Plug, Plan and Produce as Enabler for easy Workcell Setup and Collaborative Robot Programming in Smart Factories

Michael Wojtynek[1] · Jochen Jakob Steil[2] · Sebastian Wrede[1]

Abstract The transformation of today's manufacturing lines into truly adaptive systems facilitating individualized mass production requires new approaches for the efficient integration, configuration and control of robotics and automation components. Recently, various types of Plug-and-Produce architectures were proposed that support the discovery, integration and configuration of field devices, automation equipment or industrial robots during commissioning or even operation of manufacturing systems. However, in many of these approaches, the configuration possibilities are limited, which is a particular problem if robots operate in dynamic environments with constrained workspaces and exchangeable automation components as typically required for flexible manufacturing processes.

In this article, we introduce an extended Plug-and-Produce concept based on dynamic motion planning, co-simulation and a collaborative human-robot interaction scheme that facilitates the quick adaptation of robotics behaviors in the context of a modular production system. To confirm our hypothesis on the efficiency and usability of this concept, we carried out a feasibility study where participants performed a flexible workcell setup. The results indicate that the assistance and features for planning effectively support the users in tasks of different complexity and that a quick adaption is indeed possible. Based on our observations, we identify further research challenges in the context of Plug, Plan and Produce applied to smart manufacturing.

[1]Research Institute for Cognition and Robotics (CoR-Lab), Bielefeld University, Universitätsstr. 25, 33615 Bielefeld, Germany, {mwojtynek, swrede}@cor-lab.uni-bielefeld.de
.

[2]Institute for Robotics and Process Control (IRP), Mühlenpfordtstr. 23, 38106 Braunschweig, Germany, jsteil@rob.cs.tu-bs.de

1 Introduction

As the potential driver for smart factories and Internet of Things (IoT) in the shopfloor [1], collaborative robots and interconnected production devices are key components to enable flexibility in the production. In small, medium and large scale companies the products became more and more customized and specialized to raise customer satisfaction and to add product variety. Collaborative robots and interconnected devices support workers and close automation gaps by being integrated swiftly and autonomously into the production system to perform a new task. In recent years there has been a significant amount of research on smart factories [2], [3] and the corresponding industrial architectures [4] with vertical integration [5] from shopfloor to the business level.

The vision is to transform a conventional production with isolated automatized applications into a smart and intelligent factory with a higher grade of automation and data exchange. A key factor for a smart factory is the ability to adapt as quickly as possible to new industrial tasks and requirements. In literature this ability is known as reconfigurability [6]. For a versatile reconfiguration of an existing production sequence, modular principles are introduced [7]. The peripheral equipment such as feeders, boxes and conveyor belts have to become parts of the modular concept. Embedded systems in the modular components are used for local computation and task execution, and the ability to connect with other devices of the factory.

In an automatized factory, physical handling systems are required to reach a high degree of flexibil-
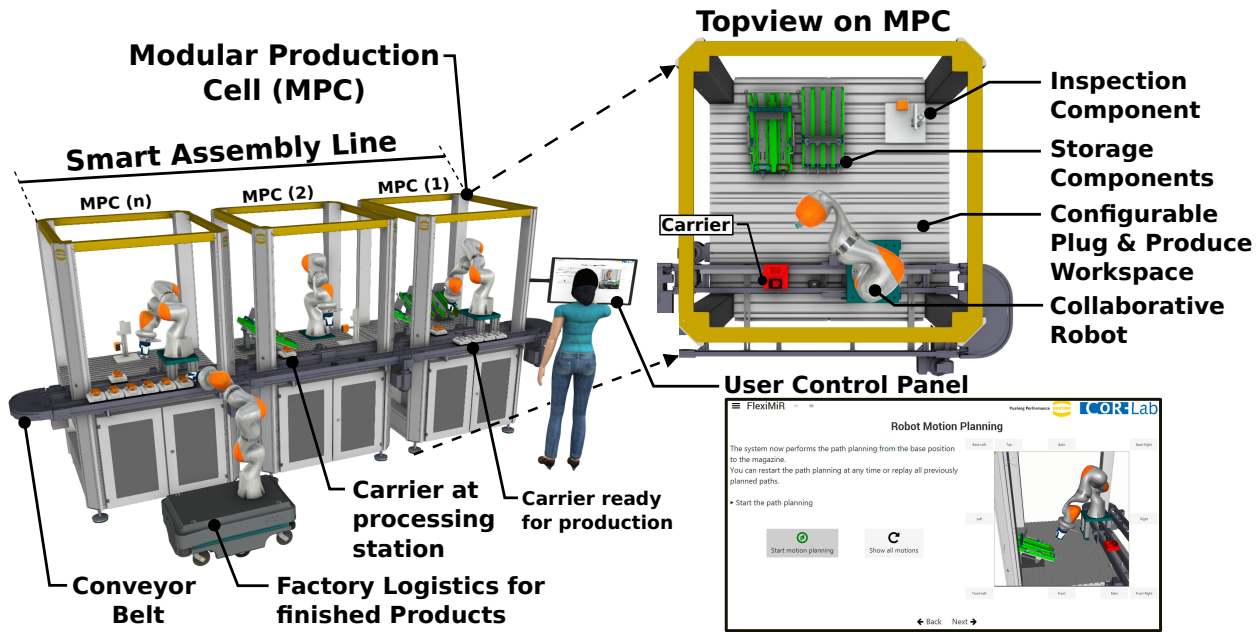
Fig. 1: A Smart Assembly Line consisting of Modular Production Systems equipped with collaborative robots and modular components for the flexible production.

ity to manipulate objects in the factory environment with purposes like pick-and-place operations or machine tending tasks. Collaborative robots (co-bots) are intended to relieve the factory worker from one-sided actions or hazardous tasks, and to work side-by-side with the human [8], [9]. An example is shown in Fig. 1.

The challenge is to combine collaborative robots with other components of the smart factory into one coherent system. The efforts to set up a smart and modular workcell and to program the robot for a certain application are enormous and expensive. At the same time, this process involves repetitive and tedious phases that often require a high grade of specialized knowledge. Thus, it is necessary to come up with new integration methods with standardized descriptions and reusable elements, and with recurring task schema to drastically reduce efforts and uncertainties during a robot's setup and its peripheral equipment for supporting the worker.

In this contribution, we investigate the question how an integrated system can assist the worker technically with situation aware, suited information and provide advice during the setup phase of an assembly workcell. Further, we investigate if it is possible to use human sensing and interaction for effective and precise robot programming and task parameterizing. To this end, we have developed an assistance system for setting up a workcell, which assists the worker with purposeful instructions. The combination of different modular peripheral devices is realized with standardized connectors via a Plug-and-Produce concept that allows an easy integration of components into the workcell.

The Plug-and-Produce method as part of a production presented in [10] is extended by an extensive Plan phase for the integration of robot relevant components. During the Plan phase the operator receives instructions on how to interact with a collaborative and lightweight robot, how to explore the environment and how to integrate peripheral equipment into the assembly and production flow. We evaluated the collaborative system in a user study concerning usability and user-friendliness to measure the system's ability to convey interactive instructions to the user. During the study, we collected data about the cognitive load of the participants to find out if we can improve the system towards the perceived workload of mental or physical stress.

In Section 2 we present our research goals and the requirements for a modular production workcell that is equipped with a collaborative robot. Further, we survey relevant literature and related work regarding the collaboration between humans and robots in industrial applications. Section 3 describes the system capabilities that we developed for an industry-oriented production process and introduces a lifecycle and an extended Plug-and-Produce concept to form a Plug, Plan and Produce scheme. We evaluate the presented system in Section 4 in a user study, and we discuss the results. Section 5 concludes our contribution and sketches an outlook for future research.

Requirement	Definition
R1	Easy setup of the system
R2	Quick adoption to new tasks
R3	Intuitive usage during runtime
R4	Modularity of components for flexibility in workspace
R5	Simulation as frontend for interaction and for internal simulation

Table 1: Requirements for the realization of a smart production system proposed in this contribution.

2 Research Questions and Goals

For the realization of a smart, easy and fast workcell setup combined with a collaborative robot several approaches have been investigated in recent years. The assumption is that Plug-and-Play solutions, which are denoted in the industrial sector as Plug-and-Produce (PnP) or Plug-and-Work (PnW), are the essential enablers for a smart production [11], [12], [13]. PnP and PnW is the way to integrate and interconnect automated and peripheral components easily and quickly with a smart and modular workcell. Flexible robot workcells that use PnP for and during the workcell setup are focused in Hennecke et al. [14] and Pfrommer et al. [15] have shown how to utilize PnP for service-orientated architectures. Requirements for a human-centered design in modular production systems were published in [16].

Using the Plug-and-Produce principles for the mechanical setup of the robotic workcell takes the operator in charge to configure the physical layout. The robot motion programming is based on the workcell layout, the position of the peripheral components and the task to be solved. The operator is responsible for the final validation of the physical behavior, in particular for the robot motions, and the approval of the setup. Thus, it is significant to analyze the interaction between the system and the operator. We are investigating the human-machine interface with respect to the following research questions:

1. Is the human capable of interpreting interactive working instructions to perform a complex robot-based machinery setup along the Plug, Plan and Produce Phases? (RQ1)
2. Is it possible to use human's sensing and interaction for effective robot programming? (RQ2)

The research community has worked and published exciting content answering these research questions and providing intuitive programming methods addressing co-bots in smart workcells. Steinmetz et al. [17] introduced RAZER, a GUI based framework for task-level programming of robots. The framework addresses robot experts as well as shopfloor operators and provides different access levels for the users. Robot experts are responsible for defining and creating robot skills (e.g. for screwing, drilling, assembling), which are used and filled with parameters by shopfloor workers. The RAZER framework was evaluated in a user study that can be compared in some regards, such as the IsoNorm results, with the study presented in Sec. 4.

CoSTAR is a task editor for high-level robot programming that is based on Behavior Trees [18], [19]. Behavior Trees are a formalism to design and structure robot tasks in a hierarchical order. The goal of CoSTAR is to allow users a natural way to create an elaborate task plan. He/she is supported visually with a user interface and by integrated features like a waypoint manager that guides the user with a mixture of demonstrations and explicit instructions to compose quickly complex task plans.

ArtiMinds is a patented Programming Suite that combines online and offline programming on various hardware platforms [20], [21]. Programming is performed visually using action blocks, which are combined into task sequences via drag and drop operations. A wizard for online robot teaching and setting parameters supports the user during the integration step. In contrast to conventional robot programming in the shopfloor with the teach pendant, ArtiMinds provides 3D-Visualization of the environment, collision checking and simulation of the programmed motions.

Further approaches on smart production include vision systems for observing the worker and interpreting gestures to control the robot with play or stop commands [22]. Also Wein et al. have presented a camera-based system for object recognition in the workspace. The peripheral components in the workspace are tagged with two-dimensional data codes used as markers to achieve a free positioning and recognition of the modules. The geometrical data is used to perform an offline robot path planning in simulation and to support execution in the real world [23].

In smart factories, intuitive and easy robot programming is required for a fast and efficient workcell setup that doesn't overwhelm or hinder the user from work. Rossano et al. [24] have categorized industrial and collaborative robot programming into main groups like: (1) Flow-Based Programming for configuration of data flow diagrams with functional blocks, and (2) CAD-Based Programming, where the user imports CAD data and generates manually, semi-automatically or automatically a robot program. Further, (3) Wizards-Based Programming was introduced that guides the user through a wizard to create a robot program, and (4) Lead-Through Programming lets the user perform

manually movements of the robot, which will be stored and replayed later.

Requirements for a highly automated and integrated system have been identified by Furman et al. in [25]: The setup of the system (R1), the adoption to new tasks (R2) and usage during runtime (R3) must be as simple as possible (R1-R3) for the machine setters to compete in terms of flexibility and re-usability with non-automated systems. Regarding the role of the human, Furman et al. have stated clearly that a machine setter cares exclusively about the visible and physical system and doesn't care about the wiring of physical devices or deploying of software to the control logic.

Further, modularity is a crucial concept for smart factories (R4), where the unpredictable real world is organized in distinct and highly independent modules that take over particular tasks. It is required that the modules can be combined easily to create a complete working system, where the links between the modules are established by the modules autonomously.

Sauer et al. [26] have identified another requirement of a smart factory relating to the interface between the human, machine and robot. According to this contribution, a simulation is utilized as a frontend for interaction with the operator and a built-in core is used for real-time simulation that enables fast reaction in unforeseen situations (R5). These essential requirements are summarized in Tab. 1.

The identified, single requirements (R1-R5) have been distilled from the related work to compose a collaborative system, which assists the user in the configuration of an autonomous robotic production cell. The Plug-and-Produce concept aims to realize an easy setup of the robotic workcell (R1). A localization method for easy adoption to new tasks and a setup lifecycle for usage during runtime are introduced to meet the requirements (R2, R3). The presented Plug-and-Produce system is of a modular and standardized shape (R4) and the user interface embeds a simulation for computing and validating robot paths, which are simulated before the execution on the real robot (R5).

3 Plug, Plan and Produce

The intention is to build a system with high diversity in solving industrial tasks and thus the Plug, Plan and Produce concept with the self-description of components and the interaction lifecycle is introduced in this section. A smart workcell combined with a collaborative robot is supposed to be used in a wide range of fabrication processes such as pick-and-place of diverse parts, machine tending, and assemblies. Typical robotic tasks, which can be found as part of these processes, are peg-in-hole operations, screwing, gluing, and challenging assemblies with snapping pieces.

The human's interaction with the modular and smart workcell-system has to be organized with self-explaining elements to ease the use of the robotic system for the operator (R1). The user interface on the control panel assists the operator with known and recurring elements to perform his or her task quicker and with more confidence (R2). Thus, we are proposing a scheme (in Fig. 2) consisting of three phases such as the Plug phase for connecting components to the workcell, the Plan phase to integrate and localize components and to calculate robot motions and collision-free paths, and finally the phase for Produce, once the planned robot motions and workcell behavior is executed automatically without human presence. To realize such a concept following the principles of a smart production, where Plug-and-Produce mechanisms are used to interconnect components in hard- and software, the workcell and its tools are designed in a modular way (R4).

At least two human roles are involved in the configuration process. A production or robot engineer defines in advance process plans for the entire production down to the component's services and functionalities. The workcell operator gets access to these functionalities and performs the integration process. Modularity in the process plans allows a high degree of reuse and outer parametrization like it is commonly used in skill architectures [27], [28], [29].

3.1 Self-Description used for the Plug, Plan, and Produce Process

As shown in Fig. 2 the first step of the integration process starts with the manual plug-in of the component connector in a socket of the workcell. Each component is equipped with an Identification Module (ID-Module) containing a Management Shell to organize communication endpoints and to transmit the self-description of the hardware component to the workcell. For the self-description, the Automation Markup Language (AML) is used to represent the model of the component's structure, its functional role, and geometrical information.

The structure of the AML model is presented in Fig. 3 and consists primarily of the *UnitClass*, *RoleClass* and *InterfaceClass*. The *UnitClass* represents a concrete class of a component that is instantiated for a distinct production or assembly process when the connector is plugged-in. A storage for items or an inspection camera might be a *UnitClass*, for example.

The *RoleClass* expresses the role of the component to solve a task, which could be *ProofQuality* in case of the camera or *ProvidedItem* in case of the storage. The *InterfaceClass* includes the *DeviceConnector*, the
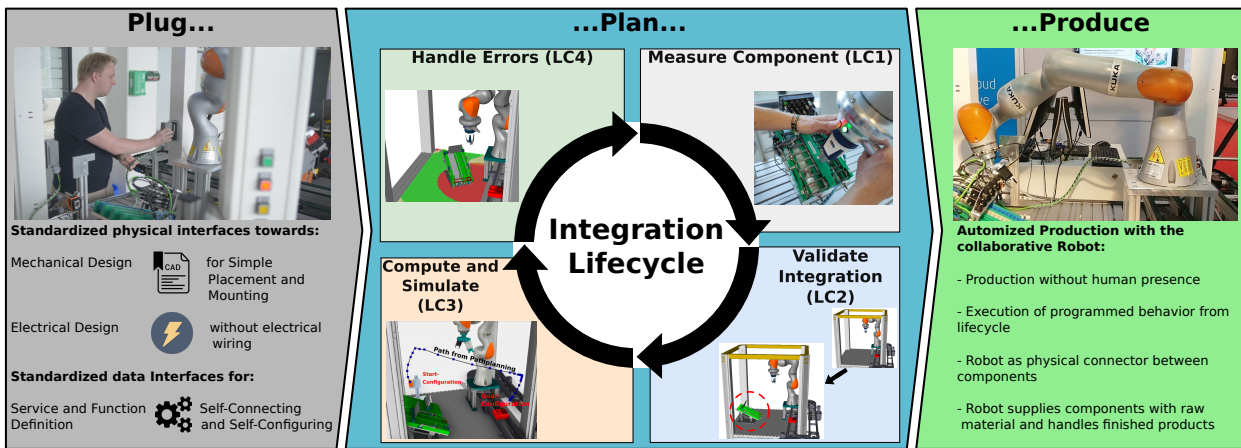
Fig. 2: Plug, Plan and Produce — Plug phase for connecting a component to the workcell, the Plan phase for localization and dynamic motion planning, and the Produce phase for autonomous manufacturing.

ProcessConnector, an interface for CAD-models and a *FrameProvider* for organizing working points. The *DeviceConnector* resolves communication endpoints for automatic discovery purposes while the *ProcessConnector* provides services and functions that can be utilized and triggered by a process engine. CAD-models used in the planning phase (Sec. 3.2) are referenced by the according interface and provide the visual and collision models of components and the environment. These CAD data are used by a path planner to generate collision-free robot motions, which are simulated and shown to the operator.

The *FrameProvider* in the AML model facilitates the dynamic placement of components in the workspace during the initial setup and reconfiguration of the workcell. A frame describes a point in position and orientation with respect to a reference coordinate system. The robot aligns the axis of its gripper frame (Fig. 3, $F_{Gripper}$) with a goal frame for approaching and picking an object ($F_{Approach}$, F_{Pick}). Component-dependent task frames like the *Base-, Approach-, and Pickframe*

Fig. 3: Model for Self-Description of components using the AutomationML notation.

are stored as working points in the self-description. The *BaseFrame* is defined from the results of the localization process (described in Sec. 3.2, LC1) and the first calibration point sets its origin. If the component is initially located or moved during the integration process, the *BaseFrame* and the transformation for the related frames (*ApproachFrame*, *PickFrame*) are updated and the component changes its position and orientation internally in the AML model and externally visible for the operator in the simulation.

3.2 Plan Phase: The integration lifecycle

A key part for the integration of Plug-and-Produce components is the human interaction with the smart and robotic-based production system. Therefore, we have designed a model for the human-machine interaction organized as a lifecycle. This integration and interaction lifecycle is shown in Fig. 2 and consists of the following sequential parts: Measure Components (LC1), Validate Integration (LC2), Compute and Simulate (LC3) and Handle Errors (LC4).

To start the integration process, the user interface instructs the operator to plug-in the connector of a particular component into a socket of the workcell to establish the physical connection (see Sec. 3.3). Successful powering and data exchange between the newly connected component and workcell doesn't allow an immediate start of an assembly or production task, because the component's position and orientation is undefined in reference to the workspace.

The system is designed and used without image processing or sensor feedback to avoid possible inaccuracy, complexity, and noise when localizing components. Hence, the operator uses the collaborative robot as an advanced measuring tool (Fig. 5) and guides the end-effector of the robot to specific calibration points
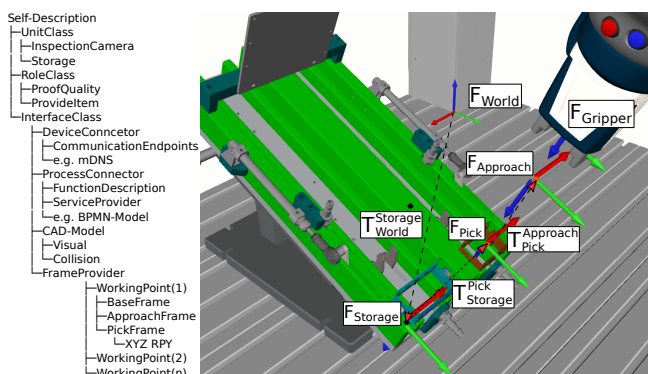
(LC1). The user control panel instructs the operator, how to insert a measuring tool for this task and how to perform the measuring. The system benefits from the manual, cognitive, and spatial capabilities of the human that are used to localize components and to guide the robot manipulator through the unknown workspace.

The result of the measuring procedure and the spatial integration of the component have to be validated by the operator (LC2). Therefore, the user dialog embeds a simulation and visualization of the virtual robot and the static surroundings. In the case of successful localization, the component is added dynamically to the virtual representation of the simulation considering the position and orientation in the real world. The operator checks and confirms visually the proper placement of components in the simulation taken from the real world. The measurement routine must be repeated if the component was misplaced in the simulation resulting from the integration process in LC2. During the production, small misplacement is compensated by compliant robot motions that tolerate minor deviations in the position. So far the system is limited that larger deviations can only be detected if the process is executed with reduced speed during a test mode.

The correct placement of the component in the real world and consequently in the internal representation is the prerequisite for computing and simulating collision-free robot motions for the production sequence (LC3). The robot path planning is computed internally based on the CAD-model of the static robotic workcell and the dynamically integrated components. For verification purposes by the operator, the resulting robot trajectories are visualized interactively in the embedded simulation. If the path planning algorithm can't calculate appropriate and collision-free paths, the user has to go through an error handling routine (LC4). In this routine, the system uses a component-dependent and precomputed heatmap to indicate the user appropriate positions for the components. A new iteration of the lifecycle begins, when the component has to be relocated in the workspace. Once all components are successfully placed, the integration is finished, and the production is ready to start.

3.3 Design and Function of Plug-and-Produce Components

The idea behind Plug-and-Produce is to integrate modular components as easily and quickly as possible into the industrial production of goods. An exemplary component is presented in Fig. 4 with an explicit component structure regarding the mechanical and electrical design as well as to the physical and data interfaces.
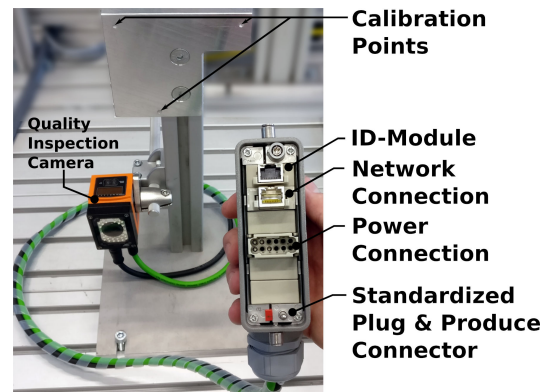

Fig. 4: Plug-and-Produce component with a unified hardware connector.

Each component is equipped with a unified connector that enables the wired connection with the modular production cell at any available socket. The connector uses standardized plugs for power transmission and wired networking, as well as an ID-Module for a unique identification of the hardware. The ID-Module contains a self-description of the component, which consists of geometrical data that describes the shape of the component (e.g. boundary box or CAD-model) and the working points for the robot (e.g. pick or place point). Further, the self-description contains a service layer that provides the component's functionalities, which can be used and controlled by workcell or the robot. The ID-Module was realized with an ARM-based embedded system, which deploys a runtime container for the Plug-and-Produce software services. A discovery service detects and registers available components that have been plugged-in and keeps track of their provided services.

3.4 The Modular Production System in Detail

Three modular production cells combined to an exemplary assembly line are depicted in Fig. 1. Each cell consists of the following parts: a modular production cell, a collaborative lightweight robot, components such as storages, a user control panel for interaction, and a carrier for transporting goods. The human operator receives instructions from the user control panel and configures the workcell according to the requirements of the process order. He/she is guided, where to place components in the workspace, how to put them into operation and how to integrate them into the robot process.

Considering the software, the core component of an automated, robot-based production is the process engine that receives upcoming tasks or outstanding assembly orders. It is responsible for the interpretation as well as the execution of predefined production plans. These production plans are represented in our system as BPMN2.0 process models (Business Process Model and

Notation). A robot plugin, as part of the process engine, integrates the robot into the running process instance and sets up communication ports as needed for further robot activities. For example, a port might connect the process engine with the robot controller to transmit a start and goal frame and to execute a motion command. The necessary frames are programmed interactively using kinesthetic teaching before a new production plan is executed. Detailed work on the architecture and the technology of the Plug-and-Produce concept was published in [30].

Furthermore, a path planner for the robot motion behavior is provided by the simulation plugin, which may be triggered by the process engine during the user-orientated lifecycle processes in LC3. In the presented system, the Open Motion Planning Library [31] with the RRTConnect planner was integrated. The path planner computes collision-free trajectories and stores these inside the representation of an environment model, which maintains the frames (position and orientation) of the newly integrated components. For assembling a product, the process engine executes activities from the production plan using the simulated and precomputed trajectories derived from the environment model for performing the collision-free robot motions in the recently (re)arranged real world environment.

Together the core services address the identification and integration of components into the modular production system to perform localization and dynamic path planning using a collaborative robot, human input and an environment representation combined with a simulation.

4 User Study

This section describes the user study with the assisted and collaborative robotic system and presents the results of $n=17$ subjects. The user study is divided into three main parts, which have to be completed by the participants: (I) A *Tutorial Phase* to learn basic functions and to gain proficiency for handling the smart robotic workcell, (II) *Hands-On Phase* of two setup scenarios with increasing difficulty focusing on the interaction with the robot and the Plug, Plan, and Produce system, (III) *Questionnaire Phase* to quantify the user experience and the satisfaction.

The user study is carried out with participants from the university spectrum and laboratory co-workers with technical background and average robotics knowledge. However, the system usage, functionalities and features are completely unknown for all participants. The tasks, which have to be solved by study participants, are pick-and-place operations after a successful setup of the
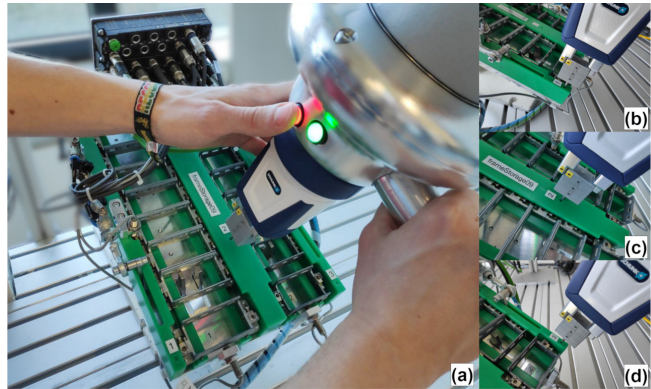

Fig. 5: (a) Collaborative measuring process with an inserted measuring tool at one calibration point. (b)-(d) the total calibration process of points 1-3.

modular workspace. We are simulating typical situations in small batched manufacturing like the quick integration, initial teaching of positions and the reconfiguration of components. Considering the research questions RQ1 and RQ2, we explore in this user study, how difficult it is to set up a workcell, how the user experiences to receive instructions from the system and how this affects the mental workload. A detailed video documentation on the user study is available at [32].

4.1 Phases, Tasks and Conduction

The *Tutorial Phase* provides basic descriptions of the robotic workcell, parts and components. The user control panel introduces general functions using textual descriptions with situation-suited images, animated and simulated robot movements, and color- and function-coded buttons at the robot gripper. In general, the operator is taught specific and collaborative interactions methods, how to plug-in a component's connector, how to guide and move the robot, and how to insert the measuring tool. A further part of the tutorial is a simple path planning in the embedded simulation followed by the motion execution on the real robot. Completing this phase ensures that all participants have a comparable level of knowledge about the workcell, robot and the Plug-and-Produce mechanisms before solving two pick-and-place tasks in the *Hands-On Phase*.

Scenario 1 (S1): The participant performs a full integration process of a storage component containing the Plug phase for connecting the component with the workcell, the Plan phase for component localizing, followed by the Produce phase, which moves the real robot along a collision-free path through the constraint workspace. The task goal is to perform a pick-and-place operation, where the robot grasps a workpiece from the storage and places it on the carrier (Fig. 6a, dashed line). S1 is simplified for the participant in such a way

that the integration is limited to only one component, which is fixed beforehand by the study conductors. The participant gets used to the handling of the Plug, Plan and Produce system before being confronted to place a component freely in the workspace.

Scenario 2 (S2): The components that have to be integrated are a storage component followed by a camera for quality inspection. The participant must place, screw and connect the components as part of the integration process. In this task, the camera is used to design the robot handguiding through the workspace and the resulting motions more challenging. In this extended and more difficult task, the robot has to pick first an object from the storage, show it into the lens of the quality camera, and finally place it on the carrier (Fig. 6a, continuous line). The user control panel indicates in the virtual workspace of the embedded simulation, where the component should be placed approximately. A colored rectangle in the simulation highlights the desired position and the participant decides by him or herself, where, how and with what kind of orientation the component has to be placed in the real workspace.

4.2 Methodology and Questionnaire

The user study aims to answer the research questions from the technical as well as from the user perspective view. Thus, we evaluate on the one hand the technical aspects, like the usability of the smart system, and on the other hand human factors, such as the experienced task load. For the quantification of the user experience we have used three questionnaires in total.

Subsequent to the *Hands-On Phase*, the participants answered two questionnaires, which measure user satisfaction for operating the system. The questionnaire IsoNorm ([33],[34]) measures the usability by asking the participants questions about Task Suitability, Self-Descriptiveness, Controllability, Conformity with User expectations, Error tolerance and Learning Suitability. The results of the questionnaire show the system performance in these different categories, which is a real advantage to identify specific categories with bad performance. Categories with performance below the average can be improved within the next development cycle.

Also the System Usability Scale (SUS) [35] has been used to collect general indicators about the user satisfaction. The SUS questionnaire uses the Likert-Scale to measure the usability and expresses it in one value for quick comparison with other systems. A suitable system reaches a score of 68% where 100% expresses the top score. The SUS value indicates on a high-level, if the system is ranked as user-friendly or if usability issues exist. In contrast to the IsoNorm questionnaire, SUS doesn't provide any information about the categories that can be improved through further development.

A questionnaire developed by NASA researchers to measure the Task Load Index (TLX, [36]) was used in its raw form as a tool to quantify the experienced cognitive workload by the participant. TLX measures dimensions like mental stress using an unknown system, the physical workload operating and moving a collaborative robot through the real workspace, and frustration level if a task hasn't worked out. The participants rated the experienced load after each practical phase of the study, and the result shows the progress of the workload during the different tasks.

4.3 Results

The user study shows interesting results regarding user-friendliness, usability and the perceived cognitive workload. The SUS questionnaire has resulted in an average score of 86.2%. Considering the value of 68%, where a system is already rated as usable, the reached score is pretty good. Even though, the SUS value reached a high score, a detailed view on the usability categories of the IsoNorm questionnaire, helps to improve the robotic and the Plug, Plan and Produce system.

The users rated the smart and robot-based system on a scale from -3 (worst) to +3 (best) answering the IsoNorm questionnaire and Fig. 6b shows the results. The *Task Suitability* category aims with questions on aspects of the user interface. A task suitable interface is intended to show only those information, which are related to solve the task effectively and efficiently. In this category, the resulting data is very dense with a positive median score of 2.4. This result shows that the developed system suits the task and fulfills the requirements for easy setup (R1) and intuitive usage (R3).

Controllability defines the ability to keep and influence the direction of the user interaction during the whole process and reaches from the very beginning of the task until the user has reached his or her goal. The result of *Controllability* of the IsoNorm questionnaire shows a large variance in the data from -0.8 to +3.0. The result is an indicator that some participants experienced the interaction with the system as stiff and rigid, whereas 50% have felt comfortable being guided through the system.

The result of the *Controllability* shows correlations to the *Self-Descriptiveness*. The self-descriptiveness is defined as each step on the user control panel being immediately comprehensible through feedback from the system. On the one hand, 50% of the participants rated the self-descriptiveness with high scores of 2 and above, and on the other hand, the remaining 50% rated the system with a negative tendency to -0.8.
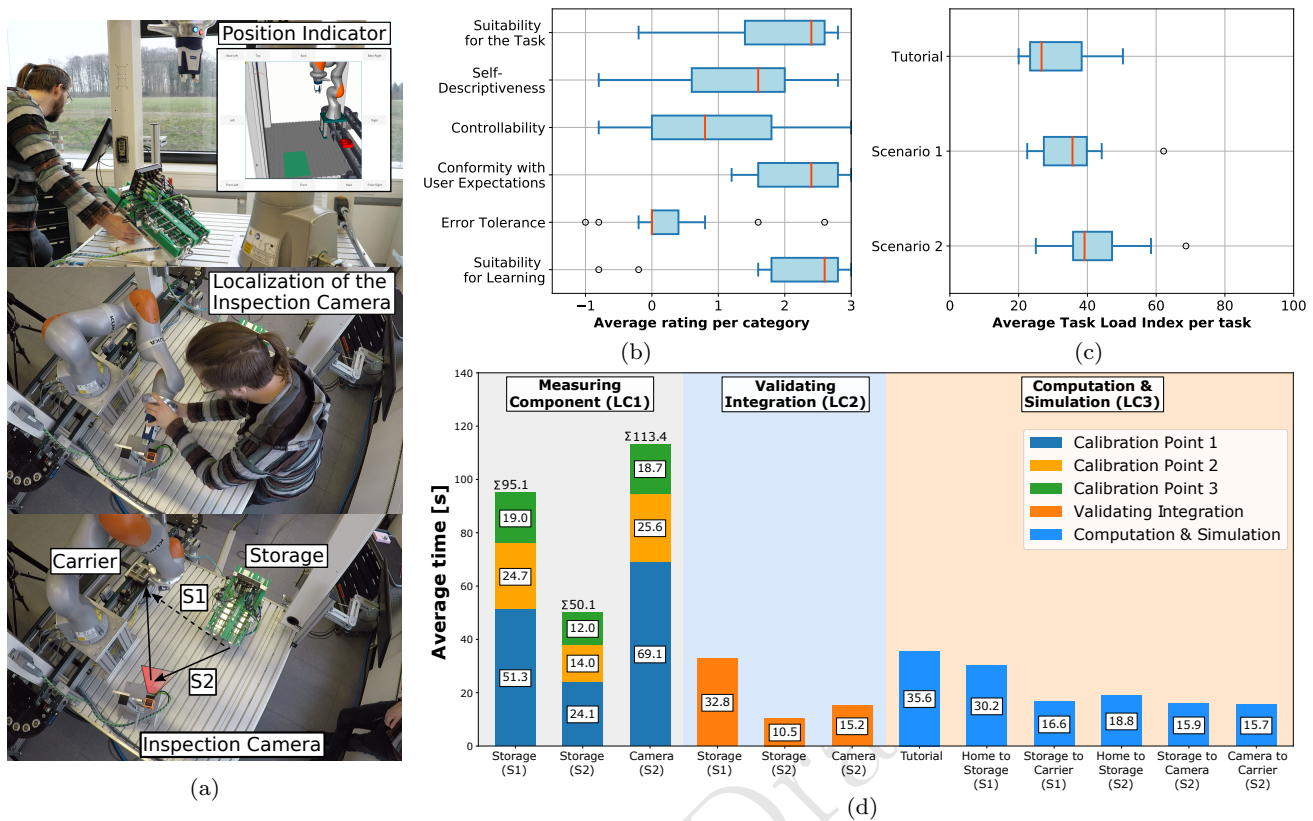
Fig. 6: (a) Interactions with the co-bot and smart system during the user study. (b) IsoNorm questionnaire results (c) Task Load Index. (d) Average time of all participants performing phases from the integration lifecycle.

Conformity with user expectations show low variance and a median at 2.4. This means that the instructions on user screen correspond to the user's task knowledge, experience and common conventions. The developed system has met the user expectations, and the requirements for quick adoption (R2) and intuitive usage during runtime (R3).

Error Tolerance is an important factor in a smart factory, where the desired result of the interaction can be achieved with no or minimal corrective action. 50% of the participants valued the proposed system with negative scores, whereas 50% have had only small positive experiences with the system in terms of error tolerance. Concerning the error tolerance, the proposed system of this contribution shows similar results like the RAZER system [17]. From the user perspective, it can be stated that error tolerance is emphasized by users utilizing a smart and modular robotic system. In contrast to this user demand, the development of such systems focuses much on the successful task performance, precise motion execution and robust runtime performance, whereas the error handling or tolerance plays a subordinate role in the development of such systems.

The minimization of the learning time is expressed in the category *Suitability for the learning* and addresses directly the requirement for quick adoption to

new tasks (R2), when users are guided through the learning stages. The results of the questionnaire show extremely positive results with a median around 2.6 and relative low variance compared to the other categories. This result can be explained with the detailed and verifiable Tutorial Phase at the beginning for introducing the tasks, components and the Plug, Plan and Produce concept. It indicates that an extensive and comprehensible introduction phase helps the operators using the robotic system, even if the learning curve is increasing, because of more challenging tasks.

The results of the task load questionnaire are shown in Fig. 6c and indicate that the index is increasing from the Tutorial Phase to Scenario 2. The study was designed in such a way that the participants gain increasing experience with the robotic system during the study, which makes it possible to estimate the difficulty of the tasks. The results show that the demands of participants during the *Tutorial Phase* are quite low with 26.7% on average, when the smart workcell, the involved parts and collaborative robot are explained. The task load raises about approximately 10% (35.7% total) after the first scenario was conducted. The participants are instructed to perform a plug-in of the storage connector and localize the storage. For the localization, the collaborative robot has to be moved, which resulted in

some physical and mental demands. Scenario 2 resulted in the highest task load index (39.2% on average). The participants have performed a full Plug, Plan and Produce task sequence, located two components, moved the collaborative robot in handguide mode, and performed finally the path planning for three motions in total to reach two components. A higher task load could be caused, because some users have faced problems with the joint limits of the robot. Hitting the joint limits makes it difficult to move the robot in the handguiding mode. A challenging robot pose during the localization of the inspection camera can be seen in 6a.

The *Average Time* over all study participants for the distinct lifecycle parts is denoted in Fig. 6d. The first third of the graph shows the total measuring time for the localization of a Plug-and-Produce component, which is broken down to each calibration point introduced in Fig. 5. The average localization time for the storage shows a reduction by 50% for calibration point 1 including the handguiding of the robot from a defined home position towards the component. Due to the shorter distance from the first calibration point to point 2 and 3, the average measuring time is lower. However, it shows a significant reduction between the fixed storage in Scenario 1 and the storage in Scenario 2, which was fixed by the participant. The increased time for localizing of the quality inspection camera arises from the difficulty in arranging the robot pose towards the calibration points (see Fig. 6a). It indicates that time for the localization is dependent on the robot pose and influenced by the hardware design and joint limits. The average time for the validation (LC2) shows a learning effect (Fig. 6d, second third) from component to component by reducing the time significantly and keeping a certain time level (approx. 16s) for the path planning and simulation (LC3) steps (Fig. 6d, last third). Performing the *Tutorial Phase* has taken each user in average 4.6 minutes (278.0s), *Scenario 1* 6.55 minutes (393.2s) and the more advanced setup integrating two components in *Scenario 2* 11.9 minutes (714.3s).

4.4 Discussion of the Results

The user study has shown interesting results towards the usability of modular production systems utilizing a collaborative robot combined with Plug-and-Produce components. The results, in particular the score of the IsoNorm questionnaire, indicate from categories such as *Task Suitability*, *Conformity with User Expectations*, *Suitability for Learning* that the study participants in the role as operators are capable to set up a complex automated and robot-based system (RQ1). The index for the cognitive task load demonstrates that the study participants were not overex-

tended physically and mentally during the scenarios and were able to work on the tasks appropriately.

The participants have performed robot programming and workcell setup by programming the robot in collaboration and physical interaction, which shows a positive trend on effectiveness considering the reduction of average time in the different phases of the lifecycle. Concerning RQ2, the average time measurement has shown that effective physical robot programming is possible even though the real world constraints, like joint limits, influence this parameter.

Even though the results of the user study were mostly positive, there is room for improvement of the proposed system in terms of *Controllability*, *Self-Descriptiveness* and *Error Tolerance*. A certain amount of users (>50%) were satisfied with the information on the screen and the behavior of the system in the real world considering these categories. However, the remaining participants valued the system with a negative trend in the rating. This leads to the assumption that within the participants several user groups exist. It can be suspected that a distinction between novel, intermediate and expert users could be helpful concerning usability and user-friendliness. Experienced users might need less information, assistance and controllability during the task, whereas a stiff system with reduced controllability might suit inexperienced users better. Future systems should be ready to be individualized according to the behavior of each user.

5 Conclusion

This contribution motivates an extended Plug-and-Produce approach for flexible manufacturing. The workcell setup with mandatory fabrication components is challenging and the adaptation of the required robot behavior is labor-intensive and time-consuming. Our concept extends existing Plug-and-Produce approaches with a lifecycle and structured Plan phase to perform the initial and robot-orientated workcell setup. The Plug, Plan and Produce approach was evaluated in a user study where the participants performed the setup process autonomously and executed an industrial pick-and-place operation on the real robot. The results have shown that the users were able to receive, interpret and execute the guidance from the interactive system.

Future work will focus on the enhancement of the setup process and the usability of the system, and on increasing the modeling precision for simulation processes. Flexible objects, such as cables, are hard to simulate and an obstacle for the robot. Virtual and augmented reality might help in the future to model and react to unpredictable objects, simulating the robot motion with overlaying information from the real world.

ACKNOWLEDGMENT

References

1. D. Slama, F. Puhlmann, J. Morrish, and R. Bhatnagar, *Enterprise IoT: Strategies and Best Practices for Connected Products and Services*. O'Reilly UK Ltd., 2015.

2. B. Bunse, H. Kagermann, and W. Wahlster, "The SmartFactoryKL Perspective," *Future Markets Industrie 4.0*, p. 26, 2014.

3. S. Scholz, A. Elkaseer, T. Müller, U. Gengenbach, and V. Hagenmeyer, "Smart modular reconfigurable fully-digital manufacturing system with a knowledge-based framework: example of a fabrication of microfluidic chips," 14th IEEE International Conference on Automation Science and Engineering (CASE), München, August 20-24, 2018, 2018, 43.22.03; LK 01.

4. B. Vogel-Heuser, G. Kegel, K. Bender, and K. Wucherer, "Global Information Architecture for Industrial Automation," *Automatisierungstechnische Praxis (atp)*, vol. 1, no. 51, pp. 108–115, 2009. [Online]. Available: http://ojs.di-verlag.de/index.php/atp_edition/article/viewFile/71/787

5. S. Wrede, O. Beyer, C. Dreyer, M. Wojtynek, and J. J. Steil, "Vertical Integration and Service Orchestration for Modular Production Systems using Business Process Models," 2016.

6. D. Schütz, M. Schraufstetter, J. Folmer, T. Gmeiner, and K. Shea, "Highly Reconfigurable Production Systems Controlled by Real-Time Agents," in *2011 IEEE 16th Conference on Emerging Technologies & Factory Automation (ETFA)*, 2011.

7. S. Weyer, M. Schmitt, M. Ohmer, and D. Gorecky, "Towards Industry 4.0 - Standardization as the crucial challenge for highly modular, multi-vendor production systems," *IFAC-PapersOnLine*, vol. 48, no. 3, pp. 579–584, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2405896315003821

8. R. Patel, M. Hedelind, and P. Lozan-Villegas, "Enabling robots in small-part assembly lines: The ROSETTA approach - an industrial perspective," *Proc. of German Conf. on Robotics (ROBOTIK)*, pp. 1–5, 2012.

9. B. Kuhlenkötter, A. Schyja, A. Hypki, and V. Miegel, "Robot Workcell Simulation with AutomationML Support-An Element of the CAx-Tool Chain in Industrial Automation," in *Robotics (ISR)*, 2010, pp. 1076–1082. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5756920

10. T. Arai, Y. Aiyama, M. Sugi, and J. Ota, "Holonic assembly system with plug and produce," *Computers in Industry*, vol. 46, no. 3, pp. 289–299, 2001.

11. S. Schriegel, J. Jasperneite, and O. Niggemann, "Plug and work für verteilte Echtzeitsysteme mit Zeitsynchronisation," in *Industrie 4.0 und Echtzeit*, W. A. Halang and H. Unger, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 11–20.

12. S. Schriegel and L. Wisniewski, "Investigation in automatic determination of time synchronization accuracy of ptp networks with the objective of plug-and-work," in *2014 IEEE International Symposium on Precision Clock Synchronization for Measurement, Control, and Communication (ISPCS)*, Sept 2014, pp. 20–25.

13. M. Schleipen, A. Lüder, O. Sauer, H. Flatt, and J. Jasperneite, "Requirements and concept for plug-and-work - adaptivity in the context of industry 4.0," vol. 63, Oct 2015, p. 801820.

14. A. Hennecke and M. Ruskowski, "Design of a flexible robot cell demonstrator based on CPPS concepts and technologies," in *2018 IEEE Industrial Cyber-Physical Systems (ICPS)*. IEEE, 2018, pp. 534–539.

15. J. Pfrommer, D. Stogl, K. Aleksandrov, S. E. Navarro, B. Hein, and J. Beyerer, "Plug & produce by modelling skills and service-oriented orchestration of reconfigurable manufacturing systems," *At-Automatisierungstechnik*, vol. 63, no. 10, pp. 790–800, 2015.

16. M. Birtel, F. Mohr, J. Hermann, P. Bertram, and M. Ruskowski, "Requirements for a human-centered condition monitoring in modular production environments," *IFAC-PapersOnLine*, vol. 51, no. 11, pp. 909–914, 2018.

17. F. Steinmetz, A. Wollschläger, and R. Weitschat, "RAZER - A HRI for Visual Task-Level Programming and Intuitive Skill Parameterization," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1362–1369, 2018.

18. C. Paxton, A. Hundt, F. Jonathan, K. Guerin, and G. D. Hager, "CoSTAR: Instructing collaborative robots with behavior trees and vision," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 564–571, 2017.

19. K. R. Guerin, C. Lea, C. Paxton, and G. D. Hager, "A Framework for End-User Instruction of a Robot Assistant for Manufacturing," *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6167–6174, 2015.

20. R. Jaekel and G. Dirschl, "Method and system for programming a robot," U.S. Patent 2017/0 190 052A1, July 6, 2017.

21. *ArtiMinds, Programming Suite*, (accessed on November 28, 2018), https://www.artiminds.com.

22. J. de Gea Fernández, D. Mronga, M. Günther, M. Wirkus, M. Schröer, S. Stiene, E. Kirchner, V. Bargsten, T. Bänziger, J. Teiwes, T. Krüger, and F. Kirchner, "iMRK: Demonstrator for Intelligent and Intuitive HumanRobot Collaboration in Industrial Manufacturing," *KI - Künstliche Intelligenz*, vol. 31, no. 2, pp. 203–207, 2017. [Online]. Available: http://link.springer.com/10.1007/s13218-016-0481-5

23. S. Wein, L. Wolff, A. Malik, S. Storms, and W. J. Herfs, "Concept for Automated Robot Programming Using Image Processing," 8th Congress of the German Academic Association for Production Technology, Aachen (Germany), 2018. Cham: Springer, Nov 2019, pp. 705–714. [Online]. Available: https://publications.rwth-aachen.de/record/751597

24. G. F. Rossano, C. Martinez, M. Hedelind, S. Murphy, and T. A. Fuhlbrigge, "Easy robot programming concepts: An industrial perspective," *2013 IEEE International Conference on Automation Science and Engineering (CASE)*, pp. 1119–1126, 2013.

25. K. Furmans and F. Schonung, "Plug-and-work material handling systems," in *11th IMHRC Proceedings (Milwau-*

kee, Wisconsin. USA), 2010. [Online]. Available: https://digitalcommons.georgiasouthern.edu/pmhr_2010/1

26. O. Sauer and M. Ebel, "Plug-and-work von Produktionsanlagen und übergeordneter Software," in *Aktuelle Trends in der Softwareforschung. Tagungsband zum Do-It.Software-Forschungstag 2007. Hrsg.: K. Haasis.* MFG-Stiftung Baden-Wrttemberg, Stuttgart, 2007.

27. M. R. Pedersen, L. Nalpantidis, R. S. Andersen, C. Schou, S. Bøgh, V. Krüger, and O. Madsen, "Robot skills for manufacturing: From concept to industrial deployment," *Robotics and Computer-Integrated Manufacturing*, vol. 37, pp. 282–291, 2016.

28. S. Bøgh, O. S. Nielsen, M. R. Pedersen, V. Krüger, and O. Madsen, "Does your robot have skills?" in *Proceedings of the 43rd International Symposium on Robotics*, vol. 6. Verlag, 2012.

29. C. Schou, R. S. Andersen, D. Chrysostomou, S. Bøgh, and O. Madsen, "Skill-based instruction of collaborative robots in industrial settings," *Robotics and Computer-Integrated Manufacturing*, vol. 53, pp. 72–80, 2018.

30. M. Wojtynek, H. Oestreich, O. Beyer, and S. Wrede, "Collaborative and Robot-Based Plug & Produce for Rapid Reconfiguration of Modular Production Systems," in *Proceedings of the 2017 IEEE SICE International Symposium on System Integration.* IEEE, 2017.

31. I. A. Şucan, M. Moll, and L. E. Kavraki, "The Open Motion Planning Library," *IEEE Robotics & Automation Magazine*, vol. 19, no. 4, pp. 72–82, December 2012, http://ompl.kavrakilab.org.

32. *Video: Plug, Plan and Produce for Collaborative Robot Programming in Smart Workcells*, (accessed on March 30, 2019), https://youtu.be/Mf5iPpiD2NE.

33. K. Figl, "ISONORM 9241/10 und Isometrics: Usability-Fragebögen im Vergleich." 01 2009, pp. 143–152.

34. G. Gediga, K.-C. Hamborg, and I. Düntsch, "The Iso-Metrics usability inventory: An operationalization of ISO 9241-10 supporting summative and formative evaluation of software systems," *Behaviour & Information Technology*, vol. 18, no. 3, pp. 151–164, 1999.

35. J. Brooke, "SUS-A quick and dirty usability scale." *Usability evaluation in industry.* CRC Press, June 1996, iSBN: 9780748404605. [Online]. Available: https://www.crcpress.com/product/isbn/9780748404605

36. S. G. Hart and L. E. Staveland, "Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research," in *Human Mental Workload*, ser. Advances in Psychology, P. A. Hancock and N. Meshkati, Eds. North-Holland, 1988, vol. 52, pp. 139 – 183. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0166411508623869