Agent and cyber-physical system based self-organizing and self-adaptive intelligent shopfloor

Yingfeng Zhang*, Cheng Qian, Jingxiang Lv, and Ying Liu

***Abstract*—The increasing demand of customized production results in huge challenges to the traditional manufacturing systems. In order to allocate resources timely according to the production requirements, and to reduce disturbances, a framework for the future intelligent shopfloor is proposed in this paper. The framework consists of three primary models, namely the model of smart machine agent, the self-organizing model, and the self-adaptive model. A cyber-physical system for manufacturing shopfloor based on the multiagent technology is developed to realize the above function models. Grey relational analysis and the hierarchy conflict resolution method were applied to achieve the self-organizing and self-adaptive capabilities, thereby improving the reconfigurability and responsiveness of the shopfloor. A prototype system is developed, which has the adequate flexibility and robustness to configure resources and to deal with disturbances effectively. This research provides a feasible method for designing an autonomous factory with exception-handling capabilities.**

***Index Terms*—Smart machine agent, cyber-physical system, self-organization, self-adaptation, intelligent shopfloor.**

I. Introduction

Nowadays, the fierce market competition has imposed severe pressure on manufacturing enterprises. The ever fast changes of customers' demands have forced manufacturers to move from mass production to small and medium batched ones. The wide variety of products with small volume for each kind leads to the frequent change of production organization, and increases the possibility of exceptions to occur during manufacturing execution.

In order to solve the aforementioned problems, lots of research efforts have been conducted using the advanced technologies, such as Cyber-Physical System (CPS) [1], Internet of Things (IoT) [2], Cloud Computing (CC) [3], and Service-Oriented Technologies (SOT) [4]. These works have provided the technological basis for Intelligent Manufacturing System (IMS) and smart factories. The related works include intelligent manufacturing modes [5], IMS frameworks [6], strategies for manufacturing service configuration [7], and real-time monitoring of manufacturing execution systems [8]. Despite the significant achievements, existing manufacturing paradigms are insufficient to meet requirements imposed by typical challenges and problems in the manufacturing shopfloor. These problems are listed as follows.

(1) How to tighten the cyber-physical conjoining of the bottom-level manufacturing resources to enhance the real-time sensing capacity of machines' and manufacturing services' status?

(2) How to construct a quick-respond mechanism for proactive task allocation and self-organizing resource configuration to achieve the dynamic matching between manufacturing resources and tasks?

(3) How to achieve the self-adaptive collaboration during the manufacturing execution process, and to eliminate the disturbances when exceptions occur?

In order to address these challenges, a framework for the future intelligent shopfloor is proposed with three primary models, including the model of smart machine agent, the self-organizing model, and the self-adaptive model. A cyber-physical system for manufacturing shopfloor based on the multiagent technology is developed to realize the above function models. Here, self-organization is responsible for initially and automatically matching the manufacturing resources with tasks according to the real-time machine status in an optimal way. Then, during the manufacturing execution stage, the self-adaptive model is responsible for actively monitoring manufacturing processes and autonomously dealing with disruptions. The aim of the research is to quickly organize the production, discover and deal with abnormities without human intervention, in order to meet the requirement of product customization while reducing the cost.

The remainder of the paper is organized as follows. Section II reviews the research on agent technologies, IoT-based manufacturing systems, SOT, CPS, self-organizing and self-adaptive systems. The overall architecture of the Self-organizing and Self-adaptive Intelligent Shopfloor (SS-IS) is presented in Section III. Section IV discusses the modeling of

Manuscript submitted April 7, 2016; revised September 8, 2016.

This work was supported in part by the National Foundation of China (51675441), in part by the Doctoral Fund of Ministry of Education of China (20136102110022), and in part by the 111 Project Grant of NPU (B13044).

Y.F. Zhang is with the Department of Industrial Engineering, School of Mechanical Engineering, Northwestern Polytechnical University, Xi'an, Shaanxi 710072, China. (e-mail: zhangyf@nwpu.edu.cn).

C. Qian is with the Department of Industrial Engineering, School of Mechanical Engineering, Northwestern Polytechnical University, Xi'an, Shaanxi 710072, China. (e-mail: qch.mail@qq.com).

Smart Machine Agent (SMA) based on CPS. The task-driven self-organizing model for manufacturing systems is analyzed in Section V, and the real-time information-driven self-adaptive manufacturing model is described in Section VI. Section VII illustrates the implementations of SS-IS based on a prototype system. Conclusions and future works are drawn in Section VIII.

II. Related Works

Related research on the intelligent shopfloor is divided into three categories: (1) Enabling technologies, including IoT, Multiagent Systems (MAS), and SOT; (2) Conceptual frameworks, which include CPS and Cyber-Physical Production System (CPPS); (3) Innovative modes of production, such as self-organizing and self-adaptive manufacturing. These streams of literature are reviewed respectively as follows.

A. MAS, SOT, IoT, and Their Implementations in Industry

Agent technology has been widely developed in manufacturing applications for its autonomy, flexibility, reconfigurability, and scalability [9]. Featured with the capability for decentralized control, MASs are ideal for deploying autonomous manufacturing systems [10]. Leitao *et al.* reviewed the development in the architecture of industrial MAS and discussed the standardization of MAS [11]. The software system, or the environment of MAS, was analyzed by Valckenaers *et al.* and the connections between the real-world entities and agent systems were given [12]. Related works on the implementation of MAS into industries were extensively conducted in different fields including process and quality control [13], object management [14], manufacturing control systems [15], etc. Furthermore, Valckenaers *et al.* extended the concept of intelligent agents to intelligent beings, which focus on not only the capabilities of decision-making but reflecting the physical reality [16]. Aiming at matching services between providers and consumers, SOT and the Service-Oriented Architecture (SOA) established the connections between men and systems or within different systems. Industrial applications based on agent and SOA were discussed by Vrba *et al.* [17] and Colombo *et al.* [18]. In order to apply web services in factory automation, the theoretical foundations of that, including the resource virtualization method [19], the semantic web [20], and the optimal service composition method [21], were also studied.

Recently, many emerging technologies are greatly promoting the development of IoT [22], including Radio Frequency Identification (RFID), Near-Field Communication (NFC), Bluetooth Low Energy, LTE-Advanced, etc. Many RFID-based industrial applications were also demonstrated by Huang *et al.* [23], Makris *et al.* [24], etc. By using the wireless communication technologies, the structure modeling method of an RFID-enabled reconfigurable architecture for flexible manufacturing systems was proposed by Ali *et al.* [25]. Except for RFID, technologies like IEC 61499 [26] and NFC [27] were also applied in industrial systems.

B. CPS and CPPS

The term CPS refers to the tight conjoining of and coordination between computational and physical resources with adaptability, autonomy, efficiency, functionality, reliability, safety, and usability, which was firstly proposed by US National Science Foundation in 2006 [28]. E.A. Lee defined CPS as integrations of computation and physical processes. Embedded computers and networks monitor and control the physical processes, usually with feedback loops where physical processes affect computations and vice versa [1]. CPS provides a theoretical framework for mapping the manufacturing-related things to the computing space, so that the modeling of manufacturing systems can be easily achieved. The recent advances and trends of CPS were concluded by J. Lee *et al.* [29], and Leitao *et al.* pointed out that more research is necessary on the standardization of CPS [30].

As to the implementation of CPS, Colombo *et al.* proposed the industrial cloud-based CPS with a special focus on complex industrial systems [31]. Monostori introduced the term of CPPS and discussed the major challenges to realizing CPPS, including context-adaptive and autonomous systems, cooperative production systems, identification and prediction of dynamical systems, etc. [32]. J. Lee *et al.* and Bagheri *et al.* proposed the CPS architecture for Industry 4.0 [33] and for self-aware machines in Industry 4.0 environment [34]. Leitao *et al.* also described four prototype implementations for industrial automation based on cyber-physical systems technologies [35].

C. Self-adaptive and Self-organizing Mechanisms

Facing the fast changing market, manufacturing enterprises are seeking help from autonomous and robust production systems, so as to respond rapidly to market changes. Actually, the traditional manufacturing systems with centralized and hierarchical control approaches "present good production optimization," but are weak in response to changes [36]. Implied by the manner in which holonic systems emerge, adapt and survive, Valckenaers *et al.* revealed the fundamental principles of the design of self-organizing and self-adaptive systems [37]. Leitao *et al.* presented the adaptive holonic control architecture (ADACOR) for distributed manufacturing systems to improve the system performance in terms of the agile reaction to emergency [36]. Many further studies on self-adaptive systems were based on this architecture. For example, Barbosa *et al.* further explained the biologically-inspired ADACOR architecture, analyzed the transitions between the stationary state and the transient state, and extensively discussed the evolution of it [38]. Except for dealing with changes, self-adaptation was also used in production coordination. For example, also inspired by biological characteristics, Belle *et al.* proposed the method for proactive coordination in logistic systems [39]. Monostori *et al.* analyzed the collaborative control in production systems and introduced predictive method into this field [40].

Basically, the self-adaptive and self-organizing mechanisms have similar goals, i.e. to increase the systems' responsiveness, flexibility, reconfigurability, and autonomy. In this research, as mentioned before, self-adaptation monitors processes and deals with disturbances, while self-organization focuses on maximizing the system autonomy and matching tasks with resources. In this sense, research on self-adaptation include dynamic task allocation [41], adaptive scheduling [42], and evaluating the capabilities of dynamic reconfiguration of an industrial system [43].

The literature above provide lots of theoretical models and algorithms for self-adaptive and self-organizing systems. These are the fundamental works for intelligent shopfloor of the next generations. However, more attention should be paid on the modeling of bottom-level manufacturing resources, which can greatly help with the realization of embedded intelligence. The models of the self-adaptive and self-organizing mechanism can be easily applied only if the tight conjoining of computational and physical resources is achieved.

III. Overall Architecture

This research applies the concept of CPS and develops an easy-to-deploy and simple-to-use framework to achieve self-organization and self-adaptation for the future intelligent shopfloor. Fig. 1 shows an overall architecture of the proposed agent and CPS based intelligent shopfloor. It consists of three main components to enhance the self-organizing and self-adaptive capability of the shopfloor, namely the smart machine agent, the self-organizing model, and the self-adaptive model.

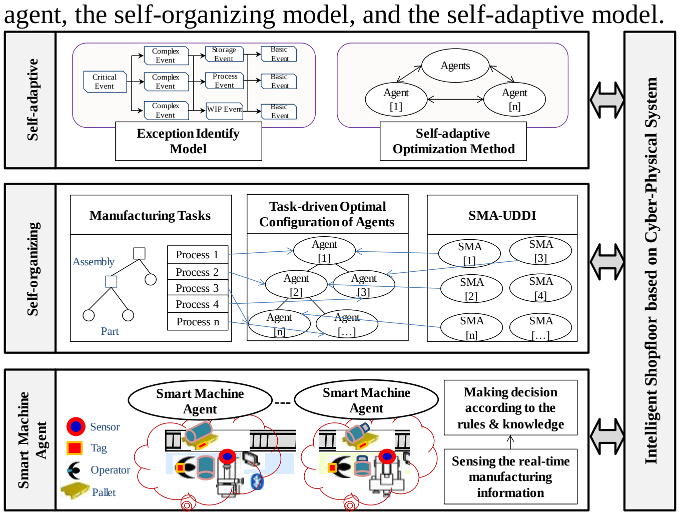

Fig. 1. Overall Architecture of the Intelligent Shopfloor

With the help of IoT devices, the real-time manufacturing information is captured by SMAs. SMAs can communicate with each other and drive the executors according to the rules, as is described in Section IV. Manufacturing tasks are decomposed into process-level, and these tasks are obtained by the SMAs, as designed by the self-organizing model. The self-adaptive mechanism monitors the manufacturing processes, trying to identify the exceptions and adapts to the exceptions autonomously. Compared to the current manufacturing paradigms, the proposed intelligent shopfloor has the following new features.

Firstly, applying the designed cyber-physical system based on the multiagent technology, the tight conjoining of the top-level intelligent models and the bottom-level manufacturing resources is achieved. Manufacturing machines can sense the real-time manufacturing environment, and have the capability of making decisions, thus the machines become smart. Secondly, based on the self-organizing model, machine resources can be configured through Grey Relational Analysis (GRA) [7] when the shopfloor receives manufacturing tasks. Thirdly, during manufacturing execution stage, production exceptions could be proactively identified, and the influence of them will be decreased or eliminated by applying the designed self-adaptive model.

IV. Smart Machine Agent based on CPS

Manufacturing machines are the basic execution units for production, thus the enhancement to the intelligence of the machines will provide strong support for the intelligent shopfloor. The SMA aims to enable the machine of sensing information and making decisions autonomously by using CPS and agent technologies. Here, CPS supports the data infrastructure, while manufacturing systems are modeled as multiagent systems under CPS. As shown in Fig. 2, SMA is composed of two main modules, namely cyber-physical machine, and smart machine agent, from bottom to top.

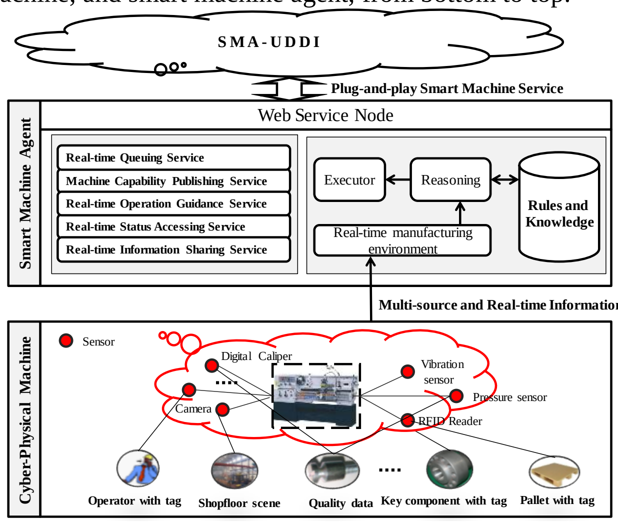

Fig. 2. Modeling of the Smart Machine Agents

A. *Cyber-Physical Machine*

The cyber-physical machine module is responsible for capturing multi-source and real-time manufacturing information around the machine by using auto-ID technologies [44]. The aim of this module is to enhance the sensing ability of traditional manufacturing machines. By applying the advanced IoT technologies (e.g., RFID, digital caliper, pressure sensor, etc.), traditional machines are enabled to capture the real-time manufacturing information proactively. Take RFID as an example, in order to monitor the real-time status of operators, assembly progress, and work-in-process (WIP) inventories, three areas are designated, namely the raw material site, the assembly site, and the finished product site. An antenna is then installed in each area to capture the manufacturing-related things (e.g., operator's ID cards, RFID tag-embedded pallets with WIP products on them, etc.). Then, the real-time manufacturing data are collected through a RFID reader connected with these antennas, indicating the current location of manufacturing things. These data streams are provided to the SMA module and can be further interpreted as manufacturing progress or state indications.

B. Smart Machine Agent

Smart machine agent provides the core services of the machine. Here, agent technologies and service-oriented architecture are used to make the machine capable of making decisions intelligently and autonomously. This module includes real-time manufacturing data perception, the pool of rules and knowledge, the reasoning mechanism, and the executor. The reasoning mechanism can match the rules from the pool with the production data. Rules are processing requirements or standards processed from raw manufacturing data, which can be interpreted by the manufacturing system. A simple example of the rule is the current process requires certain temperature. Thus, the executor (i.e., a heater) will act on the rules and adjust the environmental temperature. The executor is responsible for informing manufacturing systems the ongoing operations and acting on the instructions from the manufacturing system. Knowledge refers to the professional instructions or information that are beneficial to the effective production at the machine side. Moreover, services of the machine are wrapped by its agent, so that SMA becomes a plug-and-play component connected to the intelligent shopfloor, and can be visited by operators or interoperated by other machines through web services [19]. In other words, each SMA may provide some services so that they can be invoked using SOT. These services wrapped by different SMAs forms different web service nodes, which represent the corresponding machines. Major services provided by SMA are described as follows.

(1) Machine Capability Publishing Service

Manufacturing capability of machines is published to the industrial networks so that they can be discovered as potential resources to undertake the suitable manufacturing tasks. Machine capability information includes basic information (such as machine ID, machine name, etc.) and capability information (such as processing method, maximum processing size, manufacturing precision, processing roughness, etc.).

(2) Real-time Status Accessing Service

This service provides basic manufacturing information by applying advanced IoT and CPS technologies. By design, authorized third-party services can access to the real-time status of manufacturing things (such as WIP items, the materials of in-buffer, working area and out-buffer, parameters of machine etc.) through Internet.

(3) Real-time Operation Guidance Service

The operation guidance service is designed to provide the operators with operation details and instructions during the manufacturing process, which could greatly reduce the chance of quality defects caused by improper operations or wrong installations of materials.

(4) Real-time Information Sharing Service

This service is responsible for establishing the dynamic information connection between the upstream and downstream SMAs and manufacturing machines. These SMAs can get the collaborative information of other interrelated machines, which can assist the SMAs to timely identify the exceptions and to further come out with the proper solutions.

(5) Real-time Queuing Service

This service aims to reorder the queue of the tasks for each manufacturing machine and reconfigure related resources according to the real-time information from the upstream and downstream stations (e.g., the lack of raw materials, changed delivery time, new task with high priority, etc.).

C. Proof-of-Concept SMA Prototype

Following the architecture and core modules of the designed SMA, our research lab has developed a proof-of-concept SMA prototype by combining CPS, CC, and agent technologies.

The developed prototype consists of two main parts, i.e., hardware and software. In terms of the hardware, as shown in Fig.3, it includes an integrated computer and some sensors, e.g., RFID antennas and digital calipers. The computer serves as the digital communication interface and is responsible for connecting different types of sensors through wired or wireless connections. Production data can be obtained through sensors which are plugged to the prototype. Control signals are sent to the executors of different machines through the data interfaces provided by the computer. As to the software, it consists of agents, drivers of all kinds of sensors, and web service software (e.g., Tomcat). After connecting to the integrated computer, sensors capture and transfer data to agents by using the drivers. All the services discussed above are installed and wrapped as cloud manufacturing services. The software (or the services) inside the SMAs regard machines as agents when applying self-organizing and self-adaptive models. The SMAs are installed at the machine side, so that the real-time manufacturing information can be timely shared, and SMAs can reconfigure themselves according to the information. The hardware and the software system, as a whole, can be attached to manufacturing machines easily to form the SMAs.

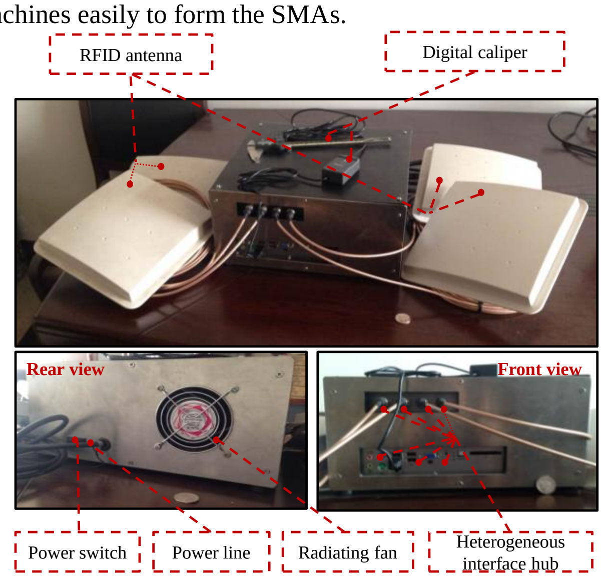

Fig. 3. Prototype of the SMA Hardware System

V. TASK-DRIVEN SELF-ORGANIZING MODEL

In order to improve the operational efficiency of SMA, the task-driven self-organizing manufacturing model is proposed to deal with the task allocation problem by invoking services provided by SMA. SOT (e.g., SOA, web service, ontology, etc.) can be applied for constructing the virtual manufacturing and service environment, which is one of the key enabling technologies to realize accessing, invoking, deployment and on-demand use of smart machines, and to realize a self-organizing factory [7]. Considering the complexity and diversity of manufacturing resources, SOTs are adopted to establish the scientific information-based model for task allocation. GRA is applied to search for the most suitable pairing between tasks and machines from a set of alternatives by analyzing relational grade among the discrete sets [7], so as to realize the optimal configuration of machine resources.

A. The Logical Flow of the Self-organizing Model

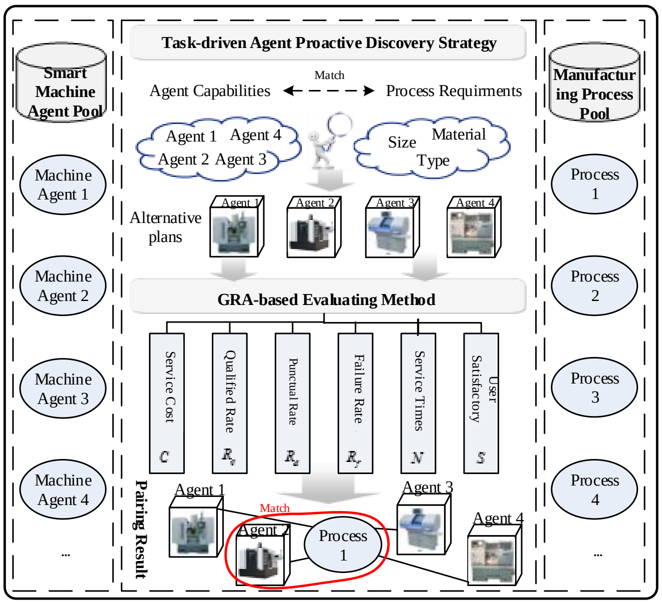

Fig. 4. Task-driven Self-organizing Model

Fig. 4 illustrates the framework of the task-driven self-organizing manufacturing model. The self-organizing process can be divided into three phases.

Phase 1: Virtualization of machines and tasks. As can be seen in the left part of Fig.4, based on a resource servitization method [19], manufacturing resources are virtualized into smart machine agents, which highlight their functional attributes and capabilities. Similarly, the information of tasks is also analyzed to emphasize the requirements of each process, which is shown in the right part of Fig. 4. Complex tasks will be decomposed into process-level ones, so that each process can be handled by a single machine.

Phase 2: Proactive discovery of tasks. As shown in the middle-upper part of Fig. 4, SMAs cyclically check if there are new processing tasks. When new tasks are detected, SMAs submit requests to undertake the processible tasks proactively according to their current status. At this stage, each manufacturing process has a candidate set of all available SMAs.

Phase 3: Optimal configuration of machines. For each manufacturing process, this phrase is designed to pick out the optimal SMA from the large-scale solution space, where all demands of the manufacturing processes are met accordingly by SMAs in an optimal way. Manufacturing processes are matched with machines in this phrase and the manufacturing execution begins.

B. SMA Proactive Discovery Modelling

A number of attributes are required to uniquely identify the individual SMAs, and these attributes can generally be divided into four categories: the basic attributes, the function attributes, the evaluation attributes, and the real-time status attributes. The basic attributes describe the general situation of manufacturing machines, which usually consist of service ID, shopfloor ID, the purchasing date, manufacturer and the usable lifetime of the machine. Function attributes show the detailed capacities of SMA, which are the essential prerequisite of service proactive discovering. Function attributes usually include processing part type, processing method, achievable processing size, processing material, processing precision, processing roughness, and other processing characteristics. Evaluation attributes are used in the optimal configuration process, as they provide measurable criteria to evaluate the capacities of machines. The cost of service, qualification rate, on-time delivery rate, reliability, service frequency, maintainability, and customer satisfaction are usually the major factors of evaluation attributes. Real-time status attributes include service status, manufacturing task sequence, the load status of the machine and detailed processing information, which provide traceable information within the entire manufacturing environment.

C. SMA Optimal Configuration

The most significant part of the optimal configuration is to establish a systematic evaluation method, which can reflect machining cost, storage cost, the agreed delivery time, delay time (if the manufacturing process exceeds the deadline), the execution reliability of SMA, energy consumption index, qualified rate, and the credit of SMA. The evaluating process can be realized by adopting the GRA-based evaluating method.

The evaluating method mainly focuses on the cost of service, qualification rate, on-time delivery rate, reliability, service times, and user satisfaction. The cost of service (*C*) usually contains the cost of production (C_p) and cost of logistics (C_l). It is assumed that the total cost of logistics is directly proportional to the cost of logistics for each product. Therefore, the cost is defined as $C = (C_p + C_l) \cdot LotSize$. *Lotsize* represents the batch of manufacturing tasks. Qualification Rate (R_q) is an important criterion of manufacturing capability. R_q is determined by the formula $R_q = 1 - n_{iw}/n_i$, where n_i represents the total production number of this type by the machine, and n_{iw} represents the scraps made by the machine. These data are provided by historical production records from the knowledge pool in SMAs. The on-time delivery rate (R_d) shows the machine's ability to process the task on time, and it is determined by the formula $R_d = N_{ot}/N$ where N represents the total production number of a machine and N_{ot} means the production number when products are delivered on time. Unreliability (R_f) is calculated by the formula $R_f = N_f/N$ where N_f represents the accumulated failure times during production. User satisfaction (*S*) shows the machine's ability to meet customers' expectations and is decided by the formula

$S=\sum_{i=1}^{N} S_i / N$ where S_i indicates the points given by individual customers for their products.

To apply the GRA-based evaluating method, three steps need to proceed. Firstly, the ideal indicator sequence is determined by both task requirement information and the types of evaluation indicators. The sequence is given as $A^*=\left(a_1^*, a_2^*, \cdots, a_n^*\right)^{\mathrm{T}}$. Secondly, due to different dimensions in the indicators, they need to be normalized so that the evaluation results can be more reliable and accurate. Let $y_j^*=\frac{a_j^*-a_j^{\min}}{a_j^{\max}-a_j^{\min}}$, $y_j^i=\frac{a_j^i-a_j^{\min}}{a_j^{\max}-a_j^{\min}}$ where $a_j^{\min}$ represents the minimum value of the j^{th} indicator ($a_j^{\min}=\min a_j^i$), $a_j^{\max}$ represents the maximum value of the j^{th} indicator ($a_j^{\max}=\max a_j^i$), $i=1,2,\cdots m$, $j=1,2,\cdots n$. Then y_j^* and y_j^i are the normalized value of a_j^* and a_j^i. Thirdly, according to the gray theory, $\xi_i(j)$ represents the relational coefficient of the j^{th} indicator of the i^{th} service and the ideal indicator, and $\xi_i(j)=\frac{\min\limits_i \min\limits_j \left|y_j^*-y_j^i\right|+\rho \max\limits_i \max\limits_j \left|y_j^*-y_j^i\right|}{\left|y_j^*-y_j^i\right|+\rho \max\limits_i \max\limits_j \left|y_j^*-y_j^i\right|}$ where ρ is distinguishing coefficient, and $\rho \in[0,1]$. Therefore, the obtained relational coefficient matrix is $E=[\xi_i(j)]_{m*n}$. The customized vector $W=\left(w_1, w_2, w_3 \ldots, w_n\right)^T$ is used to represent the weight of each indicator, which can be determined by the widely used Analytic Hierarchy Process (AHP) [45]. Then, the comprehensive evaluation matrix is achieved as $R[r^i]=EW$. The greater r^i equals, the better machine and process matches.

By applying the proposed task-driven model, manufacturing processes in the future factories are organized autonomously and efficiently.

VI. Real-time Information Driven Self-adaptive Model

The task-driven self-organizing model can provide an initial production plan, but it is still difficult for the shopfloor to response to and to deal with exceptions occurred during manufacturing execution stage due to the lack of real-time feedback of the disturbances. Therefore, the self-adaptive model is brought to this work to make the shopfloor intelligent enough to actively discover, identify and eliminate or decrease the influences caused by the exceptions.

To fulfill this purpose, two main components are discussed. They are the real-time exception identification model and the self-adaptive conflict resolution model.

A. Real-time Exception Identification Model

The event-driven real-time exception identification model is shown in Fig. 5. The real-time events of the sensors installed at distributed manufacturing machines will provide the basic data. A multi-level event structure is proposed to convert the distributed manufacturing data of the events to meaningful manufacturing information, and the Petri net model is constructed according to the relationships among different events. As a result, the multi-level events can be extracted easily by analyzing the Petri Nets, and then the exceptions can be identified by comparing the real-time events with the planned production status.

The multi-level event structure is used to define the hierarchical structure of the manufacturing system, which has four types of events, namely Primitive Event (PE), Basic Event (BE), Complex Event (CE), and Critical Event (CrE). PEs are raw sensor events, BEs are resource-level events, CEs are cell-level events, and CrEs are shopfloor/product-level events. The events are defined in a standard model: {*id*, *name*, *context*, *attributes*, *t*}, where, *id* is the unique id of the event, *name* stands for the event name, *context* specifies the context information needed to describe the event, *attributes* provide the related parameters, and *t* represents the time of the event.

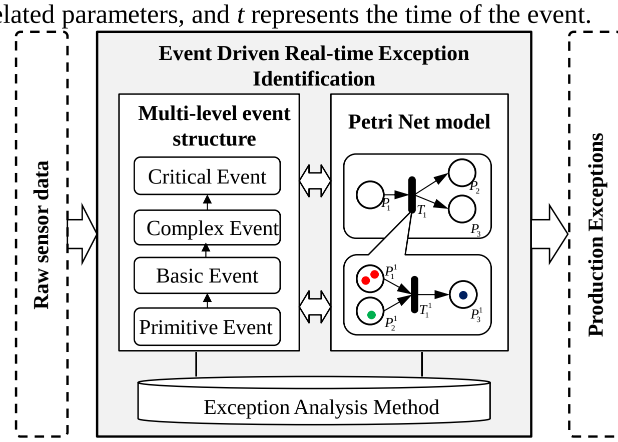

Fig. 5. The Event-driven Exception Identification Model

In the event model, PEs are simple events, which can be directly obtained through SMA. The other three kinds of events are composite events, which are aggregated from the sub-events using logical operator (e.g. and, or, negation, etc.) or temporal operator (e.g. sequential, within), and the sub-events may be either simple events or complex events. Take the real-time production progress as an example, the actual processing time can be acquired by querying the basic events, so the production of the part, assembly or product can be obtained according to the hierarchy relationships of the manufacturing bill of materials (MBOM). Compared with the production plan, the RPP can be timely calculated according to its Critical Event *CrE*= (*CrE_ID*, *Product.id*, *p*, *t*). The parameter *p* is given in (1).

$$p=\sum_{i=1}^{r}\sum_{j=1}^{p}\sum_{k=1}^{q} T_{P_{ijk}}^{current} \Big/ \sum_{i=1}^{r}\sum_{j=1}^{p}\sum_{k=1}^{q} T_{P_{ijk}}^{total} \tag{1}$$

In (1), *CrE_ID* is the unique id of the critical event, *p* is the progress of the assembly, and p_{ijk} is the k^{th} process of part *j* for Assembly *i*. *r*, *p*, *q* are the total number of assembly, parts, and processes, respectively. T_p is the process time.

The hierarchical timed-colored Petri Net (HTCPN) models are used to construct and analysis the multi-level event. A HTCPN is an 8-tuple *N*=<*P*,*T*,*C*,*I*,*O*,*G*,*D*,*M*>, where, *P* denotes places, which are used to represent activities; $T=\{T_i, T_t, T_s, T_m\}$

is a finite set of transactions with T_i, T_t, T_s, and T_m being immediate, timed, random, and macro transitions, respectively, and T_m are used to model sub-events; C denotes the color mapping from $P \cup T$ to W, an item of $C(s)$ is called a color of s and $C(s)$ is the color set of s, where s is the attributes of P or T; $I(O)$ functions denote the forward (backward) incidence matrix of $P \times T$, which represent the relationship between transitions and places; G is a guard function and maps each transaction T to a Boolean expression (called guard expression). D gives the time delays of a timed transition in T_t or random transition in T_s, once a transition in T_t or T_s is enabled, it cannot fire until D time units are elapsed; M is a marking representing the number of tokens in P and is a vector with M_0 being the initial marking. The colored tokens can carry time, quantity, and other attributes, and they are combined with IoT technologies so that the PN status can be updated according to real-time manufacturing actions. Based on the performance of Petri Net model, the multi-level events can be acquired [46].

The exceptions events are defined as the abnormal status of an object or a system. It occurs when the real-time events deviate from its plan status. Defined by the Exception analysis method, the exceptions at different levels can be detected easily, which provides the important inputs for the self-adaptive model.

B. *Self-adaptive Conflict Resolution Model*

The real-time exception identification model provides important inputs for self-adaptive conflict resolution. Since exceptions may occur at different levels, the self-adaptive conflict resolution model is designed accordingly to eliminate or decrease the influence caused by manufacturing exceptions in time. Based on the exception identification model and self-adaptive optimization strategy, the analytical conflict resolution model is designed. Fig. 6 presents the overview of the self-adaptive conflict resolution model.

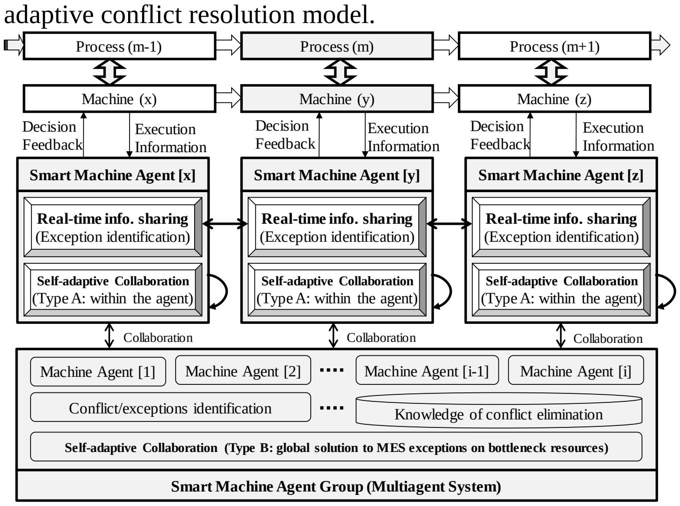

Fig. 6. Overview of the Self-adaptive Conflict Resolution Model

The exceptions are divided into two types according to the scale of influences caused by them. As shown in Fig. 6, exceptions that are caused to PEs or BEs can generally be solved at the machine side, while the exceptions that are caused by CEs or CrEs can only be solved under the collaboration of multiple SMAs or the entire shopfloor. The self-adaptive optimization processes of the two types of exceptions are defined as *Type A* and *Type B self-adaptive process*. For example, if the spindle speed of a lather machine exceeds the desired upper limit (primary sensor-level event) or the raw material has not been sent to the machine (resource-level event), the machine adjusts its operational parameters or broadcasts messages to locate the required material (*Type A self-adaptive process*). If several machines broke down (complex event), reconfiguration among SMAs would then take place (*Type B self-adaptive process*).

The following strategy is used to reorder the unfinished tasks of the relevant SMAs. In order to response to the exception rapidly and to obtain an optimal new job queue, the objective function is defined in (2). It aims to minimize the total weighted delay time of all the tasks in the reordered queue, and w_j is the weight (delay penalty). Here, the delay time DT_j^i of task J^i is calculated by (3), which indicates the deviation between the finished time of task J^i of the new job queue and the due time d_j^i of the task J^i. J^i represents the process of job J processed at station i, and ET_j^i represents the finished time of J^i. The due time of each task is continuously changed according to the exception information of the upstream and downstream machines and is calculated by (4), where p_j^i represents the processing time of J^i, and ST_j^i represents the start time of J^i.

$$F = \min \sum_{j=1}^{m} w_j DT_j^i \tag{2}$$

$$DT_j^i = \max(ET_j^i - d_j^i, 0) \tag{3}$$

$$d_j^i = \max(ET_j^{i-1} + p_j^i, ST_j^{i+1}) \tag{4}$$

The result of such calculations can be obtained by applying genetic algorithms such as Tabu Search. After the bottleneck of manufacturing resources is identified and removed, the normal manufacturing processes proceed.

VII. Case Study

To verify the effectiveness and efficiency of the proposed SS-IS, a case related to our business partner is discussed and studied. This company has a typical discrete manufacturing system for engine production in China. After investigating the assembly shopfloor for two weeks, we found that the manufacturing information reported to managers may not accurately and promptly reflect the real-life situations, and it may further intensify the production disturbances when exceptions occur. The managers have to constantly deal with the changed production orders and are busy at reconfiguring resources. Therefore, they are sorely in need of the self-organizing and self-adaptive solutions.

It is stated that the comprehensive implementations of the proposed SS-IS to a real-life company may be a difficult and complex task. To demonstrate the advantages of this research, based on the discussions above, a hypothetical case scenario is considered, and a proof of concept experiment and prototype is designed and developed. The shopfloor is equipped with RFID hardware systems to realize timely data-collecting. The modularized SMA attachment which is shown in Fig. 3 can be set up for each machine easily. Based on data captured in real time, manufacturing processes are constantly monitored through SMAs. By analyzing the data, the production can be organized autonomously. The feedback information from SMAs enables the shopfloor to optimize production dynamically and to trace exceptions effectively.

The operating procedures of the shopfloor are described as follows. To obtain the real-time multisource manufacturing data from the shopfloor, RFID tags, antennas, and many kinds of sensors are set up in the execution layer which is the physical environment of manufacturing shopfloor. With the hardware installed, machines are becoming SMAs according to the model described in Chapter IV.

A new manufacturing task is received by SS-IS and is decomposed into manufacturing processes. SMAs captured the demanding manufacturing processes and send requests to undertake the task according to their current status. Therefore, a candidate set for each manufacturing process is established. The pairing mechanism selects some pairs of machines and processes which are exact matches, and a set of machines with suitable capabilities is formed, e.g., {M1, M2, M3, M4, M5, M6} is a set of available machines for the lathe work. The evaluating indicators for each machine are given in terms of the historical information in the knowledge pool of SMAs in Table 1.

Table 1. SMAs Evaluating Indicators

Indicator	M1	M2	M3	M4	M5	M6
Cost	1510	820	1585	1700	1610	1550
Qualification Rate	83%	90%	81%	92%	95%	87%
On-time Delivery Rate	91%	85%	92%	98%	94%	97%
Unreliability	7%	14%	6%	2%	4%	5%
Service Times	820	1240	435	282	688	791
Satisfaction	85%	81%	86%	92%	89%	91%

After that, the GRA-based evaluating method is applied to pick up the most suitable machine for the process.

(1) The ideal indicator sequence is given.

$$A^* = (820, 0.95, 0.98, 0.02, 1240, 0.92)^{\mathrm{T}}$$

(2) Normalizing the evaluating matrix.

$$Y = \begin{bmatrix} 0.78 & 0 & 0.87 & 1 & 0.90 & 0.83 \\ 0.14 & 0.64 & 0 & 0.79 & 1 & 0.43 \\ 0.46 & 0 & 0.54 & 1 & 0.69 & 0.92 \\ 0.42 & 1 & 0.33 & 0 & 0.17 & 0.25 \\ 0.56 & 1 & 0.16 & 0 & 0.42 & 0.53 \\ 0.36 & 0 & 0.45 & 1 & 0.73 & 0.91 \end{bmatrix}$$

(3) Calculating the relational coefficient matrix (Let $\rho = 0.5$).

$$E = \begin{bmatrix} 0.39 & 1 & 0.37 & 0.33 & 0.36 & 0.38 \\ 0.37 & 0.58 & 0.33 & 0.70 & 1 & 0.47 \\ 0.48 & 0.33 & 0.52 & 1 & 0.62 & 0.87 \\ 0.55 & 0.33 & 0.60 & 1 & 0.75 & 0.67 \\ 0.53 & 1 & 0.37 & 0.33 & 0.46 & 0.52 \\ 0.44 & 0.33 & 0.48 & 1 & 0.65 & 0.85 \end{bmatrix}$$

(4) The weight of each indicator is given by using AHP.

$$W = (0.324, 0.143, 0.157, 0.112, 0.109, 0.155)^{\mathrm{T}}$$

(5) The comprehensive evaluation matrix is achieved.

$$R = EW = (0.46, 0.51, 0.60, 0.62, 0.54, 0.58)^T$$

The fourth element (0.62) in the R is the maximum value among other results, thus M4 is chosen to complete the required lathe work according to the task description in the task pool. Similarly, all process-level tasks will be assigned to different SMAs by repeating this calculation, and the matching between al tasks and SMAs is then achieved.

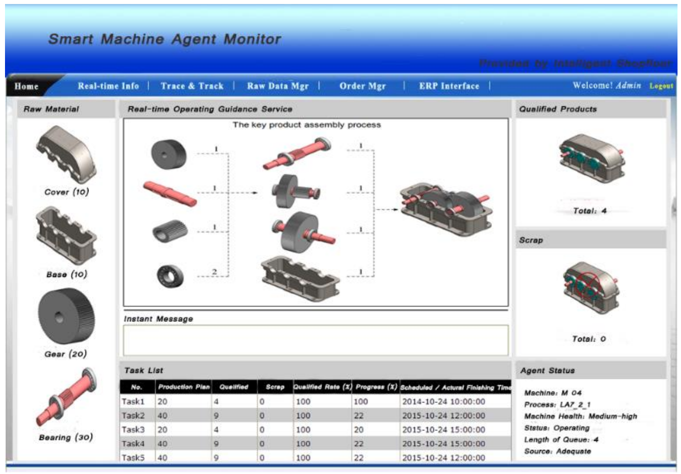

Fig. 7. Smart Machine Agent of the Prototype System

During the manufacturing execution stage, take M4 as an example, the visual interface of its SMA is shown in Fig. 7. All the tasks that are allocated to M4 are listed below. The guidance for the task will be displayed on the SMA of machine M4. The same system runs on each SMA, and it can provide information on materials that flow into them and the products that flow out of them according to the data captured from RFID readers and other sensors.

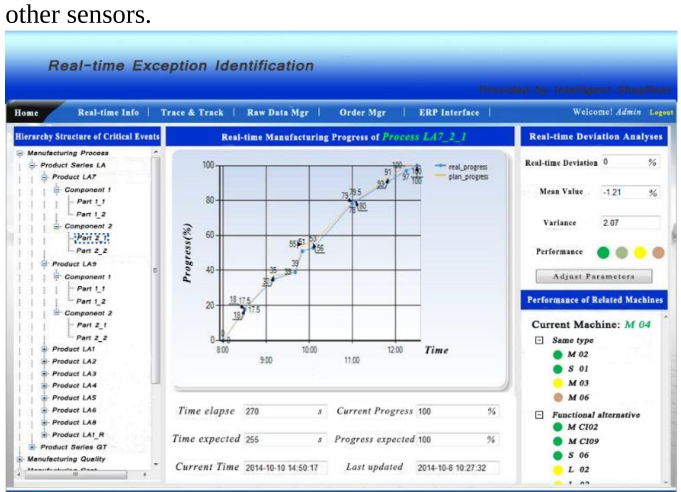

Fig. 8. The Real-time Exception Identification Module

Meanwhile, the exception identification model keeps listening for potential resource conflicts. When exceptions are identified, they will be thrown to the self-adaptive conflict resolution model, as is shown in Fig. 8. The manufacturing progress and the deviation are shown in graphic and the progress of each component can be displayed by selecting the corresponding item in the list leftwards, which is the hierarchy structure of critical events.

There are 2 machines (i.e. M1 and M2) and 3 tasks involved in our demonstration. For each task, 2 processes need to be finished one after another. The pair of number (i, j) in Fig. 9, represents the j^{th} process of the i^{th} task. Processes (1,2), (2,1) and (3,1) can only be processed on machine M2. At least 1 unit of time is needed for shifting tasks. Before the exception occurs, the production plan follows the initial scheduling result, which is shown in the Gantt chart of Fig. 9(a). At time 0, the arrival of the raw material for process (2,1) is postponed for 6-time units, which is acquired by the SMA of M2. Then the schedule for the related machines is recalculated by the self-adaptive model, which is shown in Fig. 9(b). Following the new scheduling result, the whole production is 1 unit late due to the exception. By comparison, without the designed CPS and SMA, exceptions cannot be identified timely. As a result, the tradition solutions (e.g. manual reassignment of tasks) are not supported by comprehensive manufacturing information. Also, the workers have no idea when the delayed material will arrive, so they cannot make sure which process should be brought forward. They may come to different scheduling result as shown in Fig. 9(c) and Fig. 9(d). The machine utilization in Fig. 9(c) is quite low. Since it took the workers 2 units longer to react, the scheduling result in Fig. 9(d) is also worse than that in Fig. 9(b). The intelligent shopfloor can respond more timely, and the solutions are based on calculations rather than workers' experience.

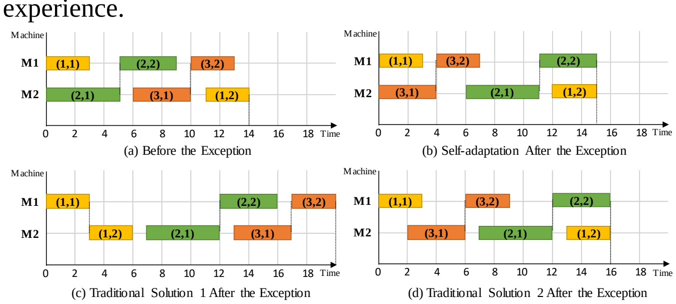

Fig. 9. The Scheduling Result before and after the Exception

VIII. Conclusion and Future Work

Manufacturing systems need to enhance their responsiveness and reconfigurability to meet the multi-type and fast-changing requirements from customers. In order to achieve the real-time, seamless and dual-way connectivity and interoperability between manufacturing machines and the shopfloor, an easy-to-deploy and simple-to-use framework for the future intelligent shopfloor is developed by applying the concept of CPS. In this research, manufacturing machines are modeled as smart agents which can collect production data and control the machines. With the help of the self-organizing model, machines can be reconfigured for different tasks to achieve the highest resource efficiency. Manufacturing processes are monitored and adjusted by the self-adaptive model when exceptions occur.

There are three main contributions presented in this work. Firstly, the architecture and the function models of the intelligent shopfloor provides a reference for the future designs. Secondly, a cyber-physical system for manufacturing shopfloor based on the multiagent technology is developed. By implementing SMAs, the tight conjoining of the top-level intelligent models and the bottom-level manufacturing resources is achieved. Finally, the self-organizing and self-adaptive mechanisms are introduced, which gives an example of how to construct a manufacturing system with high autonomy, adaptability, efficiency, and functionality. This work provided a feasible approach to implementing CPS so that the models and algorithms on self-organization and self-adaptation can be easily applied.

The insufficiency of the case analysis presents one of the main limitations of the study. Since a complete production process is complex and may include several manufacturing systems, only limited situations were tested by the case. The future work will mainly focus on the extension of self-organizing and self-adaptive methods in the intelligent manufacturing field, so that the proposed framework and models will be more robust and reliable under complex situations.

Acknowledgment

The authors would like to thank the editors and reviewers for their constructive suggestions on this work.

References

[1] E.A. Lee, "Computing Foundations and Practice for Cyber-Physical Systems: A Preliminary Report," Electr. Eng. Comput. Sci., Univ. Calif. Berkeley, Tech. Rep. UCB/EECS-2007-72, May 2007.

[2] C. Perera, C.H. Liu, and S. Jayawardena, "The Emerging Internet of Things Marketplace From an Industrial Perspective: A Survey," *IEEE Trans. Emerging Topics in Comput.*, to be published.

[3] M. Hamdaqa and L. Tahvildari, "Cloud Computing Uncovered: A Research Landscape," Adv. Comput., vol.86, pp.41–85, 2012.

[4] F. Tao, L. Zhang, V. C. Venkatesh, Y. Luo, and Y. Cheng, "Cloud manufacturing: a computing and service-oriented manufacturing model," *Proc. Inst. Mech. Eng. B*, vol.225, no.B10, pp.1969-1976, Oct 2011.

[5] F. Tao, Y. Zuo, L. Xu, and L. Zhang, "IoT-based intelligent perception and access of manufacturing resources toward cloud manufacturing," *IEEE Trans. Ind. Informat.*, vol. 10, no.2, pp.1547-1557, 2014.

[6] A. Cannata, M. Gerosa, and M. Taisch, "SOCRADES: A framework for developing intelligent systems in manufacturing," in *Proc. IEEE Int. Conf. Ind. Eng. and Eng. Manage.*, Singapore, 2008, pp.1904-1908.

[7] Y.F. Zhang, D. Xi, R. Li, and S.D. Sun, "Task-driven manufacturing cloud service proactive discovery and optimal configuration method," *Int. J. Adv. Manuf. Technol.*, vol.84, no.1, pp.29-45, 2016.

[8] Y.F. Zhang and S.D. Sun, "Real-time data driven monitoring and optimization method for IoT-based sensible production process," in *Proc. 10th IEEE Int. Conf. Networking, Sensing and Control*, Evry, 2013, pp. 486-490.

[9] F.P. Maturana, P. Tichy, P. Slechta, F. Discenzo, R.J. Staron, and K. Hall, "Distributed multi-agent architecture for automation systems," *Exp. Syst. with Appl.*, vol. 26, no. 1, pp. 49–56, 2004.

[10] M. Wooldridge, *An Introduction to Multi-Agent Systems*. Hoboken, NJ: Wiley, 2002.

[11] P. Leitão, V. Mařík, and P. Vrba, "Past, present, and future of industrial agent applications," *IEEE Trans. Ind. Informat.*, vol. 9, no. 4, pp. 2360–2372, Nov. 2013.

[12] P. Valckenaers, J. Sauter, C. Sierra, and J.A. Rodriguez-Aguilar, "Applications and environments for multi-agent systems," *Auton. Agent Multi-Agent Syst.*, vol. 14, pp. 61–85, 2007.

[13] P. Leitão, N. Rodrigues, C. Turrin, and A. Pagani, "Multiagent System Integrating Process and Quality Control in a Factory Producing Laundry Washing Machines," *IEEE Trans. Ind. Informat.*, vol. 11, no. 4, pp. 879-886, Aug 2015.

[14] Y.F. Zhang, G.Q. Huang, T. Qu, O. Ho, and S.D. Sun, "Agent-based Smart Objects Management System for Ubiquitous Manufacturing," *Int. J. Robot. Comput. Integr. Manuf.*, vol. 27, no. 3, pp. 538-549, 2011.

[15] P. Vrba, P. Tichy, V. Mařík, K. H. Hall, R. J. Staron, F. P. Maturana, *et al.*, "Rockwell Automation's Holonic and Multiagent Control Systems Compendium," *IEEE Trans. Syst., Man, Cybernet. - Part C*, vol. 41, no. 1, pp. 14-30, Jan. 2011.

[16] P. Valckenaers, H. Hadeli, B. S. Germain, P. Verstraete, J. V. Belle, and H. V. Brussel, "From Intelligent Agents to Intelligent Beings," in *Holonic and Multi-Agent Systems for Manufacturing*. Berlin: Springer, 2007, pp. 17–26.

[17] P. Vrba, V. Mařík, P. Siano, P. Leitão, G. Zhabelova, V. Vyatkin, *et al.*, "A Review of Agent and Service-Oriented Concepts Applied to Intelligent Energy Systems," *IEEE Trans. Ind. Informat.*, vol. 10, no. 3, pp. 1890-1902, Aug. 2014.

[18] A. W. Colombo, F. Jammes, H. Smit, R. Harrison, J. L. M. Lastra, and I. M. Delamer, "Service-oriented architectures for collaborative automation," in *Proc. 31st Annual Conference of IEEE*, 2005, pp. 6.

[19] Y.F. Zhang, G. Zhang, Y. Liu, and D. Hu, "Research on services encapsulation and virtualization access model of machine for cloud manufacturing," *J. Intell. Manuf.*, to be published.

[20] J. L. M. Lastra and I. M. Delamer, "Semantic web services in factory automation: fundamental insights and research roadmap," *IEEE Trans. Ind. Informat.*, vol.2, no. 1, pp. 1 – 11, 2006.

[21] F. Tao, Y. LaiLi, L. Xu and L. Zhang, "FC-PACO-RM: A Parallel Method for Service Composition Optimal-Selection in Cloud Manufacturing System," *IEEE Trans. Ind. Informat.*, vol. 9, no. 4, pp. 2023-2033, 2013.

[22] R. Want, B. N. Schilit, and S. Jenson, "Enabling the Internet of Things," *Computer*, vol. 48, no. 1, pp. 28-35, Jan. 2015.

[23] G.Q. Huang, Y.F. Zhang, and P.Y. Jiang, "RFID-based wireless manufacturing for walking-worker assembly islands with fixed-position layouts," *Int. J. Robot. Comput. Integr. Manuf.*, vol. 23, no. 4, pp. 469-477, 2007.

[24] S. Makris, G. Michalos, and G. Chryssolouris, "RFID driven robotic assembly for random mix manufacturing," *Int. J. Robot. Comput. Integr. Manuf.*, vol. 28, no. 3, pp.359-365, June 2012.

[25] V.B. Ali, V.B. Reza, and H. Majid, "Structural Modeling of a RFID-enabled Reconfigurable Architecture for a Flexible Manufacturing System," in *Proc. Europ. Conf. SmartSysTech.*, Nuremberg, 2013, pp. 1-10.

[26] V. Vyatkin, "IEC 61499 as enabler of distributed and intelligent automation: State-of-the-art review," *IEEE Trans. Ind. Informat.*, vol.7, no.4, pp. 768-781, 2011.

[27] M. Sallinen, E. Strömmer, and A. Ylisaukko-oja, "Application Scenario for NFC: Mobile Tool for Industrial Worker," in *Proc. 2nd Int. Conf. Sensor Tech. Appl.*, Cap Esterel, 2008, pp. 586-591.

[28] *NSF 08-611: Cyber-Physical Systems (CPS)*, National Science Foundation, Arlington, VA, 2008.

[29] J. Lee, E. Lapira, B. Bagheri, and H. Kao, "Recent advances and trends in predictive manufacturing systems in big data environment," *Manuf. Letters*, vol. 1, no. 1, pp. 38-41, Oct. 2013.

[30] P. Leitão, J. Barbosa, M.-E.C. Papadopoulou, and I.S. Venieris, "Standardization in cyber-physical systems: The ARUM case," in *Proc. IEEE Int. Conf. Ind. Technol.*, Seville, 2015, pp. 2988-2993.

[31] A. W. Colombo, T. Bangemann, S. Karnouskos, J. Delsing, P. Stluka, R. Harrison, *et al.*, *Industrial Cloud-Based Cyber-Physical Systems*. Switzerland: Springer, 2014.

[32] L. Monostori, "Cyber-physical production systems: Roots, expectations and R&D challenges," *Procedia CIRP*, vol. 17, pp. 9-13, 2014.

[33] J. Lee, B. Bagheri, and H. Kao, "A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems," *Manuf. Letters*, vol. 3, pp. 18-23, 2015.

[34] B. Bagheri, S. Yang, H. Kao, and J. Lee, "Cyber-physical Systems Architecture for Self-Aware Machines in Industry 4.0 Environment," *IFAC PaperOnLine*, vol. 48, no. 3, pp. 1622-1627, 2015.

[35] P. Leitão, A.W. Colombo, and S. Karnouskos, "Industrial automation based on cyber-physical systems technologies: Prototype implementations and challenges," *Comput. in Ind.*, to be published.

[36] P. Leitão and F. Restivo, "ADACOR: a holonic architecture for agile and adaptive manufacturing control," *Comput. Ind.*, vol. 57, no. 2, pp.121-130, 2006.

[37] P. Valckenaers and T. Holvoet, "Fundamentals of Holonic Systems and their Implications for Self-adaptive and Self-organizing Systems," in *Proc. 2nd IEEE Int. Conf. Self-Adapt. Self-Organiz. Syst. Workshops*, Venice, 2008, pp. 168-173.

[38] J. Barbosa, P. Leitão, E. Adam, and D. Trentesaux, "Dynamic self-organization in holonic multi-agent manufacturing systems: The ADACOR evolution," *Comput. in Ind.*, vol. 66, pp. 99-111, 2015.

[39] J. V. Belle, P. Valckenaers, B. S. Germain, R. Bahtiar, and D. Cattrysse, "Bio-Inspired Coordination and Control in Self-Organizing Logistic Execution Systems," in *Proc. 9th IEEE Int. Conf. Ind. Informat.*, Lisbon, 2011, pp. 713-718.

[40] L. Monostori, P. Valckenaers, A. Dolgui, H. Panetto, M. Brdys, and B. C. Csáji, "Cooperative control in production and logistics," *Annu. Rev. Control*, vol. 39, pp. 12–29, 2015.

[41] S. Bussmann and K. Schild, "Self-organizing manufacturing control: an industrial application of agent technology," in *Proc. 4th Int. Conf. MultiAgent Syst.*, Boston, MA, 2000, pp. 87-94.

[42] P. Leitão and J. Barbosa, "Adaptive Scheduling based on Self-organized Holonic Swarm of Schedulers," in *Proc. 23rd IEEE Int. Symposium Ind. Electr.*, Istanbul, 2014, pp. 1706-1711.

[43] P. Vrba and V. Marík, "Capabilities of Dynamic Reconfiguration of Multiagent-Based Industrial Control Systems," *IEEE Trans. Syst., Man, Cybernet. - Part A*, vol. 40, no. 2, pp. 213-223, Mar. 2010.

[44] Auto-ID Labs, "About Us," 2015. [Online]. Available: http://autoidlabs.org.

[45] D.P. Bernardon, M. Sperandio, V.J. Garcia, L.N. Canha, A.R. Abaide, and E.F.B. Daza, "AHP Decision-Making Algorithm to Allocate Remotely Controlled Switches in Distribution Networks," *IEEE Trans. Pow. Deliv.*, vol. 26, no. 3, pp. 1844-1892, 2011.

[46] Y.F. Zhang, W.B. Wang, N.Q. Wu, and C. Qian, "IoT-Enabled Real-Time Production Performance Analysis and Exception Diagnosis Model," *IEEE Trans. Autom. Sci. Eng.*, vol. 13, no. 3, pp. 1318-1322, July 2016.