- Higher efficiency and reduction of errors and rework when compared to humans, especially in repetitive, strenuous, hazardous, complex, unsafe, and/or unhealthy activities, are also helpful for unpaired operators.
- Higher availability, able to answer and process many actions simultaneously all the time.
- More user-friendly, intuitive, personal-like, affective, and effective conversation when compared to e.g., FAQs or text manuals on the Web, better supporting higher human-machine/computer symbioses.
- Easier and faster expression of the user's desire, sometimes via very short statements or talks, where the softbot can understand this and decompose that into many low-level computing actions.
- "Standard" answers, are important to guarantee

that both experienced operators and the ones in training will receive the complete enough and consolidated information needed.

- Adaptive and more objective answers regarding operators' experience, emotions, technical profile, business context, and personal goals' status, leveraging higher operators' and managers' experience and work satisfaction.
- More "qualitative time" for humans for more valuable activities, such as management and reasoning, instead of lower-value activities, like checking boring and repetitive tasks.
- Accurate, real-time, up-to-date, and filtered information. Softbots can work 24x7 and can more efficiently have access to the right and authorized sources, also helping in the companies' ICT governance and GDPR compliance.
- Intelligent content, filtering, reasoning, and pre-selecting useful data to be used in different business contexts and problems.
- Automatic gathering of operational data and dashboard generation based on business analytics (description, prediction, and prescription), for further performance evaluation as well as for assisted, interactive operators' decision-making, and conflict resolution (e.g., via automatic negotiation, interactive bargaining, etc.).
- Recommendation of actions within business processes, like e-procurement and supply chain partner replacement, help-desk management, customer relationship analyses, and e-maintenance.
- Collaborative and coordinated actions with other softbots to enhance productivity and conflict resolution (e.g., machines negotiating with each other to see which is the most suitable one to execute a given manufacturing task).
- Learning and self-evolving behaviour, also help in the transformation of operators' tacit knowledge into corporates' formal knowledge.
- Deeper integration of humans into predictive maintenance systems creates opportunities for 'hybrid-intelligence systems', where humans and computers complement each other and evolve together, making humans more deeply involved in decision-making.

Wellsandt et al. [71] grouped these benefits and created a generic categorization for them in the context of digital assistants in terms of which facilities they can provide for their users: (i) users' central access point to systems; (ii) customizations for different users; (iii) tasks delegation (for automation purposes); (iv) holding users' attention on some critical task or object; (v) users guidance when executing some more complex task; (v) hands-free operation (when using voice); (vi) mobile assistance (when users cannot be physically present); (vii) multiple interface types (via different interaction devices); (viii) permanent accessibility (replacing users when they are off for some reasons); and (ix) faster tasks execution.

Conceptually, a *softbot* in manufacturing can be associated with one single system (e.g., one CPS) or it can embrace many systems and communicate with other softbots, so becoming a system of systems (for example, one softbot representing many CPSs of e.g., a manufacturing cell; or groups of softbots representing different manufacturing areas) [14]. *Softbots* can also work collaboratively with other softbots. In the manufacturing domain, a *collaborative softbot* can be defined as *"a software agent that reactively or proactively cooperates with other softbots, CPSs, information systems and humans helping its users to solve complex or unfamiliar problems, and/or to take care of distributed information requests"* [14]. Collaboration is a result of a rational process that a softbot (in this case) can have and that ends up looking for external help [73]. This happens when it realizes it has not had enough or not trustworthy enough information and capacity or capability to accomplish a given task respecting given requirements, or when it has no interest to accomplish the task on its own regarding its plans or execution costs [72]. Therefore, collaboration does not mean forcing interactions between CPSs just to account for that, but rather to support it when needed.

In smart manufacturing, the concept of *immobots* (immobile robots) [75] seems quite applicable to CPS, although not much popular in the literature and not originally conceived for that. Designed to handle the requirements of stationary equipment in NASA space missions, *immobots* are a kind of software specifically designed to take care of equipment conditions and to take measures to keep them working permanently. Strongly relying on the autonomic computing paradigm, their core functions include self-monitoring, self-awareness, self-maintenance, and learning. Their ultimate goal is to support the equipment to respond promptly, adaptively, and intelligently (logically or physically) according to its current internal status and external conditions. The sensing aspect is pretty much based on real-time data gathered by/from sensors linked to the equipment. Sensors, transductors, and actuators may be highly distributed, depending on the equipment's purpose as well as on the instrumentation project that has been designed to get information in a way to guarantee its correct functioning and physical integrity [75, 86]. When considering Industry 4.0, autonomous and intelligent CPS is a key issue to support. Therefore, all those properties of *immobots* are plausible to be implemented in manufacturing CPS, including the possibility to gently interact with human operators, maintenance technicians, etc., i.e., to use softbots in social CPS.

3.5. Softbots within RAMI 4.0 and other Models

Industry 4.0 is predominantly an information-centric paradigm [2, 3]. Several architectures and conceptual platforms have been proposed to develop Industry 4.0 applications, the *Reference Architectural Model Industrie 4.0 (RAMI 4.0)* [87] possibly the most recognized one.

RAMI 4.0 proposes six functional layers in its vertical axis (see Figure 1) for supporting vertical integration and networked production systems. Considering the research made in this work, no reference models (such as RAMI 4.0) do an explicit mention of *softbots* (like ERP or MES systems are considered in the ISA-95 reference model). A possible reason for this is that *softbots* are just another enabling Industry 4.0 technology, likewise, cloud computing and IoT. Trying to place the *softbots* concept in the RAMI 4.0 model, it can be said that softbots correspond to the digital part that allows an intelligent interaction with virtual or physical entities, as a computing system (deployed at any company's level) and a digital twin; or as a physical asset, like as an AGV, a machining centre, or an assembly part (see left side Figure 1). Figure 1 also shows a vision of how *softbots* can be aligned to the RAMI 4.0 reference model.

Regarding the six-layer dimensions of RAMI 4.0, a *softbot* can be seen as a computing system conceptually organized in different modules. It performs a company's business processes using a set of functions. They require and exchange information/data, which are supported by proper communication infrastructures and protocols. Integration grants the interoperable transition from the digital to the physical environment (the given assets), and vice-versa.

RAMI 4.0 was conceived already based on the notion of non or less hierarchical/horizontal communication between industries' people, sectors, and systems, which is one of the fundamentals of Industry 4.0. However, most industries have a brownfield manufacturing environment, composed of dozens of distributed and heterogeneous (legacy) systems, implemented in different technologies with different levels of technological generations and obsolescence. Yet, most of them were developed still inspired by classical industry organization models, quite hierarchical, such as the five-layer reference model ISA-95[1]. Therefore, depending on the existing industry's legacy systems, the softbot's implementation model should rely on them to support its required functionalities. For example, a given softbot does not duplicate functionalities that are already deployed in some legacy systems, like in IoT devices (ISA-95 Layer 0), PLCs (ISA-95 Layer 1), supervisory systems (ISA-95 Layer 2), MES - Manufacturing Execution System (ISA-95 Layer 3), or ERP (ISA-95 Layer 4). Instead, softbots will make use of their systems' functionalities to take care of users' requests or for proactive actions. This means, for instance, and back to the RAMI 4.0 model, that the softbot itself would support but not necessarily internally implement the layers of *communication, integration*, and *information,*

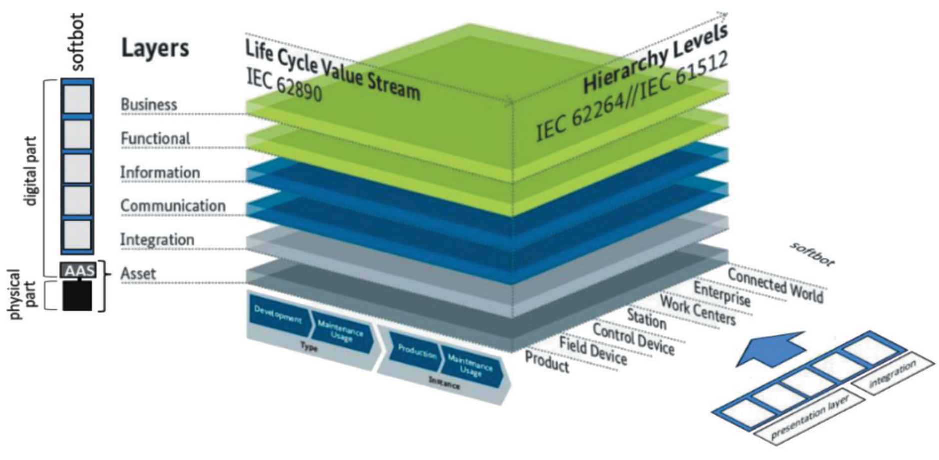


Figure 1. Reference Architectural Model Industrie 4.0 (RAMI 4.0) (Adapted from [87])

[1] https://isa95.com/isa-95-enterprise-control-systems/

as they would be already done internally by those systems. Therefore, the softbot would just have to interoperate with them by invoking their interfaces (API) according to the orchestration and choreography associated with the business process (related to the users' requests or automatic/proactive actions) to be executed by the softbot.

An important element in RAMI 4.0 is the *Asset Administration Shell* (AAS) concept (see Figure 1 left side). It acts as a computing wrapper deployed on top of a physical asset, delimitating its functional responsibility. The *AAS* turns a given asset into a *service provider*, allowing it to be flexibly connected and dynamically bound to any business process within a service-oriented global environment. *Softbots 4.0* can act as an intelligent entity of the AAS (*smart AAS*), providing human interfacing, reasoning, and task execution capabilities to the AAS given that the original AAS model is represented as a services integration/ interoperation wrapper and a resource's description.

Of particular relevance for this research work is the *Integration Layer*, responsible for the provisioning of information about the different networked production assets (viz. product, field device, control device, station, work centres), or even about more abstract entities (as an enterprise and the business networks it is engaged). In this way, information is available for computer and human processing through Machine-to-Machine (M2M) communications (via an *integration* module) and Human-Machine Interfaces (HMIs) (via diverse types of HMI available in the *presentation layer*). This is illustrated on the right side of Figure 1. Hence, it is important to recognize the need for smart HMIs in the emerging IoTSP, given the increasing number of applications that operators must monitor across a smart manufacturing environment [88, 89].

Smart HMIs include built-in integration tools for quick connections to a variety of servers, controllers, and devices, improving operators' productivity when interacting with smart machines and robots, computers, databases, and other information systems in a smart factory [88, 89].

In this paper, *Softbots 4.0* are highlighted as smart HMIs, allowing operators and different networked production assets to interact with each other more intuitively to speed up decision-making and action-taking, and hence to help to excel production management in a social CPS.

A *Softbot 4.0* can also turn a production entity into a smart one. Diverse works propose layered architectures to implement such entities. In general, they divide it into macro layers: physical layer (when this entity is associated with a physical manufacturing element, such as a machine, a component, etc.); sensing layer (responsible for data acquisition and low-level interoperability); communication and integration layer (responsible for data synthesis, and the interaction with other systems and production entities); and cognition layer (responsible to plan, interact with and/or assist humans in analyses and decision-making) [90].

Softbots can also be framed into IIRA (*The Industrial Internet of Things Reference Architecture*)[2]. Among other architectural aspects, IIRA sees an IIoT-based system (in a broad sense) under four so-called viewpoints: business, usage, functional, and implementation, that should be instantiated for given application domains. The business viewpoint attends to the concerns of the identification of stakeholders and their business objectives in establishing an IIoT system (e.g., a manufacturing softbot) in its business context. The usage viewpoint addresses the concerns of the sequences of activities involving interactions with humans or other systems to deliver the intended functionalities. The functional viewpoint focuses on the functional components in an IIoT system, their structure and interrelation, the interfaces and interactions between them, and the relation and interactions of the system with external elements in the environment, to support the usages and activities of the overall system (a manufacturing softbot, in this case). The implementation viewpoint deals with the technologies needed to implement functional components (functional viewpoint), their communication schemes and their lifecycle procedures.

4. Related Works

Few works have been found in the scientific literature applying *softbots* (or equivalent terms, as depicted in Section 3.3) to support workers in their activities of production management and shop floor control close to CPS in Industry 4.0 scenarios. This gap has been also detected in a survey on initiatives for the implementation of the *Operator 4.0 concept* [23].

In terms of general approaches for such implementation, *holonic systems* [76] and *multi-agent systems* [82] seem to be the most used ones. Thanks to the advances in Artificial Intelligence, Machine Learning, and related tools, as well as in other en-

[2] https://www.iiconsortium.org/IIC_PUB_G1_V1.80_2017-01-31.pdf

abling technologies, such as the Industrial Internet of Things, Big Data, Digital Twins, and Cloud Computing, such old approaches have been revisited and have become technologically more feasible as means to implement some Industry 4.0 principles, as decision decentralization, and systems and CPS intelligence and autonomy [86]. Several theoretical works (e.g., [17, 76, 82, 83, 91, 92]) have adopted these approaches, where agents/holons are somehow virtualized to represent different types of industrial entities to: (i) be able to reason about the current shop floor status and to establish conversations with workers; (ii) autonomously provide real-time information to other systems; (iii) dynamically and opportunistically create holon squads to solve problems collaboratively; and (iv) solve problems via e.g., negotiation strategies.

Other examples of works that can be highlighted:

- Schwartz et al. [24] proposed a concept of "hybrid teams" to face the increasing need for higher-level collaboration between humans, industrial equipment, and software. Several potential hypothetical cases were described and the requirements for a generic architecture were presented.
- Although more directed to software development, Erlenhov et al. [93] and Matthies et al. [94] have proposed frameworks where software bots could help users in the management of agile projects, aiding project managers to check developers' performance (but that could be shop floor operators, for example) as well as to proactively check project's milestones and deviations in the planned activities, including informing and supporting teamwork.
- May et al. [95] proposed a taxonomy identifying the many aspects to be considered for implementing worker-centric systems in manufacturing.
- Nazarenko & Camarinha-Matos [96] elicited general requirements for collaboration between CPSs, which included the so-called "human orientation".
- Kar and Haldar [74] identified various scenarios for conversational software bots in IIoT environments, proposing a cloud-based architecture for multi-channel softbots. The goal was to handle communication between humans and IIoT environments, easing device configuration as well as supporting users in higher-level conversations about their status and data.
- Caldarola et al. [97] proposed an architecture for a CPS with one chatbot, focusing on the problem of different semantic interpretations to better understand users' requests.
- Kassner et al. [98] proposed a general architecture for what they called a "social factory", implementing a software bot to interact with one single machine. The goal was to illustrate the potential benefits of this in a smart factory environment. In this line, Singh & Tretten [90] devised an architecture for what they called a "Human-Cyber-Physical System", identifying the different roles associated with humans, the machines, and the machines' cyber parts.
- Dersingh et al. [99] developed a chatbot to monitor and record issues of a production line and to notify corresponding workers for appropriate actions.
- Longo et al. [100] implemented a framework to support the interaction of humans with physical equipment and their digital twins in a cyber-physical environment. The novelty of this work mainly relies on making users interact with a softbot that represents a virtual entity (i.e., a digital twin). Longo et al. [69] have further developed a voice-enabled assistant (using smartphones) to help the "Operator 4.0" in asking for information from a single CPS.
- Chen et al. [101] developed an engine that captures the production plan and transforms and adapts it to the skills and experience of the involved users to improve factory effectiveness and human satisfaction.
- Gnewuch et al. [102] made some experiments to evaluate the effects of how pre-designed delays in the conversation between users and systems could positively shape users' perceptions during conversations with software bots. They realized that delays in some situations can create a more human-like environment, and hence a better symbiosis between humans and chatbots.
- Zajec et al. [103] have developed an assistant with machine learning capabilities applied to help humans in inventory forecasting decisions. After the algorithm was trained for a long time, decision-makers are assisted in their analyses and final decisions towards turning more accurate the activities of production planning and order fulfilment. To an equivalent extent, Bousdekis et al. [104] developed an assistant to address the end-of-line quality control to adopt a predictive quality strategy that links the quality control of the finished product with the design stage and the shop floor.

- COALA project [105] is a European Commission-funded project under which a human-centered Digital Intelligent Assistant is being developed for education and training purposes with the aim of better supporting operative situations characterized by manufacturing workers' cognitive load, time pressure, and little or zero tolerance for quality issues. Several papers from this project have been published by its partners.

Despite the relevance of the outcomes and theoretical artefacts provided by all these works, to the best of our knowledge – except the work of Longo [69] – we could not find examples of implementations of using softbots (or equivalent) effectively integrated into real physical CPS and that, based on the data got, could establish conversations with the so-called *Smarter Operator 4.0* to help him in his activities of production planning and shop floor control.

5. Case Studies

This section intends to present implementations of softbots to support the *Smarter Operator 4.0* close to CPS.

Given that identified gap in terms of implementation initiatives; that this paper's authors have implemented five different case scenarios of this; that Longo's [69] work is equivalent to one of these cases (although not many details were given in his paper); and that the essential goal of this paper was not to compare the authors' implementation with others, but rather to demonstrate the potentials of softbots technology when assisting the operator 4.0; this section will be focused on presenting a compilation and summarized description of those five cases to further present a global analysis. The detailed explanation of each case (e.g., in terms of softbot derivation, messages modelling, systems integration, etc.) can be seen by accessing the respective publications, whose references are mentioned when the description of each case.

The five softbot cases represent examples of possibilities of many scenarios where *Softbots 4.0* can be used in diverse usual situations in an industry in terms of production management and shop floor control, trying to cover the three softbots' modes: acting upon request, doing activities automatically, and do activities pro-actively.

They were mostly developed as proof-of-concept software prototypes. Although the implemented softbots were deployed in real CPS, this was almost all done in a controlled environment, using the university laboratory and (previously trained) engineering students to act as 'managers' or 'operators'. This implied that the integration of the softbots with other systems (like the ERP and MES) was very simplified, using the simple ones existing in the lab. Yet, the production and shop floor control scenarios were also simplified as the goal of each case was not to show the execution of extensive business processes that might involve plenty of situations and variations of human interactions for each case.

Softbots 4.0 involves not only the connection with people but with industrial machines as well. This means communicating and acting directly on physical and sometimes critical equipment and infrastructures, hence becoming a potentially big security problem in the case the softbot is somehow hacked. Despite this, and considering the essential goal of this paper, this important and complex issue was not supported at all in the implemented prototypes, which only provided the basic security resources offered by the adopted implementation tools and technologies. Many works in the literature propose techniques to support cybersecurity in critical systems, as in [106], as well as some ones that depict the most relevant types of cybersecurity problems in chatbots, as in [107].

The implementations used the *ARISA NEST* platform, an academic *PaaS (Platform-as-a-Service) environment* developed to derive and execute web-based, scalable, open, and service-oriented softbots for different application domains [67, 108].

The *ARISA NEST* has a *Web Interface* that allows *Designers* to create and manage the life cycle of multiple bots for multiple users/companies. The derived bot has a *Knowledge Base*, which is composed of contexts and dialogues, beliefs, scripts, and behavioural algorithms. Once derived, *Users* can interact with the bot via chat (e.g., telegram, webchat, mobile app, etc.) using natural language, by typing or voicing. Depending on the user's request or preconfigured action to be executed, this is internally handled either by the behaviour executor or by directly invoking web services to trigger the required actions, helped by an internal orchestrator engine. All the conversations between users and the softbot are modelled in *contexts*. A *Context* is a set of dialogues with a common subject. A bot's conversation domain can have many contexts, viewed as a tree, having a *root* as the starting point, and internally organized as a graph.

ARISA NEST supports three types of behaviour modes in its communication with end-users: (i) *reactive* – when the softbot acts in response to direct users' requests via chatting (e.g., to ask about more detailed information from a given machine); (ii) *planned* – when the softbot acts in response to predefined scheduled tasks (of different types and complexities), bringing their results to users after their execution (e.g., to generate consolidated performance reports weekly); and (iii) *pro-active* – when the softbot performs predefined tasks autonomously on behalf of users or of some system it represents, bringing their results to users if needed (e.g., to continuously checking communication problems between data collectors and MES system and to promptly take measures to solve them, or sending warnings and alarms).

It has not been found in the literature a formal and comprehensive reference framework to evaluate softbots, which, in theory, should provide resources to evaluate all their properties (see Section 3.2) and at several dimensions. Some works have proposed some *ad-hoc* perspectives of analyses, more related to what they wanted to evaluate. For example, in [109], authors used the trustworthiness of the system, the usability of the digital assistant, the cognitive workload of its users, and the overall business benefits for the corporation to evaluate voice-enabled digital assistants. In [85], the authors proposed evaluating the digital assistant based on its general execution efficiency, the quality of the predictions, the transferability of dialogues to easy analytics, and the effects of the provided analytics in the business.

In the case of the work presented in this paper, regarding its essential goal of demonstrating the potential of *Softbots 4.0* approach to help the "Operator 4.0" in its daily activities of production management and shop floor control close to industrial CPS, the indicators were not much formal. A more qualitative approach was applied, focusing on how much the softbot could assist people in executing tasks on demand, automatically and proactively, using different levels of fluid interaction and intelligence. In most cases, students and professors were used for evaluating the results; only in two cases real operators in real companies could be considered.

5.1. Operator 4.0 + Softbot Close to One Machine

This first case refers to a *single softbot* that helps machine operators in some tasks via a high-level interaction. In very general terms, this Case can be considered equivalent to the one implemented by Longo et al. [69]. Three scenarios were supported: (i) the *softbot* monitors a production process and keeps the *Operator 4.0* informed about its execution; (ii) the *Operator 4.0* interacts with the *softbot* via text or voice in the mobile phone asking about production issues; and (iii) the *softbot* publishes a report with a summary of its daily activities.

This case corresponds to a reimplementation of previous work [10]. It was implemented in the *Festo MPS didactic* plant existing in our research lab (partially shown in Figure 2). This plant is composed of six machines (stations) that work in sequence, as follows. A *distribution station* receives parts from a buffer according to the current production plan. Some initial tests are made on them. They are further received by the *testing station*, which checks several aspects to guarantee assembly conformity. In the case the part is not OK, it is put out of the production line; otherwise, it is separated/sorted according to its size, weight, colour, and material by the *separating station*. The *pick-&-place station* starts the first phase of assembling and picking parts according to the orders' due date indicated in the production plan. The *muscle press station* completes the assembling process by joining different parts to compose the final product. Finally, the *sorting station* takes the different final products and sorts them according to their types for further packing and delivery.

Each station is equipped with a PLC *Siemens S7-1200 full DC* and a set of sensors and actuators, hav-

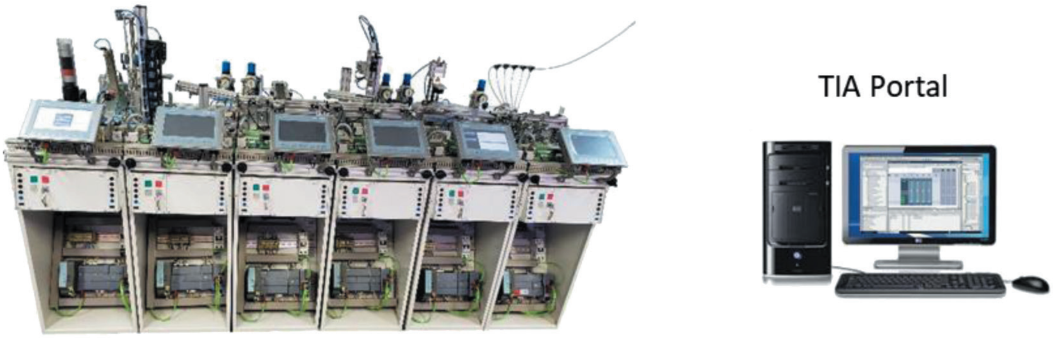


Figure 2. Festo MPS Didactic Plant

ing a link to the outside via a *Profinet* network and *OPC* protocol. It is possible to access their data by reading the many *tags* they instantiate during the assembly. This is done via the *TIA (Totally Integrated Automation) Portal*, from Siemens too, which acts as a general management environment, providing several functionalities, like CLP programming and plant supervision (SCADA). Client applications (e.g., a softbot) should communicate with the TIA Portal (and so with the plant) via OPC.

Scenario 1

In the first scenario, the business process (BP) associated with the case was designed to monitor the number of parts that are initially available and the ones necessary to start the production considering the demand expressed in the production plan. In the tested scenario, there must be a minimum of 40 parts of type '10001' in the buffer to take care of the order id '128'. There is a sensor in the distribution station that counts this, which is received by the softbot via OPC. In this case, it was detected that 27 more parts are still required (see Figure 3). Given that, a request for a quotation should be immediately and automatically triggered and executed by the softbot close to that part's supplier (called '*UFSC Ltda*'), previously registered in the e-procurement system. This transaction would cost $55.00 and has '800' as its id quotation.

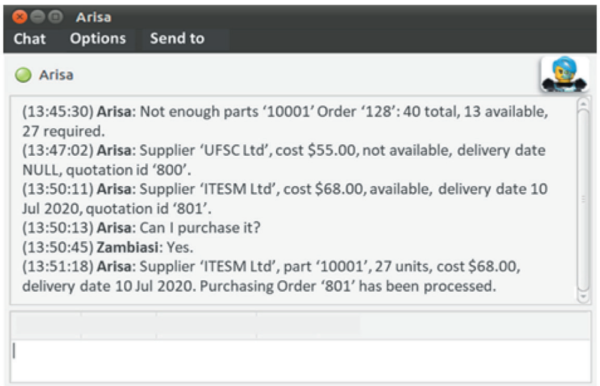


Figure 3. Softbot Main Dialog Interface [10]

However, this supplier currently does not have the requested number of parts. The softbot then decides to look for another supplier, choosing the second in the list, '*ITESM Ltd.*'. It sends another request for quotation (it uses an API to call the [hypothetical] ITESM's ERP function responsible for that), receiving the price ($68.00) and the expected delivery date (10 Jul 2020) as a result. Following the company's governance model, which should also be reflected in the softbot's autonomy model, Mr. Zambiasi should authorize this transaction. Once authorized, the softbot does the purchase and notifies Mr. Zambiasi about it, informing him that 27 parts have been purchased, via order '801'. The generated conversation between the operator and the softbot has happened in the softbot's desktop interface environment, using the Google *Gtalk* tool, as shown in Figure 3.

Scenario 2

In the second scenario, the operator, via his mobile phone, asks the softbot if there is some pending order regarding the production plan in place. The softbot answers that there are two orders of id '102' and '226', and their respective planned starting date. Later, the operator wants to check if Order '226' has been finished, and the softbot answers that it is delayed and shows the new planned starting date. This has been implemented using the *Twitter* tool, and it is shown in Figure 4.

Figure 4. Operator 4.0 Mobile Phone

Scenario 3

In the third scenario, a summarized report is generated by the softbot and sent out to the operator (and/or managers) at the end of the day, for final checking and further storing for future auditing purposes. The softbot publishes it in one of its user interfaces and has used an open-source *blog* tool. This is shown in Figure 5.

The softbot could perform the actions as planned and could assist the human operator in some tasks, both in terms of interacting with him when needed and by automating task execution. These were the general indicators used to evaluate this prototype [14].

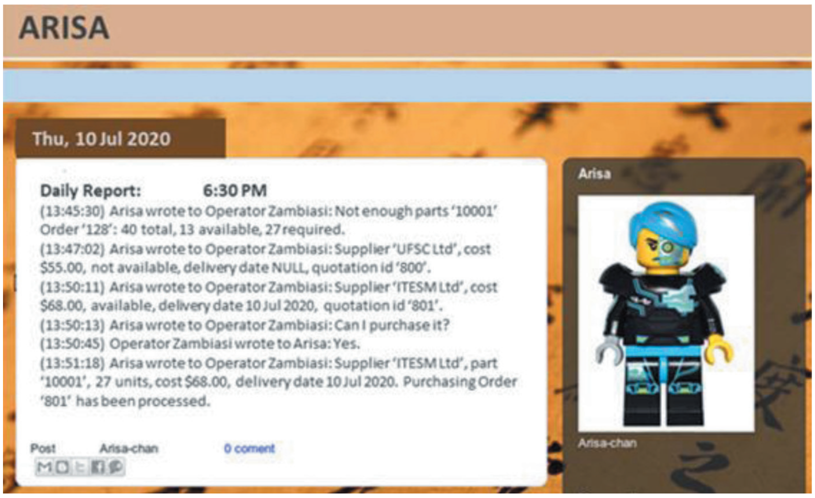


Figure 5. Softbot Daily Report [10]

5.2. Operator 4.0 + Softbot Close to Several Machines

This second case refers to *collaborative softbots* interacting with the operator or with other softbots. Five scenarios were implemented: (i) the *Operator 4.0* asking the *softbot* about current deviations in the production plan considering the number of final assembled products; (ii) the *softbot* helping the *Operator 4.0* in checking possibilities of rescheduling due to delays in the production plan; (iii) two *softbots*, pick-&-place softbot and distribution softbot, interact with each other to see how to overcome current stock issues after a new production schedule; (iv) *softbots*, acting as a *digital andon system*, warn operators via an SMS message about production problems; and (v) *softbots* proactively do a production follow-up to determine if the production plan is being followed as scheduled.

This case has been detailed and described in previous work (i.e., [14]) and was developed based on the before-mentioned *Festo MPS didactic plant*. Nevertheless, there are three basic differences to the previous case: each station is no longer a "simple" machine, but it is wrapped with a 'management & communication' layer transforming it into a *Cyber-Physical System (CPS)*; there is one softbot per CPS, creating a team of independent but *collaborative softbots*; and there is more than one operator – Operator #1 and Operator #2 – who can also interact to each other, simulating the *One Worker Multiple Machines* (OWMM) philosophy.

Scenario 1

The Operator #1 asks, via voice recognition, the *separating station* CPS's softbot called *'Roy'* (which is a name similar to a human friend instead of a formal name for a computer system) about current deviations in the production plan according to the production schedule considering the number of final assembled products – add up by a counting sensor in the *sorting station* – due to some problems occurred during the work shift (see Figure 6a). *'Deviation'* word and *'today'* are the keywords the softbot is prepared to hear, which in turn is handled by one of the CPS's manager services (see Figure 6b) responsible for calculating the current production deviations based on the production schedule. The *softbot 'Roy'* understands that *'today'* means verifying what is the day *'today'* production schedule and takes this date as the target to identify the production deviations. The *separating station* CPS's softbot reacts to this voice request, accesses the MES's database directly (without the need to broadcast any messages to other CPSs to try to get this information as it is stored in the MES or ERP system) and sends the information back to the Operator #1, notifying him about the production deviations in terms of expected vs. actual assembled products amount. Figure 6c shows an excerpt of the coded script to perform this querying action to the MES system about production plan deviations [14].

Scenario 2

Given the delays in the production plan (as fewer products have been assembled according to the production schedule), the Operator #1 asks, by typing in the keyboard of his/her smartphone, to the *pick-&-place station's* softbot if it is possible to 'anticipate' the production of *Bottle A* without delaying *Bottle B*'s due date (see Figure 11a). Some typed words are taken by the softbot as keywords besides the fact that the Operator #1 knows that such products' names are valid in the ERP database system.

Figure 6. Softbot Querying the MES System about Production Plan Deviations [14]

The softbot *reacts* to this text request by executing the proper CPS manager service, responsible for interacting with the ERP's scheduler module via its API (see Figures 7b and 7c). The scheduler can return two parameters after some calculations to the Operator #1: 'NO', meaning this is not possible according to the current production plan schedule; or 'YES' and *'date&time'* value, meaning it is possible, and that the activity linked to 'Bottle A' at the *pick-&-place station* should start on the given date and time to reschedule the production of 'Bottle A' without affecting the production of 'Bottle B'. With this information, 'YES' or 'NO', at hand (see Figures 7d and 7e), the Operator #1 evaluates the situation and, given the autonomy philosophy in Industry 4.0 (i.e., human-in-the-loop), (s)he takes the final decision. If (s)he agrees on 'YES', then the softbot invokes again the ERP's scheduler module to update the production plan, which in turn updates the dispatcher's plan. The dispatcher updates the *pick-&-place's* PLC program so that the new production sequence can be performed. The operator also can access the ERP database, stored in a cloud, to have a broader vision of the production via e.g., Gantt charts and performance indicators dashboards [14].

Scenario 3

During the execution of the new production schedule (see Scenario 2), the *pick-&-place station* softbot asks the *distribution station* softbot if there are enough *bottle caps* in stock to accomplish the new production plan for *'Bottle A'* according to its new schedule (see Figure 8a). The distribution station's softbot first asks Operator #2 (*'Rick'*) about the available inventory of bottle caps via voice (or text). He answers, *'I do not know'*, and then the softbot proactively accesses the inventory information stored in the ERP system database (see Figure 8b). The distribution station softbot sends the number of bottle caps in stock ('50 bottle caps') to the pick-&-place station softbot, which verifies that this is not enough stock and notifies the Operator #2 (see Figure 8c) about it. In parallel, the softbot sends an e-mail to the purchasing department to warn it about that too, always keeping "humans-in-the-loop". This proactive functionality demonstrates that interactions in an Industry 4.0 environment can be non-hierarchical, crossing many different company departments [14].

Scenario 4

The *testing station's* softbot permanently monitors the execution status of this station in the MES database following a *detect-&-repair* policy for supporting a total productive maintenance strategy. This softbot seeks to ensure that the testing station is always available for use according to the production schedule.

In the case of detecting that the station is currently stopped due to whatever problem, for example, the bottle's cap has been placed upside down as detected by a digital *poka-yoke sensor*. The softbot sends a message, via the OPC, to the *testing station's* PLC to go into alarm as well as sends an SMS message to the Operators #1 and #2's smartphones, acting as part of a *digital andon system* (see Figure 9a). Operator #1 was nearer to the station, confirms visually that the problem exists, and asks, by typing in the *testing station's* softbot interface, 'How to solve the problem?' (see Figure 9b). The softbot accesses the company's intranet and shows him/her the exact part of the machine troubleshooting manual that explains the procedure(s) to solve the problem detected (see Figure 9c) [14].

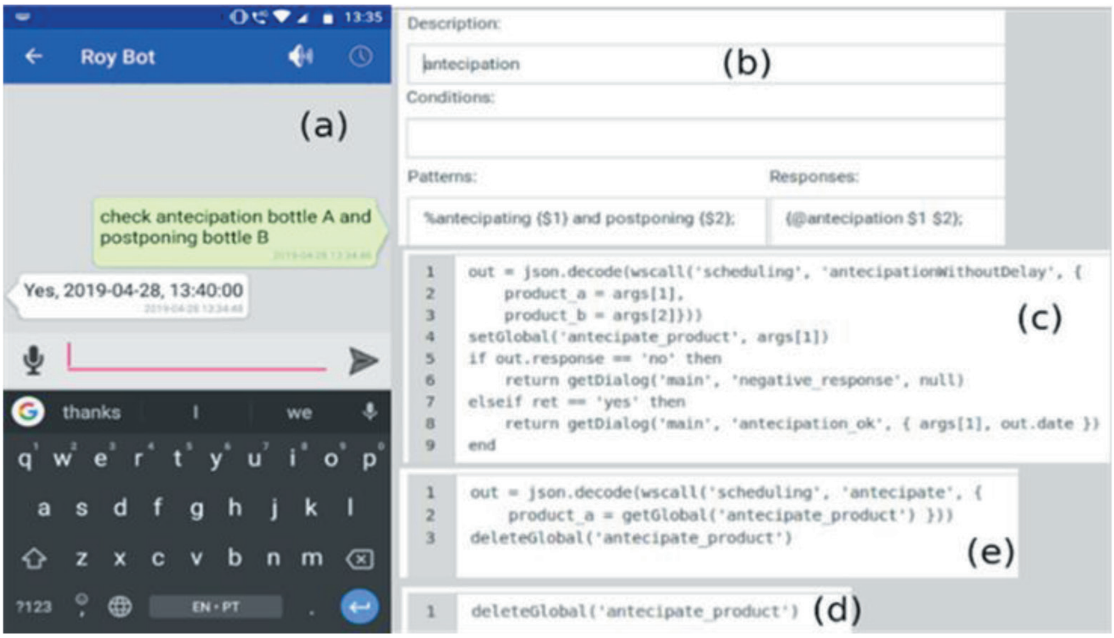


Figure 7. Softbot Querying the ERP System about the Possibility of a Production Rescheduling [14]

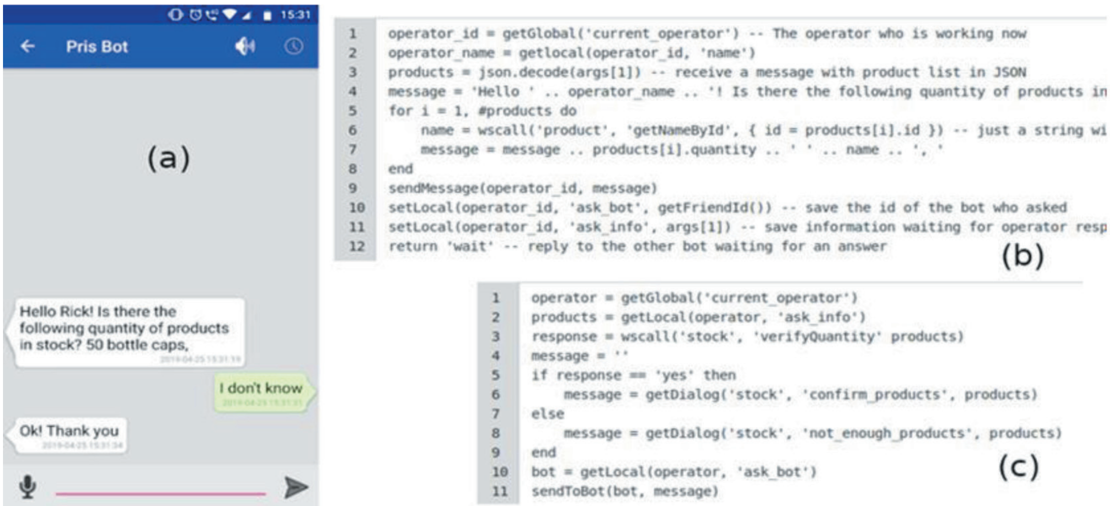


Figure 8. Softbot Querying the ERP System about Raw Materials Inventory [14]

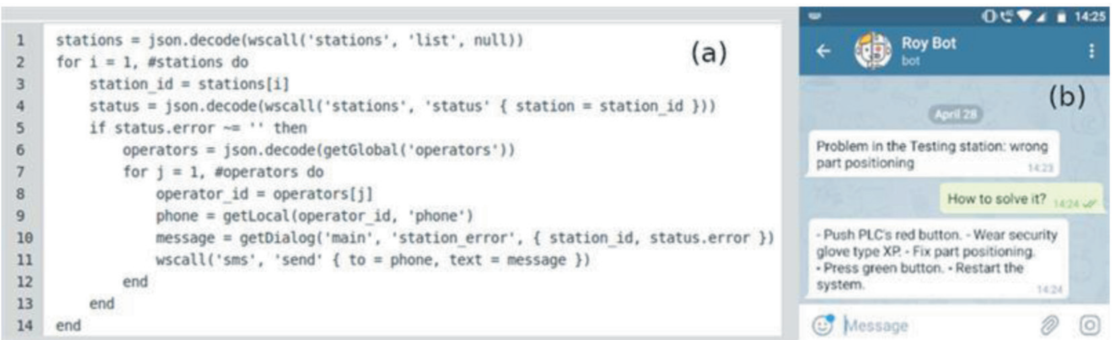


Figure 9. Softbot Querying the Station's Troubleshooting Manual [14]

Scenario 5

As part of their routines, all *softbots proactively* show a real-time log on the Festo central computer with the list of the orders in place, their due date, and if they are on time or delayed supporting production control at the shop floor. This information is obtained from the MES database, which has all data in English. However, the softbot realizes that the operator present in the work shift is Brazilian, so using its interpreter capabilities (i.e., language translation), it automatically translates the data to Portuguese to turn the communication more pleasant to the operator and also reduce potential human error due to some interpretation problem. For example: *"No horário"* means 'in time'; *"atrasado"* means 'delayed'; and *"planejado"* means 'planned' (see Figure 10) [14].

This softbot feature as an 'interpreter' is quite relevant to support an international workforce, not only in production planning and control activities but also in problem-solving ones when, for example, the troubleshooting manual for a machine is written in a different language (see Scenario 4).

Scenarios Summary

The softbots could execute the actions as planned. They have assisted operators in some tasks, they have interacted with them, they have performed some tasks automatically, and they have interacted with other softbots when necessary.

Some undergraduate engineering students (as 'operators') were trained in advance to use the softbots and asked about it after the experiments. Based on their answers, they believe that softbots can be more effective in doing many and sometimes difficult tasks automatically as well as creating a more user-friendly interaction environment when they wanted to get some production information. This includes access to information they barely would know how to get (or not so easily or promptly) from the MES system and other CPS.

5.3. Operator 4.0 + Softbot Embracing an Entire Shop Floor for OEE analyses and Maturity level

This case aimed to present an approach to how softbots can help managers in their daily management of production. It corresponds to a partial reimplementation of previous work [59], in cooperation with *Harbor*[3], one of the leading software providers of MES systems in Brazil. This case/example corresponds to a real scenario, based on real data.

The main motivation of this case relied on the fact that most SMEs are very limited to handle the so much information got from their shop floors and exposed in their MES' production management interfaces, to reason about, and respond appropriately to solve the problems. One of the main underlying problems is that these actions should be carried out during the working shift, which creates a very stressful ambient and this frequently leads to incorrect actions. Another goal refers to dealing with the usual lack of good management background from some SME managers, and with the fact that most of them are not used to running their businesses on a data-driven basis.

A softbot was created to work as a module of a cloud-based MES system (called *LiveMES*), which works on the real-time data got from the shop floor of Harbor's clients to help production managers in some of their daily activities. The softbot is called *Livia* (*LiveMES* + intelligent analysis). To this approach, we called *Production Management as-a-Service (PMaaS)*.

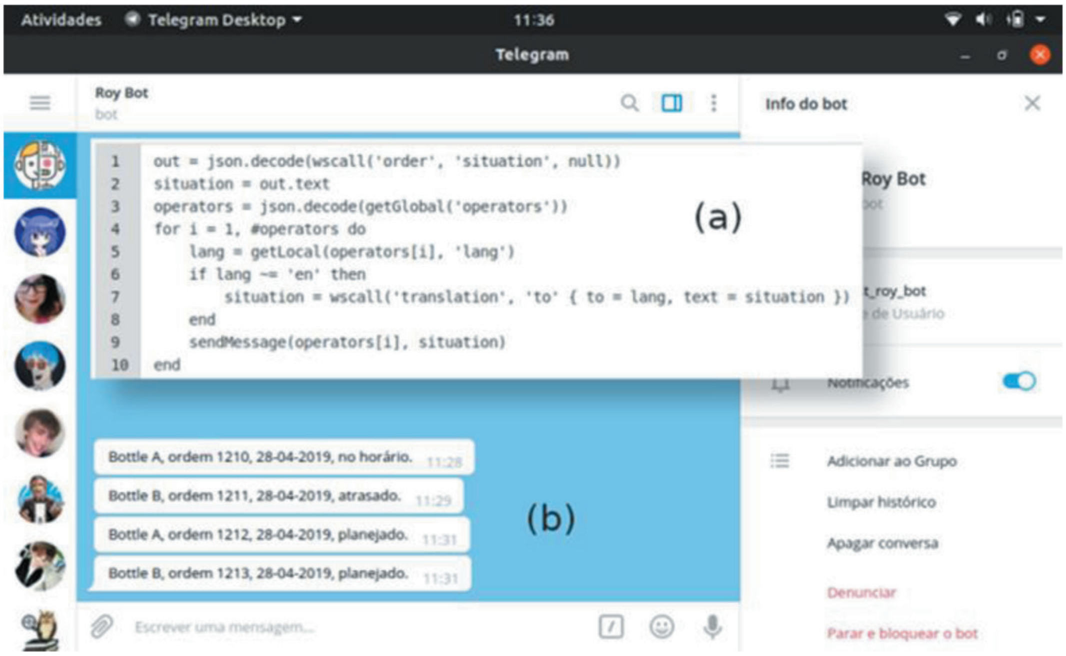


Figure 10. Softbot Translates to a Different Language the Data [14]

[3] https://www.harbor.com.br/

Besides being prepared to answer usual questions about the production, machines, etc., *Livia*'s essential goals are to assist managers in: (i) being aware of how trustworthy the data grabbed is from the shop floor (for more confident and accurate decision-making); and (ii) their analyses upon such data and the final decision making.

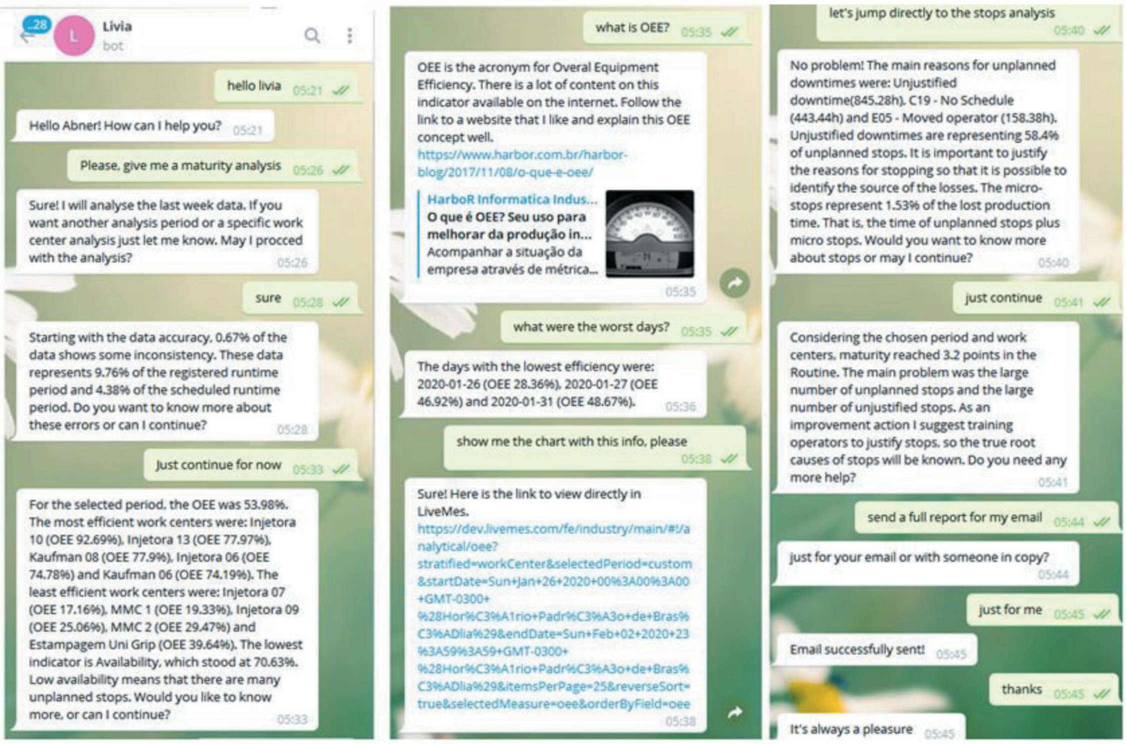
Maturity models were the core theoretical foundation used to approach the first goal. *Maturity* is a measurement of the ability of an organization to continuously improve some of its capabilities. Maturity is typically expressed in levels. The higher the maturity the better the company [110].

Four maturity levels were defined: Level 1, *Resources*, assesses if the shop floor's supporting instrumentations are properly running and measuring the expected information. Level 2, *Rigor*, assesses if the set of expected assets and production entities are properly registered and communicating with the MES system. Level 3, *Routine*, assesses if the set of predefined management and supervision actions and processes have been executed. The highest level, 4, *Run*, assesses if a high-level set of production data is being used to manage the production. The calculated assessment level is displayed in a *Radar*-like interface. This creates the so-called *'RA-RE-RI-RO-RU'* measurement cycle of the company's maturity evolution [59].

Business Analytics was used to approach the second goal. It refers to methods and techniques used to measure an organization's performance by exploring its data to gain insight and drive business [111]. Making use of those three softbots' behaviours and via interacting with *Livia*, four types of business analytics are provided: *description, diagnostic, prediction*, and *prescription*. They can be triggered either sequentially or independently from each other, depending on the situation in place. For example, when a given problem happens, it is identified (*description*) and its cause(s) named (*diagnostics*). Possibilities to solve it can be generated and evaluated (*prediction*), and straightforward measure(s) are suggested (*prescription*).

A broader view of the main steps related to one scenario of *Livia* is presented below[4]. In the current implementation, *Livia* is always ready to attend to users' requests (in this case, the manager called *Mr. Abner*). The system has been designed to leave the interaction with users the most colloquial as possible. *Abner* wants to know about the company's maturity level and *Livia* asks him about the desired timeframe or focus of analysis (see Figure 11a).

In this example, *Abner* wants to focus his analysis on the *Routine* level to check data integrity and consistency. This is important to evaluate because: sometimes operators forget to register some informa-



(a) (b) (c)

Figure 11. Chatting between Livia and the Operator/Manager

[4] The texts in the interfaces were translated *ipsis litteris* from Portuguese to English for the purpose of this paper.

tion or add them incorrectly; the grabbed data is an outlier, or the monitored production rates of some work centres are higher than their standard rates. *Livia* does that automatically, computing bunches of data in a fast way, and informing *Abner* about it right after that.

Figure 9a also shows the *descriptive* analytics part. It was observed that, in general, only 0.67% of the collected data has certain inconsistencies. However, focusing on more specific and relevant information, *Livia* observed an “inconsistency” between the actual and expected work centres’ runtime state (9.76% and 4.38%). On the other hand, since the degree of discrepancy is lower than 10% (previously configured), *Abner* prefers to go on, now aiming at performing the *diagnostics* analytics. *Livia* shows the calculated machines’ OEE (*Overall Equipment Effectiveness*) to *Abner*, complemented with reference metrics (see Figure 11a). Livia realizes that the main cause for the low mean OEE (70.63%) was several unforeseen interruptions in the machines.

Abner can request more information about any aspect he wants. In the example, considering he is a very experienced manager, he prefers to check what the system means by ‘OEE’ before continuing his analysis (see Figure 11b). An initial short answer is given, with the possibility to have a more comprehensive description of the OEE, which is provided via a trustworthy Internet URL.

Having the OEE data and now sure about the OEE meaning, *Abner* asks about the worst OEE days. In theory, *Livia* could have an internal service functionality implemented to handle such requests. However, as this is provided by the *LiveMES* system, *Livia* just indicates the URL *Abner* can use to get that. To have a broader view of the machines’ OEE, Abner wishes to see that in a graphical chart way (see Figure 11b). The current ARISA implementation does not support internal graphical representations. An Internet link is then generated to have access to the created file with the desired chart, and it can be visualized (see Figure 12).

The next step refers to *predictive analytics*. For example, *Abner* could be interested in checking, via *Livia*, the OEE prediction for the next working shift before deciding what to do at this moment to solve the current problem of low OEE. This feature, however, is still being implemented, in the form of an (internal) invocation to an AI-based prediction tool’s API to calculate that.

In terms of *prescriptive analytics*, the last step. Considering the calculated company’s maturity for the Routine level (3.2) and what *Livia* has mapped in its knowledge base (which is based on good practices), it does not have a solution to immediately solve the OEE problem. *Livia* then suggests a short/medium-term measure to handle numerous stops in the work centres, which is reinforcing training activities close to the operators. Finally, *Abner* requests an executive production report to be sent out to his e-mail (see Figure 11c).

As the last step of the global *RA-RE-RI-RO-RU* maturity cycle of the given company, a Radar-like chart is generated (see Figure 13), providing a broader analysis perspective to managers.

Besides the more user-friendly, reliable, integrated, and less stressful ambient for better, agile decisions way of managing the production, *Livia* softbot has brought some important benefits. On average, a rough weekly manual analysis (by managers) of the most important work centres (normally four) of each company took 2.5 hours. After this implementation,

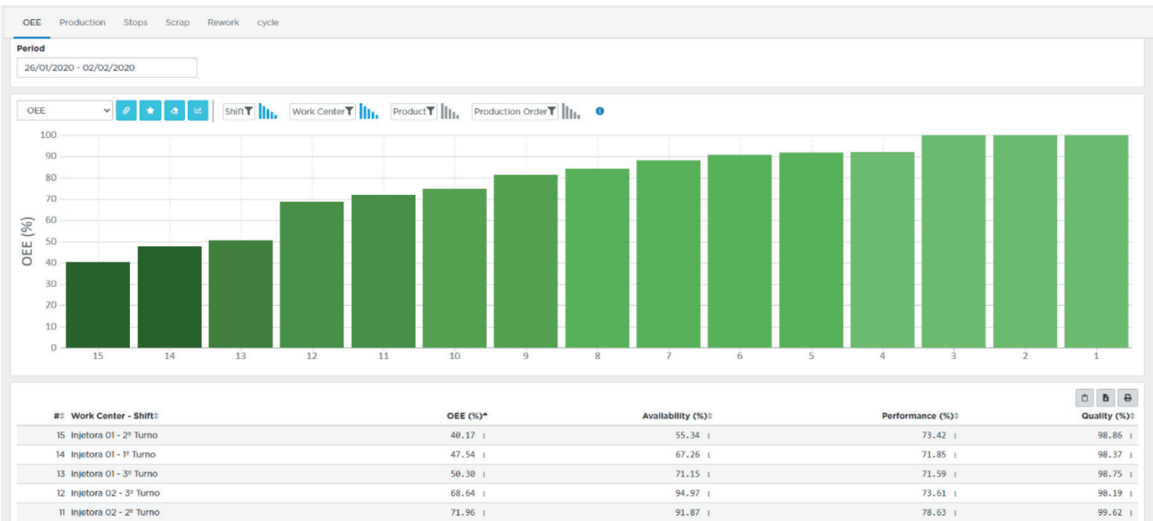


Figure 12. General OEE Dashboard [59]

MATURITY ASSESMENT

ITEM	Answer	Score	Dimension	Mean
Production Count	4. All WCs	4	Resources	
Machine Stops	4. All WCs	4		
Visibility	3. lots of devices	3		3,75
Collector Availability	4. > 97%	4		
Master data	3. partially coherent	3	Rigor	
Users	2. admin and 1 for WC	2		
Hierachy use	2. operators and supervisors	2		2,50
Orders Use	3. respected quantities	3		
Facility availability	3. complete in some departments	3	Routine	
Screens used	4. 4 or more screens always	4		
Notes	3. frequently in some WCs	3		3,25
Quality count	3. quantity/root cause delayed	3		
Losses detection	2. performance and root causes in some WCs	2	Run	
Action plans	0. no	0		
Self-evaluation	1. Informative but not trust	1		1,25
OEE is real	2. reasonable value(20-90%)	2		
Total:		**43**	**Worst mean**	**1,25**

Analyst obs.:

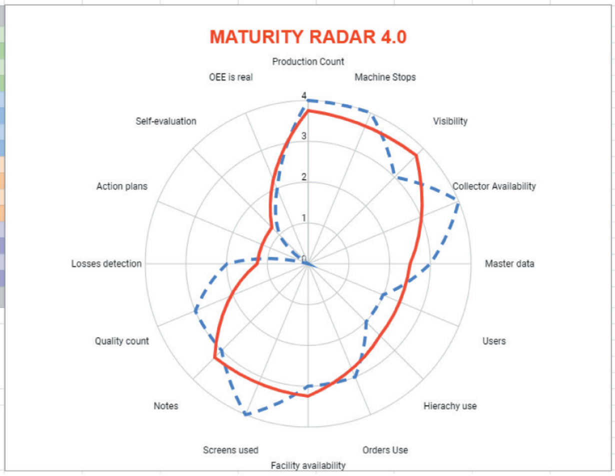


Figure 13. The RA-RE-RI-RO-RU Maturity Assessment [59]

it takes from 7 to 16 seconds for doing a complete analysis of all production issues for all work centres (it spends about 15 seconds to completely evaluate 2 months of operation of 20 work centres). It is to be highlighted that this assessment only refers to the maturity analysis. Many other actions are performed by softbot, automatically, proactively, and more accurately in the background on behalf of the managers [22].

The goal of this case is to show that softbots can do complex activities, no matter the underlying approaches used by a given company (maturity models, reasoning processes' algorithms, etc.).

5.4. Operator 4.0 + Softbot Integrated into Digital Twins

This case addresses the use of softbots as an intermediate actor between a Digital Twin (DT) and production managers. This does not aim to replace the native DT's user interface, but rather add a complementary and more symbiotic interaction layer towards leveraging the *Cognitive Operator 4.0* [112].

This implementation [112] was made on top of previous work, described in Case 1 (see Section 5.1). In this previous case, a softbot was developed to allow direct interaction between operators and a set of CPSs. A DT to mirror these CPSs was further developed allowing managers to have access to a sort of information via the DT's graphical user interface and the provided production dashboards.

The motivation to implement this case refers that, in a smart shop floor, although a DT can show production data via dashboards, many issues arose once the DT had been deployed, such as: (i) the data showed in a DT's dashboard is predefined and designed for a generic user, including the way it interacts with its users; (ii) the usual lack of integration between a DT and other enterprise systems makes users go through other information systems' GUIs (e.g., ERP or MEs) to pick the required information and to 'mentally' reason about it outside of the DT's environment; (iii) the human limitation to follow (with the eyes) many 'things' simultaneously in a DT visualization or CPS execution; and (iv) the many 'hidden information' in a DT visualization because of poor GUI design; among other issues.

In terms of DT, one of its parts is deployed at a local level. A *wrapper* was implemented to grab information from the CPS's stations in real-time and store them in the DT's database. There is a total of 135 PLCs' tags that indicate several statuses of the stations during their operation. The communication between the wrapper and the TIA Portal (see Section 5.1) is done via the OPC-DA protocol.

On the remote side, a DT has been developed (in *Python*) using the *Plant Simulation (PS) software*, from Siemens. The real-time visualization of the CPS operation is done via permanent access to the DT's database using the OPC-UA protocol. It can be accessed via its Python/COM API in the same way as its internal database. On the left side of the figure, it can be shown the DT environment with a partial view of the CPS in the background. The PC on the right side has the DT wrapper/OPC server. The DT appears on the left side, with two screens.

The developed *softbot* is a client application of the DT: the "DT softbot". It can access both other information that is not handled within the DT environment (e.g., the ERP) and the ones dealt with by the DT itself via its API. The so-called 'local softbot' corresponds to Case 5.1.

At this level, the operator: (i) can be provided with

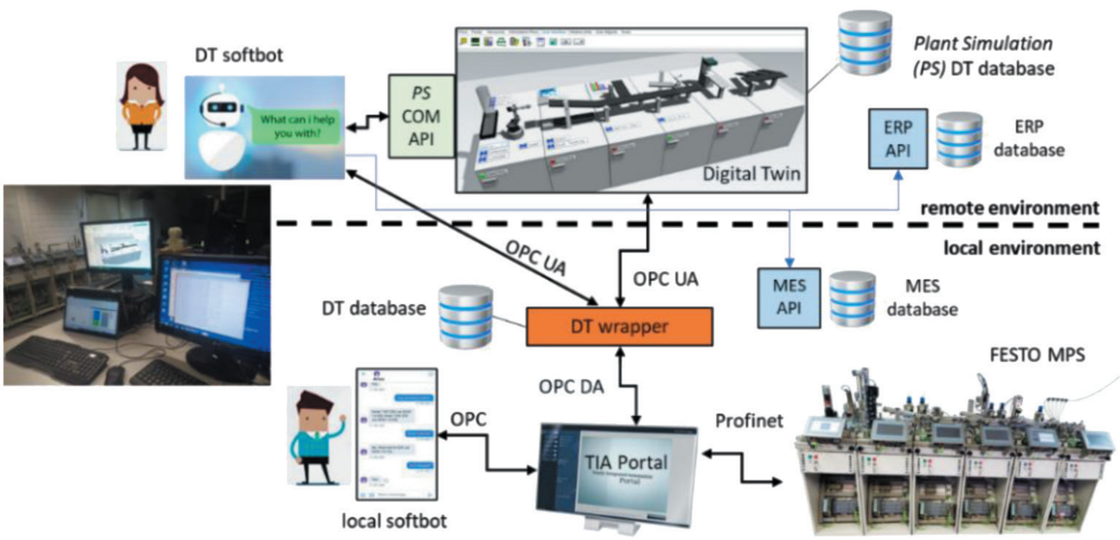


Figure 14. General Systems' Architecture [113]

past information (e.g., what has been produced, via accessing the Plant Simulation and/or the DT databases); (ii) can visualize current information (e.g., what is being produced, the machines in operation, bottlenecks formation, etc.); (iii) can have access to related information from other systems; (iv) can monitor and take care of the CPS execution; and (v) can make some (current and future) analytics (e.g., prediction) based on some data and performance indicators.

Scenario 1

In this scenario, the user checks the DT's analytics module and observes that the *Distribution station's OEE (Overall Equipment Effectiveness)* is too unstable. He realizes that some production orders will be delayed if this continues as the DT is predicting, but he does not know either which orders might be affected by that (this information is handled by the MES system) or the respective customers' names (this information is handled by the ERP system) so that he could warn both customers and the company's CRM area in advance.

As the DT does not have all the required information, this request is got by the softbot, which "reacts" to attend to it. The message is interpreted by the softbot (based on predefined keywords), and it is decomposed into "subtasks" (one for each correct subsystem – the MES and ERP in this case - that should be accessed via calling their APIs to get the information), and each subtask is executed (within/by the softbot) (see Figure 15).

Scenario 2

In this case, there is no user request. The softbot just executes the tasks that have been previously configured by the user on his behalf.

In this scenario, the softbot should automatically generate the utilization level of the *Distribution and Testing stations* at the end of each working shift for general analyses by users. As the *ARISA NEST* does not support internal graphical elements in its GUI, it does it calling and interoperating with the Plant Simulation's API to see the utilization level (see Figure 16).

Scenario 3

In this case, there is not a user request either. However, the pro-active mode relates to the possible autonomy level a softbot can have. For example, a softbot could always warn the user (e.g., sending him a message or getting into alarm) and order a given machine (the Distribution station, for example) to be turned off when its OEE gets lower than 20%. This value of 20% would be a result of a more sophisticated algorithm that calculated this after processing a long OEE historical series also considering the company's performance metrics.

In the developed prototype, that situation was just simulated. A simple algorithm (but it could be as complex as necessary) was implemented to act on the *physical Distribution station*, to be turned off in that OEE situation. Figure 17 zooms on the moment this situation occurs, also showing the change in the respective station's tag (from 'false' - it was turned on, to 'true' - it is now turned off).

The possibility to turn a given station off by the user is also allowed by pressing the red button of each station directly on the DT's graphical interface. However, this possibility poses some potential problems in terms of security and governance, as it means autonomously acting directly on physical equipment.

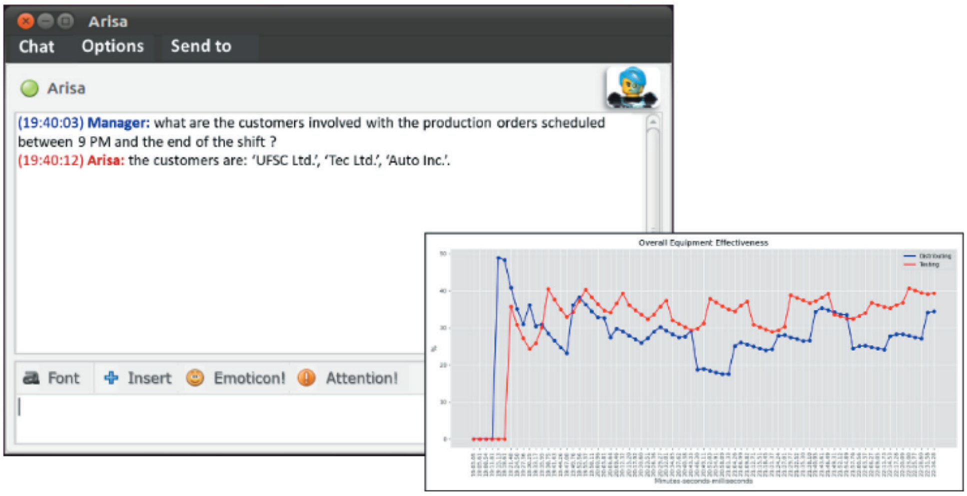


Figure 15. Scenario 1: Reactive + Request Decomposition + Access to other Systems + Analytics

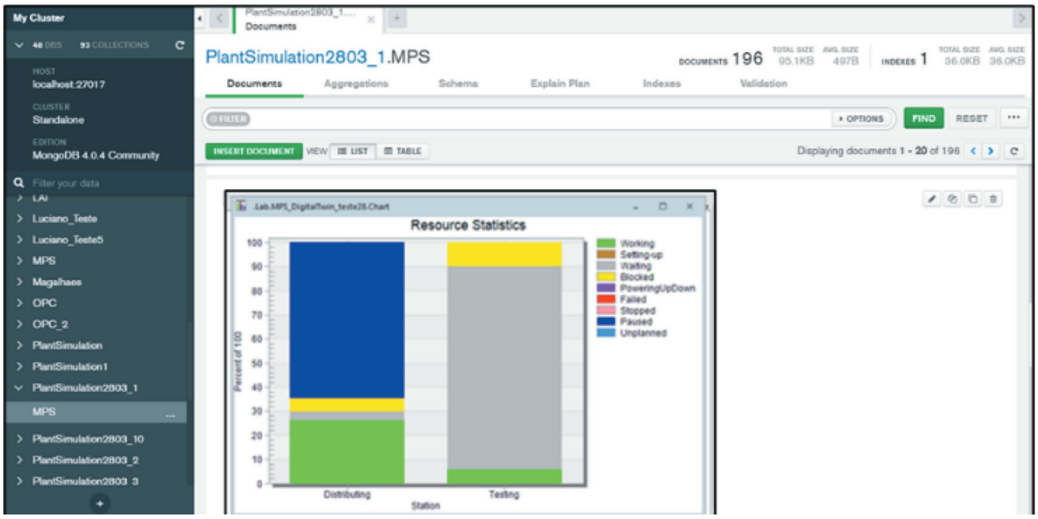


Figure 16. Scenario 2: Planned + Automatic Actions

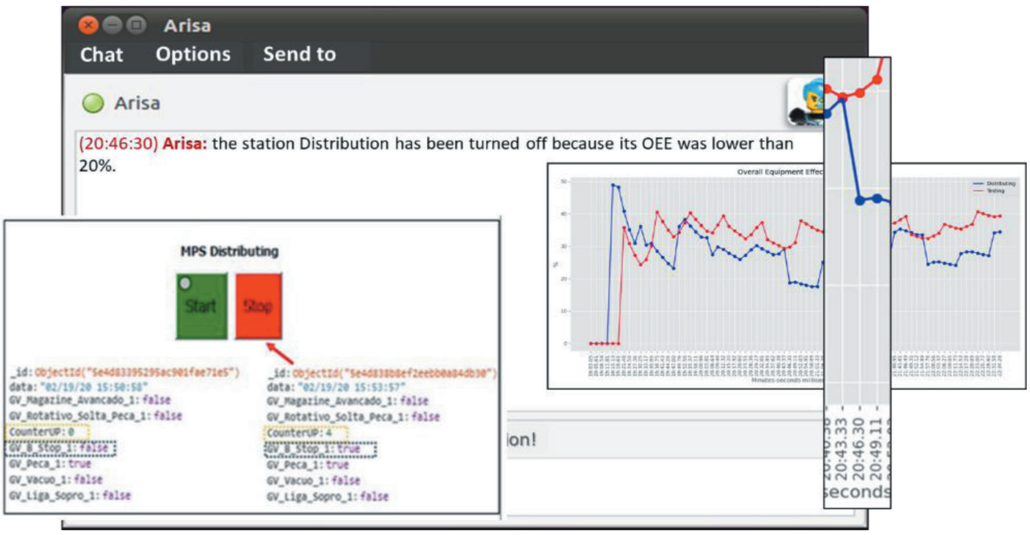


Figure 17. Scenario 3: Pro-Active + Actuation on the Real System

Another example of this scenario is to support Scenario 1, i.e., instead of doing the user to follow the OEE performance of the Distribution station "manually" over time, the softbot can proactively and permanently check it and warn the user when this situation happens (see Figure 17).

5.5. Operator 4.0 + Softbot Integrated into Augmented Reality

This case had the goal of showing scenarios where softbots and augmented reality (AR) are combined to create a working cognitive environment for the so-called "Resilient Operator 5.0" [113], extending the concept of the *Augmented Operator 4.0 subtype* [15]. In terms of domain problems, this work has focused on preventive maintenance, one of the most relevant issues being dealt with by industries in their chase for higher operational efficiency, zero error, and lower production costs [85]. The importance of maintenance operators being assisted by such technologies relies on increasing complexity in their daily activities, which includes: (i) continuous training as new products and versions are developed; (ii) manual maintenance in several and very different types of equipment; (iii) planning activities and lots of information to be constantly checked about each equipment; and (iv) the different levels of expertise and operators experience, leading to longer maintenance times and more errors; among others [85].

Like Case 5.3, this one was implemented in a real company, although as a proof-of-concept project. This company is a medium-sized enterprise specialising in producing customized industrial automated assembling solutions for the food sector. A machine called "hamburger cartooning automated system" has been chosen for the tests, and it is responsible for receiving ready pieces of raw hamburgers and packing them into ready-to-deliver paper boxes. This case is detailed and explained in [114].

The prototype has been developed as a 3D AR model. The cartooning system's parts were designed by the company in a CAD system (Solidworks). After being converted into a proper AR format, colours, and textures were added to the parts using the Blender software. The whole environment was implemented within the Unity framework/tool. Vuforia software was used to get the modelled parts and to load them in the Unity environment. Once there, all the parts' animations and choreographies, the design of buttons and heads-up displaying, functionalities programming (via scripts coded in C#), etc., could be implemented. This prototype was devised as an App to run on a smartphone, although other devices could be used, such as tablets and smart glasses.

Scenario 1

This proactive scenario refers to when the maintenance operator leaves his office and goes to the shop floor for their daily routine after knowing which machines should be checked and maintained based on previous reports. However, during his normal walk, a given machine's softbot (proactively) warns him that the machine has now an unexpected issue after receiving some information from sensors. This warning is sent out in an AR way (see Figure 18) when the softbot detects that the operator is close to the machine (this automatic 'geographic detection' has been so far implemented only in a simulated way).

Scenario 2

This reactive scenario refers to when the operator is doing the maintenance, starts an interaction with the softbot, and has some doubts about what or how to do it, whether experienced or beginner operators. A very simple 'expert system' inside the softbot is implemented, guiding the operator during the operation. All interactions happen via the AR environment, running on the operator's smartphone (see Figure 19 left side), and they can be made both via voice and text. In the case the operator wants more information about some specific aspect, an auxiliary interface pops up; in this case, the operator needs information about the 'casquilho' part and the explanation is shown (see Figure 19 right side).

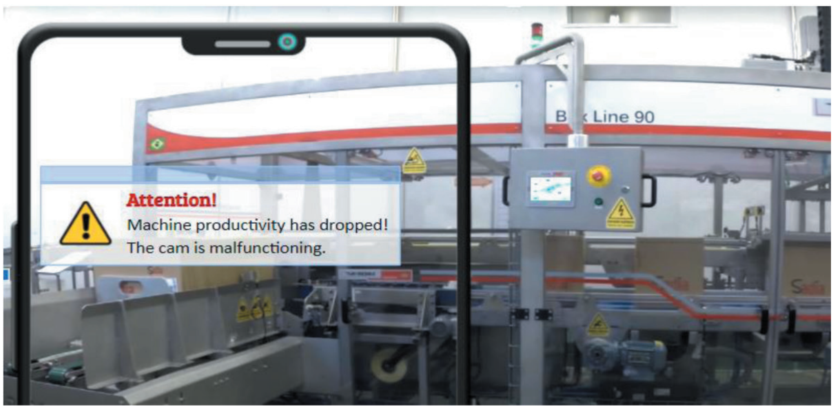


Figure 18. Scenario 1: Proactive Case – Warning about the Machine's Situation

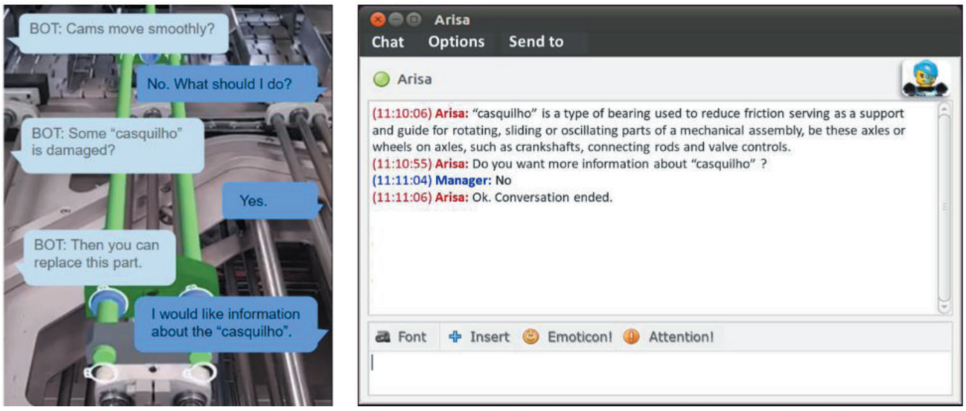


Figure 19. Scenario 2: Reactive Case – Assisting the Operator during a Maintenance Activity

Scenario 3

This planned scenario refers to when the operator is about to end his shift and wants to check the current status of the maintenance activities against what was planned. Reports are made available via the App and are automatically generated by the softbot, as planned. As *ARISA NEST* does not support per se graphical interfaces, the reports are stored and accessed at the derived softbot's web area (see Figure 20). In terms of functionality, this scenario is similar to some previous ones. The difference is that, in this case, this report is generated within the common AR environment, hence concentrating all the needed information in one place.

6. Conclusions

This paper has presented the results of a research whose essential goal was to evaluate the potential of softbots as a complementary technology to create a more symbiotic environment with the so-called *Operator 4.0*, the Smart Operator type in more particular. In this context, the concept of *Softbots 4.0* has been also introduced.

Five cases have been implemented, covering different scenarios involving manufacturing cyber-physical systems. Considering the usual issues managers and operators deal with in their daily activities, it could be observed, essentially via the implemented cases, that *Softbots 4.0* is a powerful approach to help in improving smart production management and shop control. This could be attested based on the experimental results:

- *Softbots* help hide systems' complexity, aiding managers in digging and finding the needed information over different and distributed silos of data (sometimes duplicated and stored in different sources) generated by different legacy systems.
- Such information access was faster and more trustworthy, as the *softbot* knows where to get the very right information and does that already handling interoperability problems. Besides that, it can help in granting only authorized people to request/have access to some type of information, hence helping companies in e.g., governance and GPDR compliance.
- *Softbots* could bring information in a more synthesized, structured, and compiled way,

Figure 20. Scenario 3: Planned Case – Generating Maintenance Reports Automatically

sometimes composing a given data set (from the access to different systems) or making some previous calculations before finally sending/showing it to users.
- The way users and softbots interact with each other is more similar to natural language speaking, helping to create a more symbiotic, "friendly", social working environment.
- *Softbots* could help users not only with their on-demand requests, but also with performing scheduled, complex, repetitive, and/or numerous tasks automatically for or on behalf of them. Yet, they also do actions proactively, e.g., to permanently monitor some production aspects and to warn users about them when needed. As a result, users can dedicate more time to activities that add more value to the company, and that indeed require more intelligence, creativity, and critical decision-making.
- *Softbots* can reason about data and help users in business/production analytics. They can also help users to understand what systems mean when suggesting, e.g., prescriptive actions.

One can note that some operations done via the softbot could be done by the user and/or provided by legacy systems, i.e., the softbot is not always the only way to access some data or to execute some tasks. However, it is important to remember that one of the softbot's goals is to ease that or to hide some difficulty or complexity to do that by diverse people, also consolidating the needed information in one common user interface or environment.

The implemented cases have addressed only intra-organizational scenarios, with all the machines/CPS belonging to the same organization, and only providing basic security mechanisms supported by the used tools (security is an extremely critical issue in the industrial softbots area, but it was out of the scope of this work). However, considering the potentialities of current ICT technologies, softbots from different industries can collaborate via messages and information exchange towards helping industries (e.g., in a Virtual Enterprise or Supply Chain) in some activities, like collaborative planning, proactive prevention of the bullwhip effect, among many others. All that can be performed on-demand/under users' requests or following automatic pre-scheduled actions, or in an autonomous/proactive way taking the user mostly as the central point of decision-making. The development of higher *human-automation symbiosis* can leverage the creation of more sustainable manufacturing workforces in Industry 4.0, enhancing operational excellence, inclusiveness, satisfaction and motivation, safety, training, and continuous learning.

Important to mention that softbots do not aim at replacing managers, but at helping them instead. If on the one hand softbots can provide such help, on the other hand, it could be observed that managers' experience and insights keep being as important to make more refined analyses as well as to better tune the suggested actions to the reality of their companies. Along the same line, some enterprise information systems (like ERP and MES) keep playing a crucial role in the companies, so they are not supposed to be replaced by softbots either. The chatting part of softbots simply represents a more natural and smarter way for operators to interact with computer information systems and machines.

The implementation of real manufacturing scenarios requires facing a sort of tough issues and takes time. This includes e.g.: (i) the preparatory ETL (i.e., extraction, transformation and loading of data) steps and the modelling and integration of all the several necessary business processes and related interaction with humans; (ii) business processes are very different from each other and there are usually many different governance rules to be respected. This means, for example, that some more critical actions should be set up to be supervised or authorized by humans when softbots are executed; and (iii) the way softbots' knowledge base is modelled, populated, and maintained throughout its life cycle is crucial to guarantee answers' correctness, accuracy, and usefulness. This demands very experienced software and data science engineers, systems integrators, process domain experts, and pilot users to work together in the designing, testing, and deployment phases, including the conception of all types and dialogue situations between people and the softbot. All this gives a glimpse of the difficulties that implementing a full-fledged softbot represents. The following sentence illustrates that: *"The computational linguistics community has been looking at discourse phenomena since the inception of the field. [...] In their striving to move the technology forward, the next milestone [...] to tackle is around truly conversational interactions, [...] the ability to take into account discourse contexts rather than just treating a dialog as a sequence of independent conversational pairs"* [62].

Once deployed, softbot usage also requires deep training of operators and managers. This is important for companies to get the full benefits of this technology as well as for people to be indeed prepared to understand the provided information for better analysis and decision-making. This is not trivial.

For example, a recent survey with forty-four leading worldwide manufacturers in Industry 4.0 technologies implementation in plant environments showed that *data analytics* is the top skill to be reinforced in training programs [115]. This is a critical aspect. The market has been delivering some powerful analytics-related products linked to chatbots (e.g., [116]), but with a too strong emphasis on the product itself, without stressing the underlying hard issues to deal with to deploy real solutions.

CPS wrappers can facilitate the integration between physical CPSs and softbots. However, it is important to bear in mind that this keeps involving distributed and heterogeneous systems, which use multiple and legacy technologies and protocols at different levels. Therefore, interoperability remains a hard issue to face to guarantee correct, efficient, and reliable communication.

The more softbots are adopted, the more companies and production management can get dependent on them. From the computing point of view, this creates a potentially central point of failure in a system, an issue that would become critical to address. From the business process perspective, this issue can be even more relevant to handle as more advanced softbots are likely to be deeply integrated into other systems (sometimes from other companies and sometimes to critical systems, machines, and infrastructures), exchanging information and triggering actions close to them, besides the intensive interaction with users. Therefore, proper security schemas as well as fault-tolerance/resilience strategies should be internally (or systemically) implemented, as proposed in [117].

The general area of softbots is relatively old, but now it is getting more mature to be used in real cases [26]. Nevertheless, there are still several complex intrinsic issues to be coped with (as previously mentioned) and that represents scientific open points for new research. They include, for example, a more intense use of machine learning in messages understanding, processing, and reasoning; conversations and answers design regarding different levels of operators' experience; natural language processing and translations; semantics interoperability, including dealing with different human idioms and contexts; privacy data control and sharing about users' information; cybersecurity, guaranteeing that softbots are not hacked or do malicious actions; and analytics based on big data. Emergent issues, such as the use of softbots to assist different types of unpaired operators, starts to get increased attention.

Other works in the literature have mentioned additional issues when dealing with softbots in manufacturing (or equivalent terms), to be highlighted [72]: the usual very long time to deploy and to maintain a softbot ready-to-use and always up-to-date and integrated; some cultural resistance from operators to adopt it confidently; most of the current implementations make use of keyboard and voice to interact with softbots. However, new means are emerging (such as virtual reality, augmented reality, wearables, and metaverse), sometimes combined with other technologies (such as IoT and big data), which brings up new technological and usage challenges; eventual misuse of information got from operators' performance during their work; connectivity problems, especially in industries and regions where communication infrastructures are not so robust; cybersecurity cares (in the softbot itself and the communication channel being used); the use of softbots to help companies in supervising working habits of operators for security and health purposes; business and context awareness in different situations and operators; and the risk of operators to become 'lazy' when interacting with the softbot, expecting it can solve every problem, have all answers, check all data, and recommend all decisions.

Despite the mentioned difficulties and some challenges that a larger adoption of softbots by industries represents, all the researchers and developers have been pointing out the high potential of this technology in terms of higher human assistance, productivity, and a more friendly environment in the execution of tasks in the increasingly digital and connected factory world.

Funding

This work was partially supported by the Brazilian Agency for Higher Education (CAPES), under project PrInt CAPES-UFSC "Automation 4.0".

References

[1] L. M. Camarinha-Matos, R. Fornasiero, and H. Afsarmanesh, "Collaborative Networks as a Core Enabler of Industry 4.0 BT - Collaboration in a Data-Rich World," 2017, pp. 3–17, doi: 10.1007/978-3-319-65151-4_1.

[2] Y. Liao, F. Deschamps, E. de F. R. Loures, and L. F. P. Ramos, "Past, present and future of Industry 4.0 - a systematic literature review and research agenda proposal," Int. J. Prod. Res., vol. 55, no. 12, pp. 3609–3629, Jun. 2017, doi: 10.1080/00207543.2017.1308576.

[3] A. Kusiak, "Smart manufacturing," Int. J. Prod. Res., vol. 56, no. 1–2, pp. 508–517, Jan. 2018, doi: 10.1080/00207543.2017.1351644.

[4] J. Lee, B. Bagheri, and H.-A. Kao, "Recent Advances and Trends of Cyber-Physical Systems and Big Data

Analytics in Industrial Informatics," in International Conference on Industrial Informatics (INDIN), 2014, doi: 10.13140/2.1.1464.1920.

[5] M. Hermann, T. Pentek, and B. Otto, "Design principles for industrie 4.0 scenarios," in Proceedings of the Annual Hawaii International Conference on System Sciences, 2016, pp. 3928–3937, doi: 10.1109/HICSS.2016.488.

[6] S. Kaushik, Production and Materials Management. Centrum Press, 2014.

[7] V. B. Vukšić, M. P. Bach, and A. Popovič, "Supporting performance management with business process management and business intelligence: A case analysis of integration and orchestration," Int. J. Inf. Manage., vol. 33, no. 4, pp. 613–619, 2013, doi: https://doi.org/10.1016/j.ijinfomgt.2013.03.008.

[8] D. Romero and F. Vernadat, "Enterprise information systems state of the art: Past, present and future trends," Comput. Ind., vol. 79, pp. 3–13, 2016, doi: https://doi.org/10.1016/j.compind.2016.03.001.

[9] J. D. Méndez, R. J. Rabelo, F. Baldo, and M. H. Cancian, "A Plug and Play Integration Model for Virtual Enterprises," in Collaborative Networks of Cognitive Systems. PRO-VE 2018. IFIP Advances in Information and Communication Technology, 2018, pp. 312–324, doi: 10.1007/978-3-319-99127-6_27.

[10] R. J. Rabelo, D. Romero, and S. P. Zambiasi, "Softbots Supporting the Operator 4.0 at Smart Factory Environments," in Advances in Production Management Systems. Smart Manufacturing for Industry 4.0. APMS, 2018, pp. 456–464, doi: 10.1007/978-3-319-99707-0_57.

[11] ASG Technologies, "THE ASG TECHNOLOGIES 2020 SURVEY REPORT: Barriers to Success with Information-Powered Business: Bad Data Burdens the Modern Enterprise," 2020.

[12] I. Spasojević, S. Havzi, D. Stefanović, S. Ristić, and U. Marjanović, "Research Trends and Topics in IJIEM from 2010 to 2020: A Statistical History," Int. J. Ind. Eng. Manag., vol. 12, no. 4, pp. 228–242, 2021, doi: 10.24867/IJIEM-2021-4-290.

[13] A. McDermott, "Information Overload Is Crushing You. Here are 11 Secrets That Will Help.," Workzone, 2017. [Online]. Available: https://www.workzone.com/blog/information-overload/. [Accessed: 08-Mar-2021].

[14] R. J. Rabelo, S. P. Zambiasi, and D. Romero, "Collaborative Softbots: Enhancing Operational Excellence in Systems of Cyber-Physical Systems," in Collaborative Networks and Digital Transformation. PRO-VE 2019. IFIP Advances in Information and Communication Technology, 2019, pp. 55–68, doi: 10.1007/978-3-030-28464-0_6.

[15] D. Romero et al., "Towards an Operator 4.0 Typology: A Human-Centric Perspective on the Fourth Industrial Revolution Technologies," in Internation Conference on Computers & Industrial Engineering (CIE46), 2016, pp. 1–11.

[16] D. Romero, T. Wuest, J. Stahre, and D. Gorecky, "Social Factory Architecture: Social Networking Services and Production Scenarios Through the Social Internet of Things, Services and People for the Social Operator 4.0," in Advances in Production Management Systems. The Path to Intelligent, Collaborative and Sustainable Manufacturing, 2017, pp. 265–273, doi: 10.1007/978-3-319-66923-6_31.

[17] E. Valette, H. Bril El-Haouzi, and G. Demesure, "Toward a Social Holonic Manufacturing Systems Architecture Based on Industry 4.0 Assets," in Service Oriented, Holonic and Multi-Agent Manufacturing Systems for Industry of the Future, 2021, pp. 286–295, doi: 10.1007/978-3-030-69373-2_20.

[18] J. Leng, P. Jiang, C. Liu, and C. Wang, "Contextual self-organizing of manufacturing process for mass individualization: a cyber-physical-social system approach," Enterp. Inf. Syst., vol. 14, no. 8, pp. 1124–1149, Sep. 2020, doi: 10.1080/17517575.2018.1470259.

[19] M. Breque, L. De Nul, and A. Petridis, "Industry 5.0: towards a sustainable, human-centric and resilient European industry," Whitepaper, European Commission, Directorate-General for Research and Innovation, 2021.

[20] J. de A. Dornelles, N. F. Ayala, and A. G. Frank, "Smart Working in Industry 4.0: How digital technologies enhance manufacturing workers' activities," Comput. Ind. Eng., vol. 163, p. 107804, 2022, doi: 10.1016/j.cie.2021.107804.

[21] B. Meindl, N. F. Ayala, J. Mendonça, and A. G. Frank, "The four smarts of Industry 4.0: Evolution of ten years of research and future perspectives," Technol. Forecast. Soc. Change, vol. 168, p. 120784, 2021, doi: 10.1016/j.techfore.2021.120784.

[22] D. Romero, J. Stahre, and M. Taisch, "The Operator 4.0: Towards socially sustainable factories of the future," Comput. Ind. Eng., vol. 139, p. 106128, 2020, doi: 10.1016/j.cie.2019.106128.

[23] D. P. Valentina, D. S. Valentina, M. Salvatore, and R. Stefano, "Smart operators: How Industry 4.0 is affecting the worker's performance in manufacturing contexts," Procedia Comput. Sci., vol. 180, pp. 958–967, 2021, doi: 10.1016/j.procs.2021.01.347.

[24] T. Schwartz et al., "Hybrid Teams: Flexible Collaboration Between Humans, Robots and Virtual Agents," in Multiagent System Technologies, 2016, pp. 131–146, doi: 10.1007/978-3-319-45889-2_10.

[25] W. Kopeć et al., "Hybrid Approach to Automation, RPA and Machine Learning: a Method for the Human-centered Design of Software Robots," in CSCW 2018 The 21st ACM Conference on Computer-Supported Cooperative Work and Social Computing, 2018.

[26] Chatbots Magazine, "Chatbot Report 2019: Global Trends and Analysis," Chatbots Magazine, 2019. [Online]. Available: https://chatbotsmagazine.com/chatbot-report-2019-global-trends-and-analysis-a487afec05b. [Accessed: 08-Mar-2021].

[27] C. Lebeuf, A. Zagalsky, M. Foucault, and M.-A. Storey, "Defining and Classifying Software Bots: A Faceted Taxonomy," in 2019 IEEE/ACM 1st International Workshop on Bots in Software Engineering (BotSE), 2019, pp. 1–6, doi: 10.1109/BotSE.2019.00008.

[28] A. Elliott, The Culture of AI, 1st Editio. London, UK: Taylor & Francis Group, 2018.

[29] Gartner Inc., "Gartner Predicts 25 Percent of Digital Workers Will Use Virtual Employee Assistants Daily by 2021," Newsroom Press Releases, 2019. [Online]. Available: https://www.gartner.com/en/newsroom/press-releases/2019-01-09-gartner-predicts-25-percent-of-digital-workers-will-u. [Accessed: 09-Mar-2021].

[30] D. Romero, P. Gaiardelli, D. Powell, T. Wuest, and M. Thürer, "Rethinking Jidoka Systems under Automation & Learning Perspectives in the Digital Lean Manufacturing World," IFAC-PapersOnLine, vol. 52, no. 13, pp. 899–903, 2019, doi: 10.1016/j.ifacol.2019.11.309.

[31] D. Romero, P. Gaiardelli, D. Powell, T. Wuest, and M. Thürer, "Total Quality Management and Quality Circles in the Digital Lean Manufacturing World," in Advances in Production Management Systems. Production Management for the Factory of the Future, 2019, pp. 3–11, doi: 10.1007/978-3-030-30000-5_1.

[32] J. Preece, Y. Rogers, and H. Sharp, Interaction Design, Beyond Human-Computer Interaction. Wiley, 2006.

[33] P. Papcun, E. Kajáti, and J. Koziorek, "Human Machine Interface in Concept of Industry 4.0," in 2018 World

Symposium on Digital Intelligence for Systems and Machines (DISA), 2018, pp. 289–296, doi: 10.1109/DISA.2018.8490603.
[34] O. Korhan, M. Fallaha, Z. M. Çınar, and Q. Zeeshan, "The Impact of Industry 4.0 on Ergonomics," in Ergonomics - New Insights, O. Korhan, Ed. Rijeka: IntechOpen, 2022, p. Ch. 6.
[35] J. Butsmann, "The future about Digital Assistants and Chatbots in SAP S/4HANA Cloud (incl. on premise)," SAP blog, 2022. [Online]. Available: https://blogs.sap.com/2022/04/08/the-future-about-digital-assistants-and-chatbots-in-sap-s-4hana-cloud-incl.-on-premise/. [Accessed: 22-Feb-2023].
[36] E. Hitzke, "6 powerful reasons why Oracle Digital Assistant is a must-have," Oracle blog, 2020. [Online]. Available: https://blogs.oracle.com/ai-and-datascience/post/6-powerful-reasons-why-oracle-digital-assistant-is-a-must-have. [Accessed: 22-Feb-2023].
[37] O. Etzioni and D. Weld, "A Softbot-Based Interface to the Internet," Commun. ACM, vol. 37, no. 7, pp. 72–76, Jul. 1994, doi: 10.1145/176789.176797.
[38] J.-H. Kim, "Ubiquitous Robot," in Computational Intelligence, Theory and Applications, 2005, pp. 451–459, doi: 10.1007/3-540-31182-3_41.
[39] J. Weizenbaum, "ELIZA—a Computer Program for the Study of Natural Language Communication between Man and Machine," Commun. ACM, vol. 9, no. 1, pp. 36–45, Jan. 1966, doi: 10.1145/365153.365168.
[40] B. Hermans, "Intelligent Software Agents on the Internet: Chapters 1-3," First Monday, vol. 2, no. 3, Mar. 1997, doi: 10.5210/fm.v2i3.514.
[41] S. Bocionek, "Software secretaries: learning and negotiating personal assistants for the daily office work," in Proceedings of IEEE International Conference on Systems, Man and Cybernetics, 1994, vol. 1, pp. 7–12 vol.1, doi: 10.1109/ICSMC.1994.399802.
[42] L. Dent, J. Boticario, J. McDermott, T. Mitchell, and D. Zabowski, "A Personal Learning Apprentice," in Proceedings of the Tenth National Conference on Artificial Intelligence, 1992, pp. 96–103.
[43] O. Etzioni, "Intelligence without Robots: A Reply to Brooks," AI Mag., vol. 14, no. 4, p. 7, Mar. 1993, doi: 10.1609/aimag.v14i4.1065.
[44] T. M. Mitchell, R. Caruana, D. Freitag, J. McDermott, and D. Zabowski, "Experience with a Learning Personal Assistant," Commun. ACM, vol. 37, no. 7, pp. 80–91, Jul. 1994, doi: 10.1145/176789.176798.
[45] O. Etzioni, H. M. Levy, R. B. Segal, and C. A. Thekkath, "The softbot approach to OS interfaces," IEEE Softw., vol. 12, no. 4, pp. 42–51, 1995, doi: 10.1109/52.391830.
[46] S. Reese Hedberg, "Intelligent Agents: The First Harvest of Softbots Looks Promising," IEEE Expert Intell. Syst. Their Appl., vol. 10, no. 4, pp. 6–9, Aug. 1995.
[47] A. Javernik, B. Buchmeister, and R. Ojstersek, "Impact of Cobot parameters on the worker productivity: Optimization challenge," Adv. Prod. Eng. Manag., vol. 17, no. 4, pp. 494–504, 2022, doi: 10.14743/apem2022.4.451.
[48] O. Baujard, V. Baujard, S. Aurel, C. Boyer, and R. D. Appel, "A multi-agent softbot to retrieve medical information on Internet.," Stud. Health Technol. Inform., vol. 52, no. 1, pp. 150–154, 1998.
[49] O. Etzioni, "Moving up the Information Food Chain: Deploying Softbots on the World Wide Web," in Proceedings of the Thirteenth National Conference on Artificial Intelligence - Volume 2, 1996, pp. 1322–1326.
[50] M. N. Huhns and M. P. Singh, "Personal assistants," IEEE Internet Comput., vol. 2, no. 5, pp. 90–92, 1998, doi: 10.1109/4236.722236.
[51] Microsoft, "Cortona," 2022. [Online]. Available: https://www.microsoft.com/en-us/cortana. [Accessed: 08-Mar-2022].
[52] C. Doctorow, "I Want Sandy – perfect productivity email bot is free and public," Boing Boing, 2007. [Online]. Available: https://boingboing.net/2007/11/14/i-want-sandy-perfect.html. [Accessed: 08-Mar-2022].
[53] Apple, "Siri," 2022. [Online]. Available: https://www.apple.com/siri/?cid=oas-us-domains-siri.com/. [Accessed: 22-Mar-2023].
[54] PAL, "The PAL Framework," 2023. [Online]. Available: https://pal.sri.com/. [Accessed: 22-Feb-2023].
[55] N. Chauvat, "Narval, the Intelligent Personal Assistant or How the French Linux Gazette is Built," Linux Gazette, 2011. [Online]. Available: https://linuxgazette.net/issue59/chauvat.html. [Accessed: 22-Feb-2023].
[56] IBM Corporation, "Watson Assistant: Build better virtual agents, powered by AI," 2023. [Online]. Available: https://www.ibm.com/products/watson-assistant. [Accessed: 22-Feb-2023].
[57] Alexa, "What Is Alexa?," 2023. [Online]. Available: https://developer.amazon.com/en-US/alexa/. [Accessed: 22-Feb-2023].
[58] OpenAI, "ChatGPT," 2023. [Online]. Available: https://openai.com/. [Accessed: 22-Feb-2023].
[59] B. Abner, R. J. Rabelo, S. P. Zambiasi, and D. Romero, "Production Management as-a-Service: A Softbot Approach," in Advances in Production Management Systems. Towards Smart and Digital Manufacturing, 2020, pp. 19–30, doi: 10.1007/978-3-030-57997-5_3.
[60] N. Zivlak, S. Rakic, U. Marjanovic, D. Ciric, and B. Bogojevic, "The Role of Digital Servitization in Transition Economy: An SNA Approach," Teh. Vjesn. - Tech. Gaz., vol. 28, no. 6, pp. 1912–1919, 2021, doi: 10.17559/TV-20210325083229.
[61] O. Salunkhe and Å. F. Berglund, "Industry 4.0 enabling technologies for increasing operational flexibility in final assembly," Int. J. Ind. Eng. Manag., vol. 13, no. 1, pp. 38–48, 2022, doi: 10.24867/IJIEM-2022-1-299.
[62] R. Dale, "The return of the chatbots," Nat. Lang. Eng., vol. 22, pp. 811–817, Sep. 2016, doi: 10.1017/S1351324916000243.
[63] M. Hoyle and C. Lueg, "Open Sesame!: A Look at Personal Assistants," in International Conference on the Practical Application of Intelligent Agents and Multiagent Technology, 1997, pp. 51–60.
[64] A. Angehrn, T. Nabeth, L. Razmerita, and C. Roda, "K-InCA: using artificial agents to help people learn and adopt new behaviours," in Proceedings IEEE International Conference on Advanced Learning Technologies, 2001, pp. 225–226, doi: 10.1109/ICALT.2001.943906.
[65] M. N. Huhns, "Agents as Web services," IEEE Internet Comput., vol. 6, no. 4, pp. 93–95, 2002, doi: 10.1109/MIC.2002.1020332.
[66] S. Schiaffino and A. Amandi, "Polite personal agent," IEEE Intell. Syst., vol. 21, no. 1, pp. 12–19, 2006, doi: 10.1109/MIS.2006.15.
[67] S. P. Zambiasi and R. J. Rabelo, "A Proposal for Reference Architecture for Personal Assistant Software Based on SOA," IEEE Lat. Am. Trans., vol. 10, no. 1, pp. 1227–1234, 2012, doi: 10.1109/TLA.2012.6142466.
[68] E. Monchalin, "An introduction to Chatbots," ATOS IT Challenge, 2017. [Online]. Available: an-introduction-to-chatbots. [Accessed: 23-Feb-2022].
[69] F. Longo and A. Padovano, "Voice-enabled Assistants of the Operator 4.0 in the Social Smart Factory: Prospective role and challenges for an advanced human–machine interaction," Manuf. Lett., vol. 26, pp. 12–16, 2020, doi: 10.1016/j.mfglet.2020.09.001.

[70] M. Richards, Software Architecture Patterns. O'Reilly Media, Inc., 2015.
[71] S. Wellsandt, K. Hribernik, and K.-D. Thoben, "Anatomy of a Digital Assistant," in Advances in Production Management Systems. Artificial Intelligence for Sustainable and Resilient Production Systems, 2021, pp. 321–330, doi: 10.1007/978-3-030-85910-7_34.
[72] J. Ferber, Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence, 1st ed. USA: Addison-Wesley Longman Publishing Co., Inc., 1999.
[73] R. J. Rabelo, M. S. Wangham, R. Schmidt, and J. S. Fraga, "Trust Building in the Creation of Virtual Enterprises in Mobile Agent-Based Architectures," in Processes and Foundations for Virtual Organizations, 2004, pp. 65–72, doi: 10.1007/978-0-387-35704-1_7.
[74] R. Kar and R. Haldar, "Applying Chatbots to the Internet of Things: Opportunities and Architectural Elements," Int. J. Adv. Comput. Sci. Appl., vol. 7, no. 11, pp. 147–154, 2016, doi: 10.14569/ijacsa.2016.071119.
[75] B. C. Williams and P. P. Nayak, "Immobile Robots AI in the New Millennium," AI Mag., vol. 17, no. 3 SE-Articles, p. 16, Mar. 1996, doi: 10.1609/aimag.v17i3.1229.
[76] H. Van Brussel, J. Wyns, P. Valckenaers, L. Bongaerts, and P. Peeters, "Reference architecture for holonic manufacturing systems: PROSA," Comput. Ind., vol. 37, no. 3, pp. 255–274, 1998, doi: https://doi.org/10.1016/S0166-3615(98)00102-X.
[77] M. Vetter, "Quality Aspects of Bots," in Software Quality and Software Testing in Internet Times, 2002, pp. 165–184, doi: 10.1007/978-3-642-56333-1_11.
[78] L. Erlenhov, F. de Oliveira Neto, and P. Leitner, "An Empirical Study of Bots in Software Development -- Characteristics and Challenges from a Practitioner's Perspective," in 28th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering, 2020, pp. 445–455, doi: 10.1007/s42461-019-00104-9.
[79] J. Lööw, L. Abrahamsson, and J. Johansson, "Mining 4.0—the Impact of New Technology from a Work Place Perspective," Mining, Metall. Explor., vol. 36, no. 4, pp. 701–707, 2019, doi: 10.1007/s42461-019-00104-9.
[80] A. F. Özçekiç, "The Role of ChatBot in Industry 4.0," Chatbots Life, 2017. [Online]. Available: https://chatbotslife.com/the-role-of-chatbot-in-industry-4-0-acf2b0ac8bb6. [Accessed: 22-Feb-2023].
[81] B. Stackpole, "Are Virtual Assistants Headed to the Plant Floor?," AutomationWorld, 2020. [Online]. Available: https://www.automationworld.com/home/article/21121066/are-virtual-assistants-headed-to-the-plant-floor. [Accessed: 22-Feb-2023].
[82] J. Barata and L. M. Camarinha-Matos, "Coalitions of Manufacturing Components for Shop Floor Agility - the CoBASA Architecture," Int. J. Netw. Virtual Organ., vol. 2, no. 1, pp. 50–77, Sep. 2003, doi: 10.1504/IJNVO.2003.003518.
[83] R. J. Rabelo and L. M. Camarinha-Matos, "Negotiation in multi-agent based dynamic scheduling," Robot. Comput. Integr. Manuf., vol. 11, no. 4, pp. 303–309, 1994, doi: 10.1016/0736-5845(95)00007-0.
[84] S. Wellsandt et al., "Hybrid-augmented intelligence in predictive maintenance with digital intelligent assistants," Annu. Rev. Control, vol. 53, pp. 382–390, 2022, doi: 10.1016/j.arcontrol.2022.04.001.
[85] S. Wellsandt, M. Foosherian, K. Lepenioti, M. Fikardos, G. Mentzas, and K.-D. Thoben, "Supporting Data Analytics in Manufacturing with a Digital Assistant," in Advances in Production Management Systems. Smart Manufacturing and Logistics Systems: Turning Ideas into Action, 2022, pp. 511–518, doi: 10.1007/978-3-031-16411-8_59.
[86] W. Roush, "Immobots take control," Technol. Rev., vol. 105, no. 10, p. 37, 2002.
[87] P. Adolphs et al., "Reference Architecture Model Industrie 4.0 (RAMI4.0)," 2015.
[88] D. Gorecky, M. Schmitt, M. Loskyll, and D. Zühlke, "Human-machine-interaction in the industry 4.0 era," in 2014 12th IEEE International Conference on Industrial Informatics (INDIN), 2014, pp. 289–294, doi: 10.1109/INDIN.2014.6945523.
[89] C. Krupitzer et al., "A Survey on Human Machine Interaction in Industry 4.0," arXiv.org, vol. 1, no. 1, 2020, doi: 10.48550/arXiv.2002.01025.
[90] S. Singh and P. Tretten, "Operator 4.0 Within the Framework of Industry 4.0," in Applications and Challenges of Maintenance and Safety Engineering in Industry 4.0, New York, NY, USA: IGI Global, 2020, pp. 120–133.
[91] P. Valckenaers and H. Van Brussel, Design for the Unexpected: from Holonic Manufacturing Systems towards a Human Mechatronics Society. Butterworth-Heinemann, 2015.
[92] Y. Lu, "Industry 4.0: A survey on technologies, applications and open research issues," J. Ind. Inf. Integr., vol. 6, pp. 1–10, 2017, doi: 10.1016/j.jii.2017.04.005.
[93] L. Erlenhov, F. G. de O. Neto, R. Scandariato, and P. Leitner, "Current and Future Bots in Software Development," in 2019 IEEE/ACM 1st International Workshop on Bots in Software Engineering (BotSE), 2019, pp. 7–11, doi: 10.1109/BotSE.2019.00009.
[94] C. Matthies, F. Dobrigkeit, and G. Hesse, "An Additional Set of (Automated) Eyes: Chatbots for Agile Retrospectives," in 2019 IEEE/ACM 1st International Workshop on Bots in Software Engineering (BotSE), 2019, pp. 34–37, doi: 10.1109/BotSE.2019.00017.
[95] G. May, M. Taisch, A. Bettoni, O. Maghazei, A. Matarazzo, and B. Stahl, "A New Human-centric Factory Model," Procedia CIRP, vol. 26, pp. 103–108, 2015, doi: 10.1016/j.procir.2014.07.112.
[96] A. A. Nazarenko and L. M. Camarinha-Matos, "Towards collaborative Cyber-Physical Systems," in 2017 International Young Engineers Forum (YEF-ECE), 2017, pp. 12–17, doi: 10.1109/YEF-ECE.2017.7935633.
[97] E. G. Caldarola, G. E. Modoni, and M. Sacco, "A Knowledge-based Approach to Enhance the Workforce Skills and Competences Within the Industry 4.0," in 10th International Conference on Information, Process, and Knowledge Management, 2018, pp. 56–61.
[98] L. Kassner, P. Hirmer, M. Wieland, F. Steimle, J. Königsberger, and B. Mitschang, "The Social Factory: Connecting People, Machines and Data in Manufacturing for Context-Aware Exception Escalation," in 50th Hawaii International Conference on System Sciences, 2017, doi: 10.24251/HICSS.2017.202.
[99] A. Dersingh, P. Srisakulpinyo, S. Rakkarn, and P. Boonkanit, "Chatbot and Visual Management in Production Process," in ICEIC 2017 International Conference on Electronics, Information, and Communication, 2017, pp. 274–277.
[100] F. Longo, L. Nicoletti, and A. Padovano, "Smart operators in industry 4.0: A human-centered approach to enhance operators' capabilities and competencies within the new smart factory context," Comput. Ind. Eng., vol. 113, pp. 144–159, 2017, doi: 10.1016/j.cie.2017.09.016.
[101] X. Chen, M. Bojko, R. Riedel, K. C. Apostolakis, D. Zarpalas, and P. Daras, "Human-centred Adaptation and Task Distribution utilizing Levels of Automation," IFAC-PapersOnLine, vol. 51, no. 11, pp. 54–59, 2018, doi: 10.1016/j.ifacol.2018.08.234.

[102]U. Gnewuch, S. Morana, M. Adam, and A. Maedche, "Faster is Not Always Better: Understanding the Effect of Dynamic Response Delays in Human-Chatbot Interaction," in 26th European Conference on Information Systems, 2018.

[103]P. Zajec, J. M. Rožanec, I. Novalija, B. Fortuna, D. Mladenić, and K. Kenda, "Towards Active Learning Based Smart Assistant for Manufacturing," in Advances in Production Management Systems. Artificial Intelligence for Sustainable and Resilient Production Systems, 2021, pp. 295–302, doi: 10.1007/978-3-030-85910-7_31.

[104]A. Bousdekis et al., "Human-AI Collaboration in Quality Control with Augmented Manufacturing Analytics," in Advances in Production Management Systems. Artificial Intelligence for Sustainable and Resilient Production Systems, 2021, pp. 303–310, doi: 10.1007/978-3-030-85910-7_32.

[105]Coala, "About COALA," 2023. [Online]. Available: https://www.coala-h2020.eu/. [Accessed: 22-Feb-2023].

[106]Q. Su, H. Wang, C. Sun, B. Li, and J. Li, "Cyber-attacks against cyber-physical power systems security: State estimation, attacks reconstruction and defense strategy," Appl. Math. Comput., vol. 413, p. 126639, 2022, doi: 10.1016/j.amc.2021.126639.

[107]Kaspersky, "Chatbots are everywhere, but do they pose privacy concerns?," 2023. [Online]. Available: https://www.kaspersky.com/resource-center/preemptive-safety/chatbots. [Accessed: 22-Feb-2023].

[108]S. P. Zambiasi and R. J. Rabelo, "Virtualization of Manufacturing Operators: An Agent-bot Model," in 8th Brazilian Symposium on Intelligent Automation, 2007, pp. 1–6.

[109]A. Bousdekis, G. Mentzas, D. Apostolou, and S. Wellsandt, "Evaluation of AI-Based Digital Assistants in Smart Manufacturing," in Advances in Production Management Systems. Smart Manufacturing and Logistics Systems: Turning Ideas into Action, 2022, pp. 503–510, doi: 10.1007/978-3-031-16411-8_58.

[110]T. Mettler, "Maturity assessment models: a design science research approach," Int. J. Soc. Syst. Sci., vol. 3, no. 1–2, pp. 81–98, Jan. 2011, doi: 10.1504/IJSSS.2011.038934.

[111]B. Pochiraju and S. Seshadri, Essentials of Business Analytics: An Introduction to the Methodology and its Applications. Springer, Cham, 2019.

[112]R. J. Rabelo, D. Romero, S. P. Zambiasi, and L. C. Magalhães, "When Softbots Meet Digital Twins: Towards Supporting the Cognitive Operator 4.0," in Advances in Production Management Systems. Artificial Intelligence for Sustainable and Resilient Production Systems, 2021, pp. 37–47, doi: 10.1007/978-3-030-85914-5_5.

[113]D. Romero and J. Stahre, "Towards The Resilient Operator 5.0: The Future of Work in Smart Resilient Manufacturing Systems," Procedia CIRP, vol. 104, pp. 1089–1094, 2021, doi: 10.1016/j.procir.2021.11.183.

[114]L. P. Zambiasi, R. J. Rabelo, S. P. Zambiasi, and R. Lizot, "Supporting Resilient Operator 5.0: An Augmented Softbot Approach," in Advances in Production Management Systems. Smart Manufacturing and Logistics Systems: Turning Ideas into Action, 2022, pp. 494–502, doi: 10.1007/978-3-031-16411-8_57.

[115]World Economic Forum, "Global Lighthouse Network: Insights from the Forefront of the Fourth Industrial Revolution," Geneva, Switzerland, 2019.

[116]Qlik Insight Bot, "Reach far more people with the power of your data," 2022. [Online]. Available: https://www.qlik.com/us/products/qlik-sense/conversational-analytics. [Accessed: 30-Jul-2022].

[117]R. O. Bezerra, R. J. Rabelo, and M. H. Cancian, "Supporting SOA Resilience in Virtual Enterprises," in Collaborative Networks of Cognitive Systems, 2018, pp. 111–123, doi: 10.1007/978-3-319-99127-6_10.