Integration of Human Actors in IoT and CPS Landscape

Doruk Sahinel[1], Cem Akpolat[2], O. Can Görür[1], and Fikret Sivrikaya[2]

[1]DAI Labor, Technische Universität Berlin, Berlin, Germany
[2]GT-ARC gemeinnützige GmbH, Berlin, Germany

***Abstract*— The Internet of Things (IoT) is envisioned to facilitate rich interactions among heterogeneous entities, ranging from simple sensing devices to complex robotic devices and from autonomous service agents to human actors. The complexity and multimodality of human actors require specific interfaces for their integration with IoT frameworks that provide suitable software architectures, data models, protocols, message types, and applications. This study focuses on the requirements and design approaches for integrating human actors in a Cyber-Physical System (CPS) within the application domain of Industry 4.0. After a systematic review and taxonomy of the related research literature, the design and implementation of a comprehensive human-integration framework is presented as part of a multi-agent IoT middleware called CHARIOT. Example applications that are developed to exploit the human integration capabilities of CHARIOT middleware are then presented, which extract data from human activities, enable multimodal interaction between humans and other IoT entities, and assist different human roles in a smart factory environment to satisfy the human-CPS integration requirements.**

***Index Terms*—IoT, human-CPS integration, augmented reality, wearables, human activity recognition**

I. Introduction

Cyber-physical systems (CPS) radically change the way how production systems are being considered, deconstructing rigid production organizations and promoting truly interactive systems with all production components cooperating with each other [1]. For human-CPS integration, proper interfaces must be developed based on the application domain, the type of operation to be performed by the human, and the kind of data to be exchanged between the human and the system. Colombo et al. [2] list seamless, service-based interaction with human actors as one of the critical challenges of industrial CPS.

CHARIOT *(a sCalable Holistic middleware Approach foR the Internet Of Things)* is a project that devises an IoT middleware integrating a number of heterogeneous devices and sensors, and enabling spontaneous interactions between the components within a smart factory environment. Designed on top of a multi-agent platform, the middleware contains a semantic data layer recognizing the device attributes and functionalities, a knowledge management layer introducing the intelligence through learning in the autonomous entities, and a service planning and orchestration component. As the human role lies at the core of CPS-based smart factories of the future [3], integration of humans to the IoT middleware is crucial in realizing and executing the Industry 4.0 (I4.0) scenarios. This human integration is achieved by wrapping software abstractions around the physical world of humans, defined as a "Cybering the Physical (CtP)" process in [4].

The main idea of CHARIOT for human-CPS integration is to provide software interfaces towards end users through a framework that allows developers to integrate human actors through well-defined and reusable interfaces. Creating interfaces between human operators and running applications through the IoT middleware gives the applications the opportunity to exploit the services of the middleware, such as collecting and analyzing data from a human operator or creating notification mechanisms among human actors and machines.

The rest of the paper is structured as follows: The requirements of integrating humans to an IoT middleware are listed in Section II, which also includes the summary of a systematic literature review. In Section III, human integration requirements are mapped to CHARIOT IoT middleware design, and the components developed to satisfy these requirements are explained together with the designed human interfaces. To demonstrate how the designed human interfaces are exploited in smart factory use cases, CHARIOT applications enabling human-CPS integration are presented in Section IV.

II. Human Integration Requirements

In this section, we list the requirements that have to be considered in designing a human-CPS interface to an IoT middleware with example design methods from the literature. Our focus is on listing the requirements that must be considered to build an interface between human actors and digital systems that makes relevant human data accessible to digital systems. Table I provides a compact summary of the available human-CPS integration patterns in the literature, based on the requirements identified in this section.

1) Ability to Understand Human Characteristics: The first requirement of integrating humans to an Industry 4.0 CPS is to understand the differences between humans and other components of the system. Sowe et al. [12] emphasize three characteristics of humans that need to be considered as follows:

Corresponding Author: doruk.sahinel@dai-labor.de

978-1-5386-4980-0/19/$31.00 ©2019 IEEE

Ref.	Understanding Human Characteristics	Support for Human Roles	Reusability of Human Interfaces	Context Awareness	Data Analysis & Collection	Managing Complexity	Mobile Device Support
[5]	✓	✓	–	✓	✓	✓	✓
[6]	✓	✓	–	✓	✓	–	✓
[7]	–	–	–	✓	✓	✓	–
[8]	–	–	–	✓	✓	✓	✓
[9]	✓	✓	–	✓	✓	–	–
[10]	–	✓	✓	✓	✓	–	–
[11]	–	✓	–	✓	✓	✓	✓

Table I: Analysis of How Human-CPS Integration Requirements are Covered in the Literature

1) Cognitive skills: Humans observe, process and react to present challenges with a mental capacity different than software entities.
2) Unpredictability: Humans do not perform the same task the same way every time. They may lose focus, or adapt to different conditions.
3) Dynamic motivation: People require incentives. Without motivation, a person may not perform a task even after agreeing to do so.

CPS vision of I4.0 includes collaborative interaction of human actors with the machines in close vicinity. In traditional work environments, humans are not allowed to enter a robotic area during the operation for safety. However, I4.0 aims at lifting such restrictions of the workspaces while ensuring the safety of the human workers [13], [14]. For this purpose, a CPS should be aware of the human actors, with their personal and social contexts, and adapt itself accordingly in order to ensure safe and efficient collaboration with humans [3], [14]–[17].

An example for defining human attributes in a smart factory environment can be found in [18]. The authors model human workers with their workload preferences, which leads to more efficient work allocations. Similarly, in our previous work we show that a robotic system incorporating a human worker's changing availability, motivation and capability into its decision-making process contributes positively to the efficiency of the human-robot collaboration in a production process [17]. Another example is given in [6], where a virtual manikin (vMan) module is developed to provide a simplified manikin skeleton with eight body joints and is used in an intuitive process modeling interface that allows engineers to simulate operator processes.

2) Support for Different Human Roles: An important requirement of human-CPS integration in a smart factory is to describe the human stakeholders in the environment together with their relationship with machines, computers, and other humans. This description should involve the capabilities of humans so that the CPS is able to understand human actions and be able to choose the right person for a given task. The main human roles in a smart factory can be summarized as follows [19]:

- *Engineers* develop and design the production system. Collaboration between different engineering groups and consistent data flow through all the different engineering tools are required in the design phase. Collaboration with other human roles in the production process such as operators and technicians might also be needed.
- *Operators and supervisors* have observational tasks such as monitoring and supervising the production. Human operators aim to manage tasks within due dates, and to keep the CPS stable. They have to make quick and efficient decisions while they are facing complex systems [20]. Their task also involves interacting with other human roles.
- *Technicians and workers* keep the production running by actively participating in production tasks. For evaluation of the increasing amount of data in CPS, the technician has to be supported by new user interface concepts that enable the middleware to guide the technician by providing information access in unforeseen situations, or by getting the daily working plan from the system.

Some examples of human model definition in the literature are as follows: A fault tracker application that provides measurements and datasheets for engineers and technicians is given in [6], and an assistance system providing system context for supervisor role is presented in [9].

3) Reusability of Human Integration Interfaces: In human-CPS integration context, reusability of interfaces means that the software modules and interfaces developed for human integration should also have the property of being adaptable to different contexts, human roles, and applications. Use of ontologies for describing human roles and CPS features is an example method for designing reusable systems. An example of forming a knowledge base with semantic descriptions can be found in [10], where the semantic descriptions contain smart factory human roles that can be recognized by other entities, thus enabling the interactions among them.

4) Context Awareness: When human integration to CPS is interpreted as understanding human behavior and reacting to this behavior, then context awareness should become part of the human-CPS integration. A CPS may benefit from this in many ways. It leads to an increased human-awareness including the awareness of systemic errors a human can make given the task and a human's task-related expertise [21] and the awareness of a human's safety given the context of the work environment [14]. Additionally, the context-dependent selection of appropriate tasks for humans provides a more efficient work allocation [18]. This process should consider contextual knowledge such as a human actor's frequently visited location, personal interests, work schedule, expertise, etc. together with the past actions of the user for task assignment.

Context awareness is used in Sophos-MS [5] framework, in which an intelligent personal digital assistant with vocal interaction capabilities to enhance the operator's ability to perceive and act within a smart factory. In another example, a human worker's actions are recognized in the context, and her capability and motivation for a task are interpreted to redistribute a task either to a robot or to the human [17].

5) Data Collection and Analysis: Data collection, data analysis, and data representation methods are required to extract intelligence from raw human data. In addition, the overwhelming amount of data collected from other IoT components should be presented to the human actors with assistance systems like information screens and simulations [20]. Examples of such assistance systems are provided in [6] and [9]. An important enabler of data collection and analysis in the smart factory domain is the cloud and edge computing technologies, examples of which can be found in [5], [10], and [11].

An important aspect of human data collection and analysis is the ethical considerations. A CPS should carefully select sensors for human data, develop secure methods to manage data acquisition, and consider human privacy toward the ethical usage, storage, and transfer of data [22]. Human actors should have the right to decide how and to what extent their collected data will be used and shared by the CPSs.

6) Managing Complexity: CPSs are typically composed of highly heterogeneous sets of components (computers, agents, devices, etc.), which can become part of a production process at any time [20]. Integrating human tasks to this heterogeneous environment and introducing the interactions between humans and all other artificial agents contribute to the complexity of a CPS. To manage this complexity, methods to achieve a certain level of abstraction are developed, e.g., in [7] and [8].

7) Multimodal Interaction through Mobile Devices: Various types of mobile devices offer the possibility for human actors to interact with the environment by making use of the applications running on them. Wearables, smartwatches, and augmented reality and virtual reality headsets are all options that can be exploited for human-CPS integration. The constraints of mobile devices resulting from the limited screen size, battery life, etc. must be taken into consideration for information selection and rendering [19]. Interfaces that support human integration with mobile devices are available in, e.g., [5], [6], [8], and [11].

III. Human Integration in CHARIOT IoT Middleware

A. Agent-Based IoT Middleware Architecture

To manage the complexity of human-CPS integration in an IoT environment, the interactions between IoT services, human actors and other artificial agents need to be defined in a distributed and abstracted way. In our project, a service-oriented multi-agent system enables distributed management by providing communication mechanisms among agents that are abstracted from the entities that they represent. Agents represent IoT services, devices and human actors as software entities that are semantically described using OWL-S [23]. The communication between an agent and the IoT device it represents occurs with MQTT protocol, and through a run-time environment that the device is connected. Furthermore, communication protocols such as CoAP, MQTT, WebSocket, and HTTP are implemented between the device and the runtime environment. This multi-agent environment allows human actors to exchange and derive knowledge from other CPS components connected to the IoT middleware.

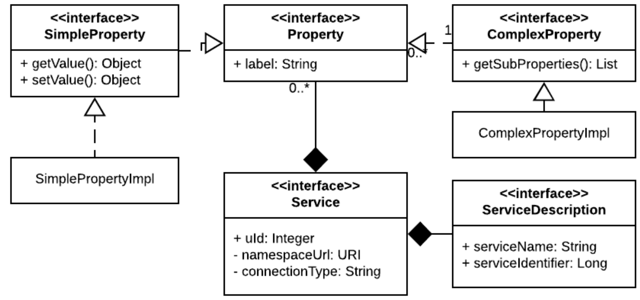

Figure 1: IoT Service Class Diagram

Human actors interact with other IoT components by making use of the IoT services connected to the middleware. For example, a notification mechanism is defined as an IoT service, and the functionalities of this service are used both by humans and robots in a human-robot collaboration use-case. A unified data model is constructed to abstract these smart factory services as agents. As shown in Figure 1, the data model of IoT services contains properties, functionalities, and communication protocols. Semantic annotation is achieved by defining service functionalities as agent actions that are stored with an OWL-S path.

A distributed service directory, which is a service description database in CHARIOT middleware [24], is responsible for storing semantic descriptions of devices, services, and humans connected to the middleware and for storing the relevant paths for communication among these components. Human agents that want to make use of other IoT entities can run a semantic search over the service directory to locate other CPS components and access their data within this IoT framework. The generic system of describing human actors with an agent paradigm and the communication interfaces build a human-CPS integration framework that is not only bound to a smart factory environment but is also reusable in other contexts.

B. Human Data Model

The human data model shown in Figure 2 is used to define the human agent that represents human actors in the cyber world. The human agent is defined in an OWL-S format, and all the components shown in this figure represent Java classes. For this reason, users are represented in the service directory with their OWL-S descriptions constructed from the data model. Human actors in the physical world should also be able to exploit the data made available by the devices, machines, and IoT services and make its data available to

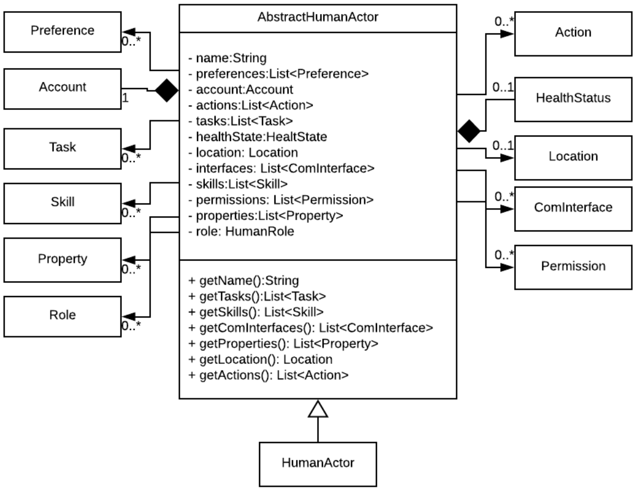

Figure 2: Human Data Model

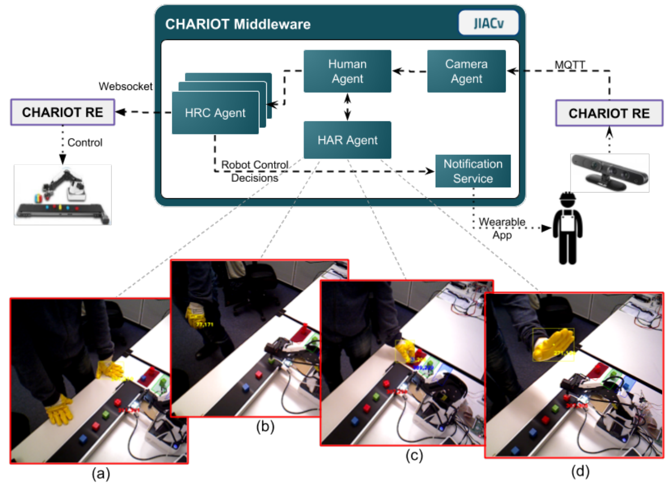

Figure 3: Human Activity Recognition Application and Human Interfacing a Robotic CPS

other components through communication interfaces (ComInterface). The agents created from this data model are able to use these communication interfaces in an abstracted way using service connection API. Besides, the human data model provides device interfaces for smartphones, smartwatches, and cameras for interaction and data exchange.

Smart factory human roles mentioned in Section II are part of the data model. These roles have default human actions, permissions and tasks defined under the generic class, but each user's role can also be customized by adding or removing actions, tasks and permissions. In addition, new roles can be defined for the users in new IoT services, by selecting a cluster of available human actions, or by analyzing user skills. Properties define the data types related with the human model, and human model can be customized by adding new properties that contain relevant information about the data such as its value, its type, and its communication interface.

A smart factory provides many alternatives for using contextual knowledge. In terms of human actors, the use of their context information enables optimized work allocation and maximum safety of the actors. If the data obtained from humans is available, a CPS can interpret, for example, their changing tiredness level, motivation, attention, capability, in order to build their characteristics. For this purpose, our human data model involves classes for data retrieval from human actors such as their health status, location, and many other multimodal inputs. Several technologies can then be integrated to observe and understand human activities. Human actors can customize their settings through user preferences in their user accounts. These preferences are also used to define privacy settings so that the users can decide which personal data they are willing to share with the other entities in the system.

IV. Applications

The example applications that we present in this section are implemented over the CHARIOT middleware and are designed to reflect the need for human-CPS integration in I4.0 scenarios. To realize the IoT middleware, the open source multi-agent framework *JIAC* [25] serves as a basis for the service-oriented distributed architecture by enabling abstracted service, device, and human actor descriptions.

A. Human Activity Recognition

One of the visions of I4.0 is to have improved cognitive capabilities in CPS, incorporating human-aware AI solutions. Human activity is one of the key information obtained from humans for the maximum context and situation awareness of a CPS interacting with humans. Nowadays, human activity recognition (HAR) systems are used for a safe, efficient and intelligent support of a robotic CPS, for example, for their human-aware collision avoidance and reliable assistance [13]. In this application, we have developed a HAR agent to recognize human activities in a cooperation scenario, where the recognized actions are further used to interpret and track a human's control signals, lost motivation, success with the assignment, and availability. This way, we build a human actor's characteristics and preferences.

In our application, we recognize human gestures of staying idle, being unavailable for the task, acting on the task and warning the robot using Hidden Markov Model with Gaussian Mixture Emissions (GMM-HMM); these gestures are respectively shown in Figure 3.a-d. Additionally, with the tracking of the products and the container they are placed in, we reason on the success status of a human on the task (as shown in Figure 3.c). Once a human actor is assigned with the role of working on the assembly line, the agent representation of the human actor (*human agent*) includes camera input as a property and receives the raw camera data from the *camera agent* (see the human data model in Figure 2 and the agents utilizing it in Figure 3). *HAR agent* retrieves this data from the *human agent* and processes the image to track the human hand and the products (colored boxes in the figure). The tracking information is then used to recognize human gestures and the state of the task is forwarded back to the *human agent* to be saved under human data model as a property.

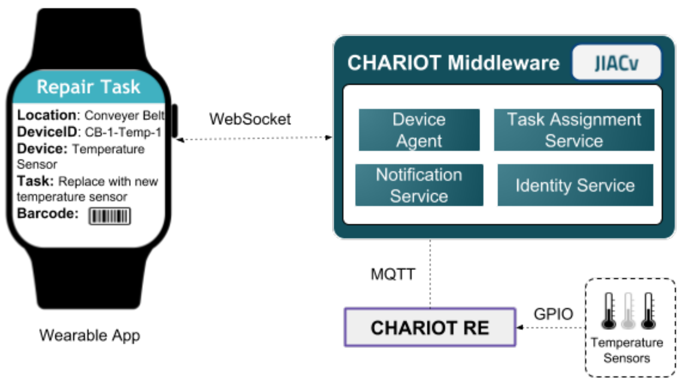

Figure 4: Smart Factory Wearable Application

Human agent multicasts these properties to the relevant agent groups in CHARIOT, thus making these properties available to a variety of smart factory applications. For example, our human-robot collaboration agent (*HRC agent*) uses an application layer protocol and subscribes to these multicasted properties. This way, the *HRC agent* can catch up-to-date human gestures, for example, to stop the robot when a warning is received for the safety of the collaboration. Additionally, *HRC agent* may decide to assist the human actor if a failure is detected with the task, e.g., misplacement of a product as in Figure 3.c, or if human is not available, e.g., the human is far away from the conveyor as in Figure 3.b or not detected in the vicinity at all. *HRC agent* also informs the robot's decisions and motions to the human actor through the notification mechanism, detailed in the next subsection.

B. Notification Mechanism

Notification mechanism depicted in Figure 4 distributes the identified tasks to the most suitable workers inside the factory. The worker is informed about the new assignment request through the notification mechanism app on her smartwatch. In addition, each permitted worker can communicate with devices such as robots in both directions. For instance, the worker can execute a service in the robot, whereas a device can inform the worker about her current status.

As part of the task assignment procedure, the health status of the worker can be monitored, should the worker accept to share her smartwatch sensor data with the IoT middleware. The monitored data is compared with the trained health data under normal conditions, and then the tracked data can be evaluated by using a model that takes parameters such as the worker's age and medical records into consideration. After this comparison, if the smart factory service detects any abnormality, the system may cancel the execution of a task by this worker.

Worker location data is also part of the notification mechanism, as a location detection module can help workers in the indoor navigation during the working hours. As many wearables have built-in Bluetooth sensors, the place of the worker can be detected by deploying Bluetooth beacons inside the factory, so that the time-critical tasks can be assigned to

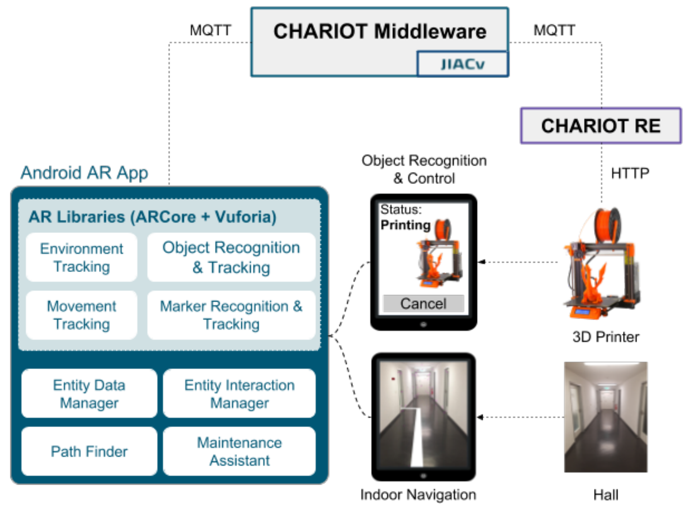

Figure 5: Smart Factory Augmented Reality Application

the worker closest to the work location with the required skills and permissions.

C. Augmented Reality (AR) Application

A smart factory CPS is a complex environment for the untrained personnel. AR enables intuitive interfaces to workers by providing the required information as to how to interact with a CPS entity, how to apply maintenance instructions, etc. AR integrates the virtually generated content to the real world in order to hide the complexity of smart factory CPS.

AR app, depicted in Figure 5, is composed of two main features: (i) object recognition and (ii) indoor navigation. Object recognition detects and identifies pre-configured physical entities using either natural feature detection or marker detection. Then, the recognized object functionalities are obtained through the CHARIOT Middleware and displayed on the worker's mobile device. Indoor navigation module can localize the user within an environment using the predefined markers that are associated with exact coordinates in the floor plan and navigate her to the target destination by computing the shortest path. This app recomputes the user's position within the learned environment and adjusts the visualized navigation path accordingly whenever a marker has been detected.

AR app retrieves all entity information from the smart factory and supplies the connected application with the required data, pushes any notification information using MQTT protocol, and delegates action requests to the environment system.

V. Conclusions

In CHARIOT, a mesh of smart factory components is represented in an IoT middleware that enables information exchange and data processing among these components including human actors. Human-CPS integration is achieved by creating a human data model, which is constructed after an analysis of human integration requirements in Industry 4.0. In this work, we present this data model together with the agent-based IoT middleware architecture, and the smart factory applications developed for highlighting the capabilities of the middleware regarding human integration. Human activity recognition application is developed for command and control of a production process on a conveyor belt. Notification mechanism handles worker task assignment by exploiting location and health status of workers, and the AR application assists human actors with object recognition and indoor navigation.

Acknowledgment

The work of authors at GT-ARC is supported by the German Federal Ministry of Education and Research within CHARIOT project with grant number 01IS16045. We thank our students Paul Braun and Zan Jelen for their efforts in developing the applications.

References

[1] P. Fantini, G. Tavola, M. Taisch, J. Barbosa, P. Leitao, Y. Liu, M. S. Sayed, and N. Lohse, "Exploring the integration of the human as a flexibility factor in cps enabled manufacturing environments: Methodology and results," in *IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society*, pp. 5711–5716, Oct 2016.

[2] A. W. Colombo, S. Karnouskos, O. Kaynak, Y. Shi, and S. Yin, "Industrial cyberphysical systems: A backbone of the fourth industrial revolution," *IEEE Industrial Electronics Magazine*, vol. 11, pp. 6–16, March 2017.

[3] KUKA, "Hello industrie 4.0 glossary," tech. rep., Kuka AG.

[4] E. A. Lee, "Cps foundations," in *Design Automation Conference*, pp. 737–742, June 2010.

[5] 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," *Computers and Industrial Engineering*, vol. 113, pp. 144 – 159, 2017.

[6] R. Harrison, D. Vera, and B. Ahmad, "Engineering methods and tools for cyber-physical automation systems," *Proceedings of the IEEE*, vol. 104, pp. 973–985, May 2016.

[7] S. Feng, F. Quivira, and G. Schirner, "Framework for rapid development of embedded human-in-the-loop cyber-physical systems," in *2016 IEEE 16th International Conference on Bioinformatics and Bioengineering (BIBE)*, pp. 208–215, Oct 2016.

[8] L. Yang, W. Li, Y. Luo, Y. Duan, and G. Fortino, "A social-d2d architecture for people-centric industrial internet of things," in *2017 IEEE 14th International Conference on Networking, Sensing and Control (ICNSC)*, pp. 744–749, May 2017.

[9] M.-P. Pacaux-Lemoine, D. Trentesaux, G. Z. Rey, and P. Millot, "Designing intelligent manufacturing systems through human-machine cooperation principles: A human-centered approach," *Computers and Industrial Engineering*, vol. 111, pp. 581 – 595, 2017.

[10] B. R. Ferrer and J. L. M. Lastra, "An architecture for implementing private local automation clouds built by cps," in *IECON 2017 - 43rd Annual Conference of the IEEE Industrial Electronics Society*, pp. 5406–5413, Oct 2017.

[11] J. Reis, R. Pinto, and G. Goncalves, "Human-centered application using cyber-physical production system," in *IECON 2017 - 43rd Annual Conference of the IEEE Industrial Electronics Society*, pp. 8634–8639, Oct 2017.

[12] S. K. Sowe, E. Simmon, K. Zettsu, F. de Vaulx, and I. Bojanova, "Cyber-physical-human systems: Putting people in the loop," *IT Professional*, vol. 18, pp. 10–13, Jan 2016.

[13] K.-D. Thoben, S. Wiesner, and T. Wuest, "Industrie 4.0 and smart manufacturing–a review of research issues and application examples," *Int. J. Autom. Technol*, vol. 11, no. 1, 2017.

[14] P. A. Lasota, T. Fong, and J. A. Shah, "A Survey of Methods for Safe Human-Robot Interaction," *Foundations and Trends in Robotics*, vol. 5, no. 4, pp. 261–349, 2017.

[15] I. Horvath, "Beyond advanced mechatronics: New design challenges of social-cyber-physical systems," in *Proceedings of the 1st Workshop on Mechatronic Design, Linz (Austria), ACCM Austrian Center of Competence in Mechatronics*, 2012.

[16] O. C. Görür, B. Rosman, G. Hoffman, and Ş. Albayrak, "Toward Integrating Theory of Mind into Adaptive Decision-Making of Social Robots to Understand Human Intention," in *Workshop on Intentions in HRI at ACM/IEEE International Conference on Human-Robot Interaction (HRI'17)*, 2017.

[17] O. C. Görür, B. Rosman, F. Sivrikaya, and S. Albayrak, "Social cobots: Anticipatory decision-making for collaborative robots incorporating unexpected human behaviors," in *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, HRI '18, (New York, NY, USA), pp. 398–406, ACM, 2018.

[18] M. Gombolay, A. Bair, C. Huang, and J. Shah, "Computational design of mixed-initiative human-robot teaming that considers human factors," *Int. J. Rob. Res.*, vol. 36, pp. 597–617, June 2017.

[19] C. Wittenberg, "Human-cps interaction - requirements and human-machine interaction methods for the industry 4.0," *IFAC-PapersOnLine*, vol. 49, no. 19, pp. 420 – 425, 2016. 13th IFAC Symposium on Analysis, Design, and Evaluation of Human-Machine Systems HMS 2016.

[20] M. Pacaux-Lemoine, Q. Berdal, S. Enjalbert, and D. Trentesaux, "Towards human-based industrial cyber-physical systems," in *2018 IEEE Industrial Cyber-Physical Systems (ICPS)*, pp. 615–620, May 2018.

[21] L. M. Hiatt, C. Narber, E. Bekele, S. S. Khemlani, and J. G. Trafton, "Human modeling for human-robot collaboration," *The International Journal of Robotics Research*, vol. 36, no. 5-7, pp. 580–596, 2017.

[22] M. Krugh and L. Mears, "A complementary cyber-human systems framework for industry 4.0 cyber-physical systems," *Manufacturing Letters*, vol. 15, pp. 89 – 92, 2018. Industry 4.0 and Smart Manufacturing.

[23] "OWL-S: Semantic Markup for Web Services ." https://www.w3.org/Submission/OWL-S/ (accessed: 2018-12-03).

[24] C. Akpolat, D. Sahinel, F. Sivrikaya, G. Lehmann, and S. Albayrak, "Chariot: An iot middleware for the integration of heterogeneous entities in a smart urban factory," in *FedCSIS Position Papers*, 2017.

[25] "Java-based Intelligent Agent Componentware." http://www.jiac.de/agent-frameworks/jiac-v/ (accessed: 2018-12-13).

Doruk Sahinel (doruk.sahinel@dai-labor.de) received his Master's degree in Information and Commmunications Systems from Hamburg University of Technology, Germany, and his Bachelor's degree in Electrical and Electronics Engineering from Middle East Technical University, Turkey. He is currently working as a researcher at DAI-Labor, TU Berlin. His main areas of research include mobility management in 5G networks and SDN.

Cem Akpolat (cem.akpolat@gt-arc.com) received his Master's degree in Computer Science from Technical University of Berlin, Germany, and his Bachelor's degree in Computer Science from Galatasaray University, Turkey. He is currently working as a researcher at the German-Turkish Advanced Research Center for ICT (GT-ARC). His main areas of research include IoT, software defined networking and fog computing.

Orhan Can Görür (orhan-can.goeruer@dai-labor.de) received his Bachelor's (2011) and Master's (2014) degrees in Electrical and Electronics Engineering from the Middle East Technical University, Ankara, Turkey. He is currently a researcher at DAI-Labor, TU Berlin, contributing to several research projects and teaching activities, mostly on Robotics and Autonomous Systems. His research interests include human-robot collaboration, human intention recognition, cognitive robotics, artificial intelligence, and multi-agent systems.

Fikret Sivrikaya (fikret.sivrikaya@gt-arc.com) received his Ph.D. degree in Computer Science from Rensselaer Polytechnic Institute, NY, USA and his Bachelor's degree in Computer Engineering from Bogazici University, Istanbul, Turkey. He is currently the scientific director at German-Turkish Advanced Research Center for ICT (GT-ARC) in Berlin, Germany. His research interests include future mobile networks, user-centric networking, wireless communications, vehicular networks, multi-hop ad-hoc networks, distributed algorithms and optimization.