Journal of Manufacturing Systems 43 (2017) 305–315

Contents lists available at ScienceDirect

Journal of Manufacturing Systems

journal homepage: www.elsevier.com/locate/jmansys

Feature-based control and information framework for adaptive and distributed manufacturing in cyber physical systems

Göran Adamson [a,*], Lihui Wang [a,b], Philip Moore [a]

[a] Virtual Systems Research Centre, University of Skövde, 541 28 Skövde, Sweden
[b] Department of Production Engineering, Royal Institute of Technology, 100 44 Stockholm, Sweden

ARTICLE INFO

Article history:
Received 30 September 2016
Received in revised form
17 November 2016
Accepted 12 December 2016
Available online 23 December 2016

Keywords:
Cyber physical system
Adaptive manufacturing
Feature-based control
Feature-level capability model

ABSTRACT

Modern distributed manufacturing within Industry 4.0, supported by Cyber Physical Systems (CPSs), offers many promising capabilities regarding effective and flexible manufacturing, but there remain many challenges which may hinder its exploitation fully. One major issue is how to automatically control manufacturing equipment, e.g. industrial robots and CNC-machines, in an adaptive and effective manner. For collaborative sharing and use of distributed and networked manufacturing resources, a coherent, standardised approach for systemised planning and control at different manufacturing system levels and locations is a paramount prerequisite.

In this paper, the concept of feature-based manufacturing for adaptive equipment control and resource-task matching in distributed and collaborative CPS manufacturing environments is presented. The concept has a product perspective and builds on the combination of product manufacturing features and event-driven Function Blocks (FB) of the IEC 61499 standard. Distributed control is realised through the use of networked and smart FB decision modules, enabling the performance of collaborative run-time manufacturing activities according to actual manufacturing conditions. A feature-based information framework supporting the matching of manufacturing resources and tasks, as well as the feature-FB control concept, and a demonstration with a cyber-physical robot application, are presented.

1. Introduction

Interest is progressively growing in utilising new technological and organisational advancements, such as Cloud Manufacturing, Internet of Things (IoT), Cloud Computing (CC), Semantic Web, virtualisation, service-oriented technologies, and advanced manufacturing models, within the manufacturing shop-floor domain [1,2]. Through a closer relationship between the cyber computational space and the physical factory floor, enabling monitoring, synchronisation, decision-making and control, CPSs are transforming the manufacturing industry into its next generation, Industry 4.0 [3]. By combining many of these technologies, offering globally available information, data and services, embedded intelligence, as well as connected machine and sensory networks, the evolving CPS paradigm will move discrete manufacturing activities towards the seamless collaborative and distributed sharing of manufacturing resources [4–8]. However, the level of complexity regarding manufacturing planning and control will become significantly higher for these multi-collaborative and distributed manufacturing environments. Therefore, one of the major technological challenges within CPSs will be how to support scenarios in which dynamically configured groups of dispersed resources cooperate to complete joint manufacturing missions. Currently, many manufacturing systems are unable to reach their production goals as they are negatively influenced and restricted by a multitude of undesired variations and conditions. There are both external and internal system events contributing to this uncertainty, which result in decreased manufacturing productivity and product quality. These events often necessitate the revision of existing process plans, requiring the re-programming of manufacturing equipment, which is often both tedious and time-consuming, even for minor changes. The extent of this control code renewal depends on the effects and impact of the events, and since control programs are often created for dedicated equipment, program portability is usually very low. Most manufacturing planning tools, e.g., Computer-Aided Process Planning (CAPP), and control systems are not able to handle unforeseen changes efficiently. In addition, major limitations using traditional

* Corresponding author.
E-mail addresses: goran.adamson@his.se (G. Adamson), lihuiw@kth.se (L. Wang), p.r.moore@me.com (P. Moore).

http://dx.doi.org/10.1016/j.jmsy.2016.12.003

CAPP tools within distributed environments such as CPSs are: centralised decision-making, static system structure and off-line data handling [9]. This greatly limits their ability to adapt pre-made process plans to shop-floor run-time variations. In a distributed CPS environment, which may involve a disparate multitude of participating and cooperating resources and resource types, the quantity of unpredictable variations that could disrupt and cause negative impacts is significant. To cope with this uncertainty effectively, and to handle the volatile nature of manufacturing resources' conditions and availability, an adaptive and distributed planning and control approach is required, which can adapt to variations and be distributed to different resources and levels of a CPS [10]. Such a control approach would need to be able to react and respond to different kinds of uncertainties, by utilising distributed and dynamic decision-making. Effective control and execution will then require real-time monitoring of all involved manufacturing resources, in order to continually and accurately determine their dynamic availability and operational status.

An effective approach to solving many of the above mentioned manufacturing issues is to apply feature-based manufacturing. This approach, which stems from a product perspective, since it builds on the product manufacturing feature concept, is a viable and effective method for adaptive and distributed manufacturing. Feature-based manufacturing can be realised through manufacturing services. By using the concept of manufacturing services, in a similar manner to the use of services within cloud computing, manufacturing resources and capabilities can be provided in distributed environments, such as CPSs, in which device network capabilities, such as IoT, may enable access for controlling distributed manufacturing equipment. Through the use of feature-based and event-driven Function Blocks (FBs) as smart and distributable decision modules, run-time manufacturing operations in a distributed CPS environment may be planned and executed, in order to meet prevailing manufacturing conditions and requirements. Developed for distributed manufacturing equipment control, these modules can be combined to satisfy different levels of control needs, and ultimately realise the idea of Manufacturing-as-a-Service (MaaS) [11,12].

Matching a wide array of available resources to requested manufacturing tasks is also a major issue within CPSs [5]. In order to express the various capabilities of manufacturing resources, and to improve the quality and efficiency of resource discovery and intelligent matching to manufacturing tasks, a unified information framework is necessary. For a description of manufacturing tasks, a feature-enriched product data model is developed, and for manufacturing resources, a feature-level capability model. These two models facilitate the discovery and matching of available resources to requested tasks to be undertaken.

The purpose of this paper is twofold: 1) to describe an adaptive FB-based control approach for distributed manufacturing resources in CPSs and 2) to present the outline of an information framework supporting this approach. The control approach and the information framework both build on the concept of feature-based manufacturing, with the use of product manufacturing features (MfgFs) as the cornerstone. Control is achieved through event-driven IEC 61499 FBs in a two-level planning and control structure for adaptive control, integrated as a service in a proposed control structure for CPS distributed manufacturing management. Examples are given for robot assembly tasks but the control approach is applicable to manufacturing equipment more generally. The information framework proposed supports the retrieval and matching of manufacturing tasks to manufacturing resources' capabilities, since it builds on a shared product MfgF perspective.

The paper is structured as follows: while Section 2 describes the concept of product manufacturing features, Section 3 presents adaptive manufacturing equipment control through a combination of event-driven FBs and MfgFs. How the control approach can be implemented in a distributed CPS environment, and how the control system's integration into the CPS management functionality can be achieved, is described in Section 4. This is followed in Section 5 by an outline of a supporting feature information framework for the discovery and matching of manufacturing resources to manufacturing tasks. Section 6 presents a description of a system implementation for the MfgF-FB control approach in a CPS, while Section 7 specifies the authors' concluding remarks.

2. Product manufacturing features

Feature-based descriptors, and the concept of manufacturing features (MfgFs) have been used in both product design and manufacturing for many years to identify the relationships between product features and the manufacturing operations required for their creation. The feature technology "Design by Features" [13], involves the definition of product designs through combining the necessary manufacturing features to realise the product. The opposite process is performed in "Feature Recognition" [14], in which existing product designs are examined and evaluated, in order to identify the manufacturing features and operations required to create the product.

Different categories of MfgFs can be realised by identifying, classifying and mapping discrete low-level or atomic manufacturing operations required for the creation of unique product features. Different manufacturing domains require different features, e.g., machining features (MFs) for machining tasks and the creation of unique product designs, and assembly features (AFs) for product assembly tasks. A higher-level manufacturing task is made up of a sequence of different lower-level basic manufacturing operations, e.g., *Side*, *Face*, *Step* in machining [15,16] and *Place*, *Insert*, *Move* in assembly [17], which can all be defined as separate MfgFs.

Manufacturing resources may provide one or more MfgFs and a specific MfgF may be available from different resources and be used in different manufacturing tasks. Another important advantage is that the MfgF representation is independent of both the implementation scenario and manufacturing equipment. Consequently, portability and transferability is good since the functionality mapped into the FB will be performed equally well by different manufacturing resources. The use of MfgFs has many advantages in manufacturing, as they can be applied for different, and cooperative, purposes. Central in this work is a combined approach for the use of MfgFs, established from a product perspective. This provides great flexibility since MfgFs are used: to detail the product model and the manufacturing task, describe generic capabilities of manufacturing resources as well as support and simplify the planning, control, programming and run-time generation of control instructions for manufacturing resources (Section 3). Using MfgFs to describe both products and resources is an important property for discovering and matching manufacturing task requests to available manufacturing resources (Section 5). With regard to resources, MfgFs for different manufacturing domains are used to describe the resource's ability to complete unique manufacturing operations as product features, through combining and aggregating the functional capabilities and properties of manufacturing resources. With regard to products, MfgFs are used to describe how they are to be manufactured.

2.1. Product assembly features

For a more specific and detailed description of the use of MfgFs, a robotic assembly task is selected and the concept of AFs is described in more detail. Here, robotic assembly relates to the manipulation of components (parts and sub-assemblies) for the creation of

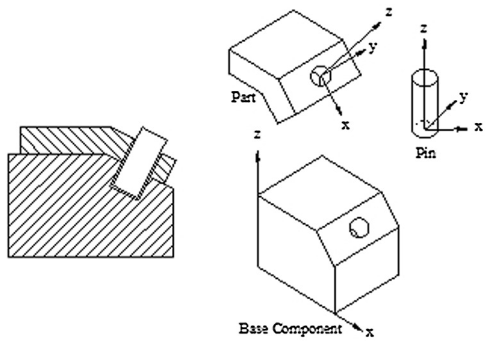

Fig. 1. Assembly task.

assembled products, and AFs encode the assembly method between connected components. As such, unique assembly operations e.g., *Insert*, *Screw* and *Place* are realised and mainly used for the coordination of parametrised motions. Except for motions, there are also various actions which need to be performed, including: signal processing, program logic, decision-making, and communication. Grouping these different AFs enables the coordination and control of a set of robot motions and actions. Furthermore, depending on required functionality, AFs are controlled by various input parameters, e.g., motion modes, target locations, velocities, tolerances, tool IDs, signal IDs and values, etc.

The combination of the two AFs *Pick* and *Insert* to realise picking and inserting of a pin into a hole, as part of a simple assembly task (Fig. 1), is shown in Fig. 2.

The two AFs are realised as a sequence of robot movements and actions, the movements following a series of p_i positions of the robot tool's Tool Center Point (TCP). P_is for robot path generation can be calculated once the locations (positions and orientations) of the hole and pin are known, as well as the trajectory of the inserting operation. These calculations are performed by FB embedded algorithms, as described in Section 3. Coordinates of objects to be assembled can also be received from sensor systems, e.g., a laser scanner.

In order to use MfgFs for adaptive manufacturing equipment control, an executional mechanism is required, which is capable of automatically generating the required equipment control instructions. By combining MfgFs with event-driven FBs of the IEC 61499 standard [18], a MfgF-FB control unit is created, as described in the following section.

3. Feature-based function block control

An adaptive control approach for manufacturing equipment, capable of instantly generating control in response to prevailing requirements and conditions, is described in the following two sections. It is constructed from a combination of event-driven International Electrotechnical Commission (IEC) 61499 FBs and MfgFs. The control approach is generic and can be used for different manufacturing applications, however, in the following sections it is described for robotic assembly applications.

3.1. IEC 61499 event-driven function blocks

The FB concept, and its use as a language in software to create PLC control for industrial equipment and processes, was first introduced as the graphical Function Block Diagram (FBD) language by the IEC 61131 standard in 1993. In the later IEC 61499 standard, the FB concept is further defined as an event-driven and component-oriented approach for distributed control systems and process measurement. In IEC 61499, FBs with different functionality and applicability are described as distributable software components which are able to encapsulate functionality. The standard also describes the development, implementation and use of these FBs. Manufacturing knowledge can be encapsulated as event-driven software components, which can be combined and distributed into an automation control network for the decentralised monitoring and control of manufacturing equipment. Some different types of FBs are defined, both for creating various levels of functionality and applications, and for handling communication. The Basic FB is constructed with the following main constituents, which are depicted in the AF-FB for inserting operations in Fig. 3:

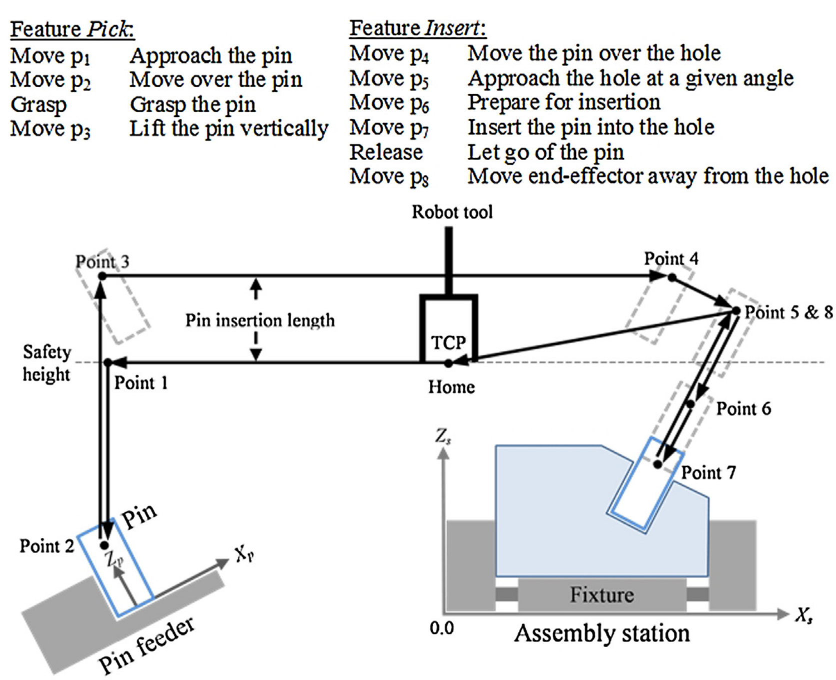

Fig. 2. Step-wise assembly sequence for features "Pick" and "Insert".

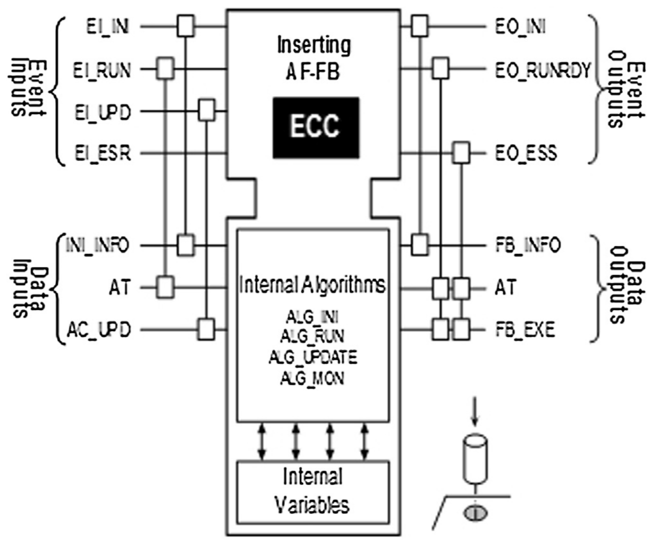

Fig. 3. Graphical definition of IEC 61499 Basic FB.

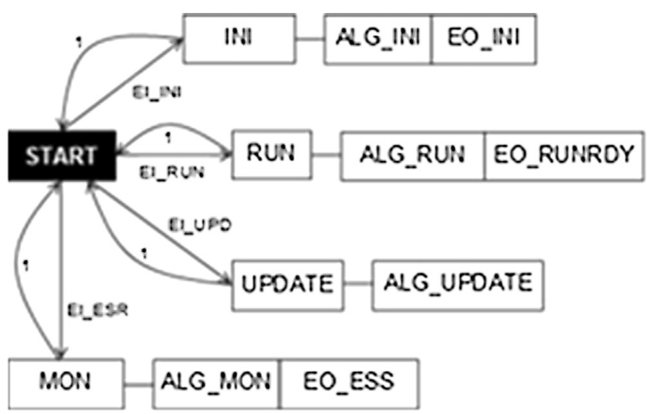

Fig. 4. FB execution control chart.

- inputs and outputs for events and data,
- Execution Control Chart (ECC): a finite state machine,
- algorithms for run-time generation of equipment control instructions,
- internal variables.

The FB functionality is encapsulated into internal algorithms, the execution of which are triggered by incoming input events. When algorithms execute, they read and use available input data, as well as internally available data, to dynamically create output data. After execution, output events are used to announce the completion and availability of the generated output data. Following this procedure, these FBs can be regarded as run-time decision-making modules. The overall FB behaviour is determined by the finite state Execution Control Chart (ECC), which controls the scheduling and execution of the internal algorithms as input events are received. The ECC in Fig. 4 defines the operational behaviour of the basic FB in Fig. 3.

A practical and effective use of this event-driven approach is to define output data as required robot control code. The control code will then be generated dynamically and automatically, in a truly adaptive manner, according to the real-time conditions of the manufacturing situation and changing or varying requirements. The traditional generation of robot control code is rigid in both control program structure and content, since it relies on sending pre-determined control instructions to the robot controller without regard to actual run-time conditions. As such, it is not able to adapt to variations.

The practical implementation and use of IEC 61499 FB-based control systems has been described for a number of different manufacturing applications, of which most have been focused at low-level device control [19].

3.2. Adaptive manufacturing equipment control

To practically use MfgFs for the adaptive control of manufacturing applications, an implementation approach for the planning and execution of MfgFs is necessary. By combining the distributed run-time decision-making properties of event-driven IEC 61499 FBs with the manufacturing "know-how" of MfgFs, it is possible to create the adaptive manufacturing control unit MfgF-FB. Furthermore, by mapping the MfgFs to the algorithms of the FBs, this unit provides the encapsulation of manufacturing functionality, as well as data transfer, the event-driven process and execution control. Through the mapping of individual MfgFs to FBs, unique functionality is embedded in the FB algorithms, creating function-specific MfgF-FBs. The algorithms are thus designed to create the desired MfgF functionality, therefore, when triggered, they will dynamically generate the manufacturing control instructions required to perform a basic manufacturing operation, e.g., *Insert* for a robotic assembly task. This assembly scenario is demonstrated in Fig. 2, in which the targets and the trajectory of the insert operation are calculated by the embedded algorithms of the AF-FB. Creating robot assembly functionality includes setting typical property parameters, such as robot move mode, target frames and paths, TCP speed, level of accuracy, tool or work object to use, etc. These parameters are then available to the FB as data inputs, and the generated control instructions are published as data outputs. Demonstrating this concept, a combination of an event-driven Basic FB with the AF "Insert", instantiating an AF-FB, is depicted in Fig. 3, with its associated ECC shown in Fig. 4. This AF-FB holds a set of algorithms for the generation of run-time adapted, robot control instructions to realise the specific assembly operation *Insert*. The construct of the control approach provides the ability to dynamically adapt to variations in assembly tasks, such as: changes in product designs, changes in assembly component locations, performing operations with a different robot tool, completing the operations of another robot, etc. In contrast to traditional process planning and programming of manufacturing equipment, at an early stage in the product development process, the required robot control code is here generated instantly. This adaptive behaviour originates from the AF-FB's ability to adjust output data to actual, run-time conditions. The FB behaviour is defined in the finite state machine of the ECC, following the sequence of: read the values of data inputs, sequencing and execution of the algorithms, and publishing control instructions as data outputs. By continuously monitoring the status of the robot and its working environment, the data inputs can be fed with the real-time information of the ongoing assembly task. When incoming input events trigger the execution of algorithms, decision-making can be performed on the basis of the prevailing status of the system. This control approach provides great flexibility since the functionality mapped into the FB may be able to generate the same manufacturing result when performed by different manufacturing resources. For this, different algorithms are created and included in the FB, each customised to match specified robots, tools and assembly scenarios. A data input is then used to read the ID of the robot, for the selection of the corresponding algorithms. However, the level of manufacturing adaptability is not unrestricted, but bounded by the manufacturing system's ability to adapt to changes through the functionality of the control system. Architectural hardware perspectives or different physical reconfigurations of manufacturing equipment, such as robots or CNC-machines, are not considered

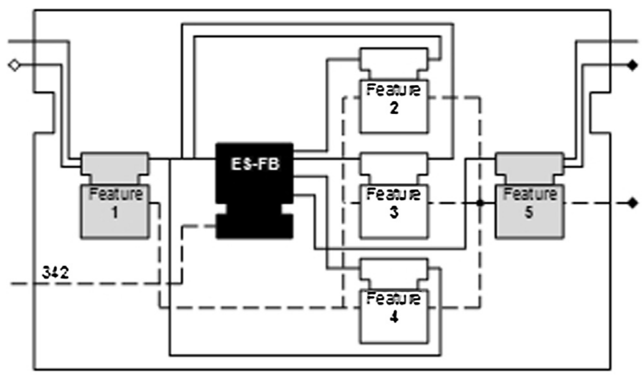

Fig. 5. IEC 61499 CFB control application.

as such in this research. The level of control adaptability is fully dependent on the sophistication of the construction of the ECC and the FB algorithms, as well as the scope of available real-time information to be accessed and processed, in order to generate control. Control approaches may range from reading robot target information from a single sensor, to be input to AF-FB algorithms generating robot move instructions for an *Insert* operation, to the complex processing of information, by algorithm implemented AI technologies, from a network of sensors, actuators and controllers, in order to generate optimal control for a manufacturing task.

Libraries of MfgF-FBs can be formed which can be re-used to create and automate manufacturing tasks. By selecting and combining appropriate MfgF-FBs into a control network, e.g., an IEC 61499 Composite FB (CFB), Fig. 5, higher levels of manufacturing functionality can easily be formed. Creating control for manufacturing applications is thus rationalised to the arrangement of predefined MfgF-FBs into a network, with event and data interfaces interconnected.

The networked MfgF-FBs in a CFB are executed in a sequence: one by one from left to right, propagating the control information between each other in implementing the desired control functionalities in accomplishing the actual manufacturing task. The MfgF-FB control concept in this way simplifies and accelerates the equipment programming effort. In addition, the control creation procedure could be automated to a large extent, if it were combined with a manufacturing feature recognition system.

Combining MfgFs with IEC 61499 FBs, to create adaptive control solutions, has been successfully demonstrated for different manufacturing scenarios [20,21]. A CPS demonstrator system has been produced whereby verification of this control approach in adaptive robotic assembly has been undertaken; the demonstrator is described in Section 6. In the following section, the implementation of the demonstrator in a distributed CPS environment is described.

4. Adaptive and distributed control in cyber physical systems

Since the introduction of CC, with models that offer software, infrastructure, platforms and applications in the form of services, cloud technology has been extended to the manufacturing domain [22]. Various distributed manufacturing systems have been presented [23–25], offering on-demand and scalable manufacturing services over the Internet [26], from a shared pool of distributed manufacturing resources. These resources range from facilities, work-cells, machine tools and robots, to capabilities and software. The use of cloud manufacturing services in CPSs is fundamental for distributed manufacturing and facilitates collaborative manufacturing missions, both within and between CPSs.

The MfgF-FB control approach, described in this paper, has previously been demonstrated in the local shop-floor domain, for realising adaptive control of robot applications [21]. However, the distributed features of IEC 61499 FBs are an important property for their use in distributed manufacturing environments, such as CPSs. Besides different FB types, the standard also defines the interaction and communication between distributed FBs. This enables networked MfgF-FBs to be integrated in a CPS platform for the planning and execution of manufacturing tasks at different system control levels. The following section describes the system structure, including modules, for distributed manufacturing management in CPS.

4.1. CPS distributed manufacturing management

The management of all CPS service activities is performed by the CPS Service Management (CSM) module (Fig. 6), which is responsible for resource discovery and matching, as well as dynamically coordinating manufacturing planning and execution control of distributed manufacturing resources. To perform a manufacturing task, a variety of services in combinations are possible. One approach is to use a single, high-level service which has all the necessary manufacturing operations to be performed by one service provider, e.g., Manufacturing/Assembly-as-a-Service (MaaS/AaaS), in a resource-service many-to-one mapping. Another approach is to use a combination of services of varying complexity and scope performed by different providers, e.g., Robot Control as-a-Service (RCaaS) together with Robot Hardware-as-a-Service (RHaaS), a combination of many one-to-one mappings. (RHaaS suggests that a provider offers the use of a robot system, which could originate from dedicated assembly/robot centres or providers wanting to sell spare robot capacity.)

In this example, the combination-of-services approach is described to emphasise the multiple resource sharing and collaborative perspectives of CPSs. In the following description of the CPS Service Management, three service providers participate to jointly deliver the functionality required to complete an assembly task request. It is assumed that a robot control provider supplies the robot control capability as RCaaS, and two providers supply the robot hardware as RHaaS. The RCaaS is the instantiation of the MfgF-FBs control approach as a distributable service, and its executional control unit is a CFB. In this example, the CFB is an assembly process plan (APP) that includes the necessary AF-FBs in the correct sequence to perform the assembly task.

RCaaS includes 5 cooperating modules, each performing different tasks (Fig. 6):

- Supervisory CPS Planning (SCP),
- Feature Identification and Sequencing (FIS),
- AF-FB Library (AFL),
- CPS Robotics Control (CRC),
- Local Operation Planning (LOP).

The control procedure is initiated upon receipt of an assembly task request. This is analysed by the CSM, which selects and combines the necessary services, and then triggers RCaaS by sending the compiled assembly task information. This launches a sequence of activities within RCaaS, which performs a two-level FB-enabled planning procedure for the generation of an AF-FB based control structure, the APP. In this process, generic and robot-specific information are separated into Supervisory CPS Planning and Local Operation Planning, to enable efficient and smart decision-making. The process includes the following steps and activities:

1 SCP is performed once for the assembly task requested:

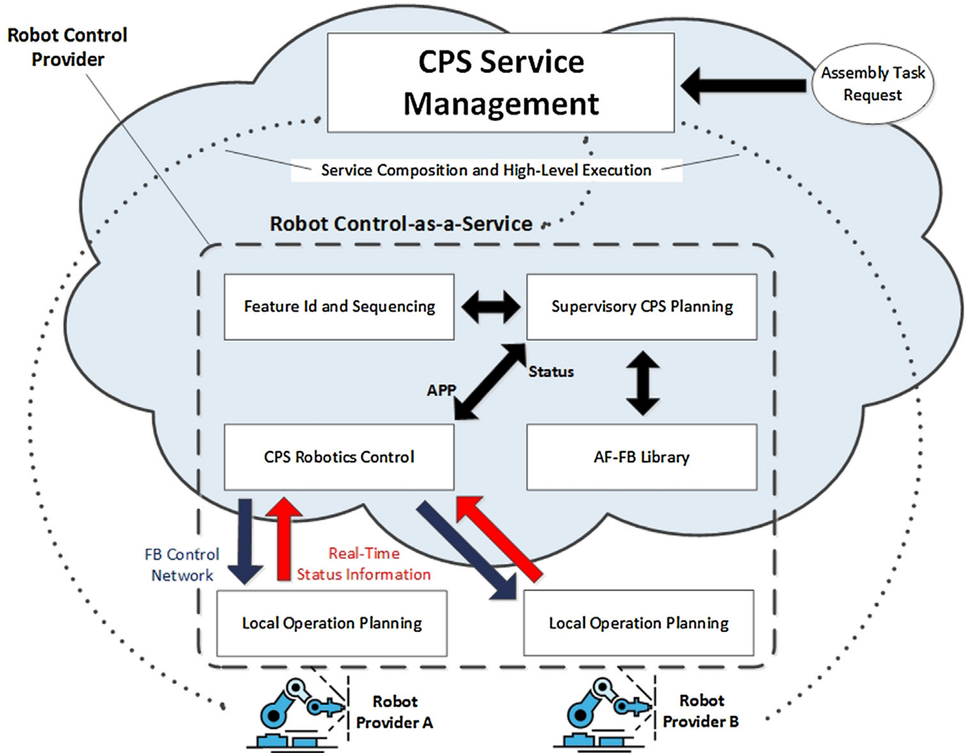

Fig. 6. CPS distributed manufacturing management.

- Assembly features are identified and sequenced by the "Feature Id and Sequencing" module.
- By using pre-defined AF-FBs from the AF-FB Library module, the SCP creates an APP by mapping necessary AF-FBs into a sequenced network of AF-FBs.

The APP created only contains necessary AF-FBs and their critical assembly sequence. This entails that it is generic and not tied to a specific robot. It can therefore be reused as well as ported to alternative robotic systems. (It is assumed that the activities of the FIS module are performed and input to the SCP module. These activities are beyond the scope of this research).

2 The CRC module receives the APP from the SCP and has the following responsibilities:

- Distribute AF-FB control structures to the selected robot providers,
- Coordinate AF-FBs between different providers,
- Coordinate AF-FBs operation planning locally at each robot provider,
- Dynamic scheduling of resources and activities included,
- Perform robot initialisations,
- Perform FB execution control (start, stop, pause, resume, etc.)
- Monitor local robot execution, status and feedback to SCP.

3 Robot-level operation planning and execution:

- The generic APP is detailed through robot-level operation planning, since the embedded algorithms read their data inputs.
- Since LOP executes AF-FBs one by one, robot-specific control instructions are created at run-time through controller-level decision-making.

This control approach provides a high degree of adaptability to changes, by detailing the generic APP at LOP. Planning and execution is thus performed on demand, based on run-time information.

The APP is a generic AF-FB control structure for a distributed CPS environment, realised as one or more CFBs, and is instantiated and detailed at run-time, locally at each provider. The SCP module handles global variations, e.g. product design changes, provider or resource changes, while local run-time shop-floor variations, e.g., robot failures, fixture/tool changes are primarily handled by the LOP or the CRC.

5. Feature-based information framework

To support the FB-based control approach described in the preceding sections, and to be able to match manufacturing requests to the capabilities of the resources, reference information models of these are required. Building on the concept of product MfgFs, a unified information framework describing manufacturing tasks and the capabilities of manufacturing resources from a shared product feature perspective is outlined. For a description of manufacturing tasks, a feature-enriched product data model is presented, and a feature-level capability model is introduced for the manufacturing resources' capabilities. Together, these two models facilitate manufacturing resource discovery and the matching of resources to requested manufacturing tasks. For structured and formalised semantic model representations, implementation through ontologies and Web Services is described.

5.1. Feature-enriched product data model

A product data model expressing necessary manufacturing processes and requirements for product creation would greatly facilitate the automated discovery and matching to possible manufacturing resources. This is especially true in distributed manufacturing environments in which the complete manufacturing

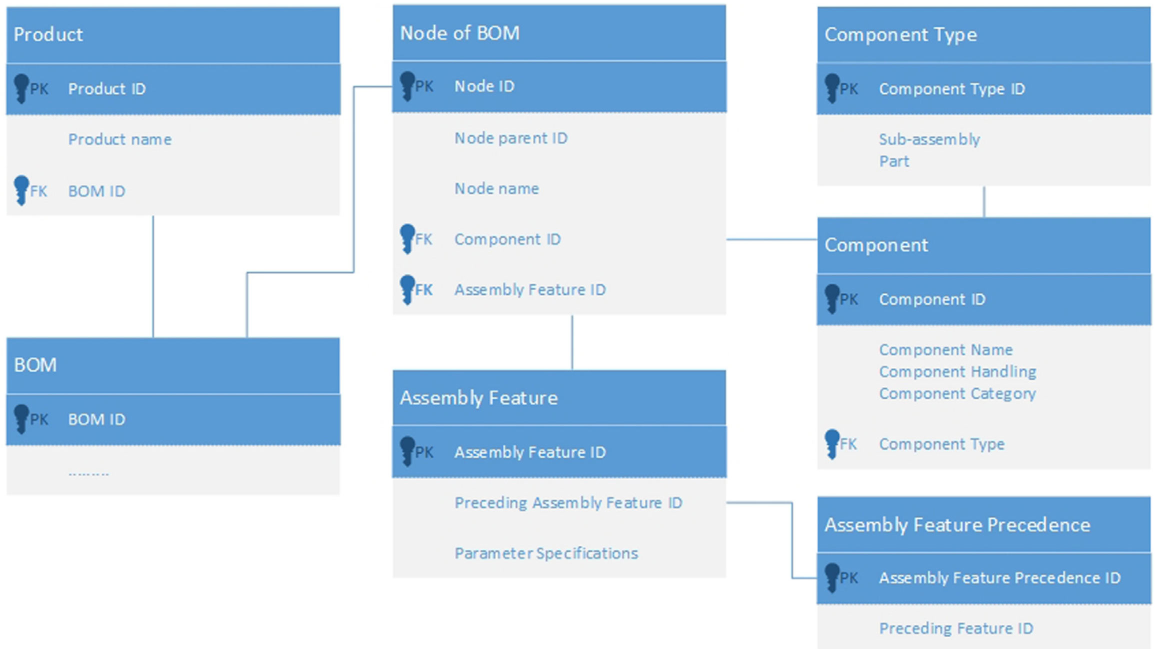

Fig. 7. Assembly feature-enriched product data model.

information may not be locally available. Following the feature technology concept "Design by Features" [13], the product data model can be enriched with the necessary information regarding MfgFs to manufacture the product. This method of further detailing the product model is the starting point for a combined approach for adaptive MfgF-FB enabled manufacturing equipment control and effective resource-request matching, permeated from a product perspective. A manufacturing task request implies product manufacturing within one or more manufacturing domains. An assembly task request thus relates to product assembly. For this request, the required AFs and their order of precedence need to be specified. Assembly-related parameters and constraints also need to be included in association with each AF. This specification is preferably performed during the product design stage, however, feature extraction methods can also be used to add this information to the product data model at a later stage [27].

In Fig. 7, a feature-enriched product data model is shown. To simplify understanding, the model is limited to focusing on the data entries mainly associated with the feature concept. The "Parameter specifications" field can hold a multitude of data entries, representing various feature parameters and constraints, such as, motion type, target frame, velocity, reach, tolerance, weight, force, etc. Some of these may be optional, as not all are relevant to every MfgF. In the matching procedure between manufacturing tasks and resources, both feature types and their associated parameters will define the search outcome.

5.2. Feature-level manufacturing resource capability model

5.2.1. Manufacturing resource capability

With some moves to globalisation of manufacturing enterprises and the associated distribution of production systems, the use of dynamic supply networks that can respond to changes quickly and are cost-efficient suggests that traditional use of rigid supply chains is no longer the best solution. A key factor in collaborative manufacturing missions is communicating the capabilities of manufacturing resources, e.g., functionality, production processes, capacity, cost, quality, etc. between manufacturing participants. For the creation of agile supply chains, it is vital that accurate and interoperable manufacturing capability information of all supply chain resources is available. A reference model describing manufacturing resources' capabilities is necessary, since they constitute the fundamental elements that can collaboratively satisfy manufacturing tasks.

A major problem currently is that capability information models are often developed as proprietary models of the resource providers, leading to many accuracy and interoperability issues [28–30]. These models typically use their own semantic representation and structure, often including unstructured, as well as conflicting and ambiguous data. Differing representations of production requirements, as well as different taxonomies for categorising manufacturing resources, are commonly used. As a result, various interoperability issues arise related to resource sharing, leading to incomplete or sub-optimised manufacturing solutions, for which the most suitable resources are sometimes not used. The largest obstacle to agile assembly and effective manufacturing supply chains in distributed CPS environments is the absence of an agreed reference capability model. The often intricate process of matching a manufacturing task to manufacturing resources is performed manually, and therefore tends to be inefficient, since the available manufacturing data is superfluous and its interrelationships complex. Lack of agreed interpretation of terms used in the manufacturing domain further magnifies this problem. Therefore, an information model for the description of resources and their capabilities is required, as well as a language for its implementation.

5.2.2. Manufacturing resource capability models

Models for the description of manufacturing capability often have different perspectives and cover different levels of capability. In this research, the most interesting capability, i.e., the functional capability of manufacturing equipment, describes what manufacturing operations the equipment is able to perform. Many other aspects of capability are, of course, also of interest in the matching of tasks and resources. Resource properties, such as cost, quality, availability, delivery times, resource location, consumer ratings, etc., may also be expressed in manufacturing capability models. The functional capability answers the question of "what" the resource can perform in regard to production capacity, while other capability descriptions answer the question of "how", with

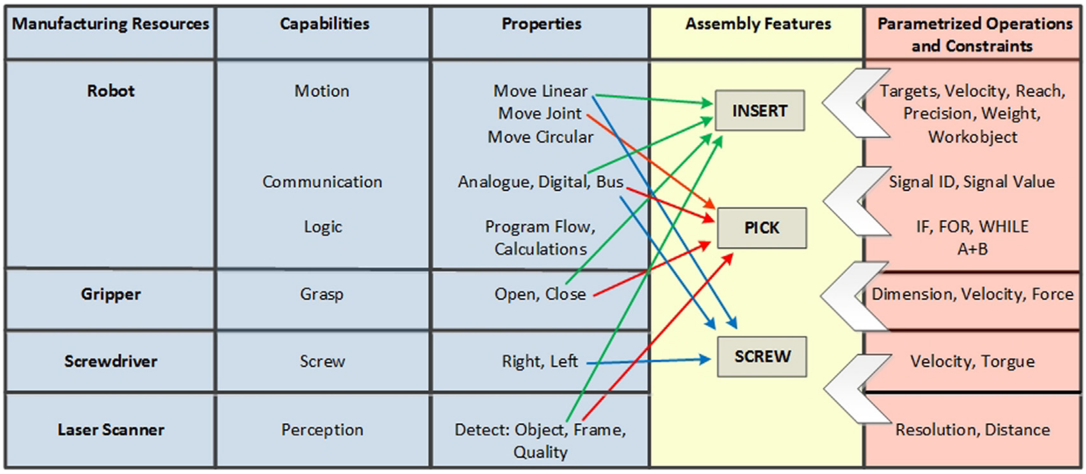

Fig. 8. Feature-level assembly resource capability model.

respect to resource properties such as the above mentioned. As such, capability models often include more than the functional resource properties, and describe both static and dynamic information. A manufacturing machine model with four different capability attributes is presented by [31]. The model comprises the following attributes: *Basic*, *Function*, *Real-time status*, and *Evaluation*. A cell level model for more complex manufacturing tasks is also described and features: *Description*, *Status*, *Evaluation*, and *Resource* information. The use of condition information for dynamic modeling of manufacturing equipment capability is described by [28]. The unified description model presented includes three fundamental ontologies: for basic information, manufacturing function, and manufacturing processes. The basic information and manufacturing function describes static information and characteristics, while the manufacturing processes describe the manufacturing conditions and the dynamic capabilities. On the basis of the mapping relationship between the real-time condition data and the model of the manufacturing equipment capability ontology, the knowledge structure is able to update in real-time. A 5-level manufacturing system capability model supported by the formal ontology Manufacturing Service Description Language (MSDL) is presented by [32]. The model comprises: *Supplier*, *Shop*, *Machine*, *Device*, and *Process*. Each level is described by an ontological model which enables the capabilities of the higher level entities, e.g., machine tools and shop floor, to be inferred through aggregation of device-level capabilities. They also describe manufacturing capabilities in the following dimensions: *Technological*, *Operational*, *Geometric*, *Quality*, *Relational*, and *Stochastic*.

A review of some proposed capability models reveals that models often differ in information scope, the representation of static and dynamic information, and the number of capability levels, often referred to as granularity levels. It is obvious that the lack of a unified standard for manufacturing vocabulary and definitions, in combination with the wide range of manufacturing resources and their applicability in different processes and industrial scenarios, is reflected in disparate capability models. Discussions regarding appropriate granularity levels for the most effective task-resource matchings are scarce.

5.2.3. *Feature-level capability model*

The level of granularity in manufacturing capability descriptions plays an important role from the perspective of discovering, combining and matching of resources in response to a manufacturing task. In the wide span of various levels of capability descriptions' complexity, two extremes exist: the low-level discrete functional capability of a single manufacturing resource, e.g., the moving of a robot joint, and the high-level capability of an enterprise, e.g., the manufacturing of a complete car engine. Between these two extremes, a variety of approaches for representing manufacturing capability exists, differing in complexity and number of levels. In some respects, there are advantages with this partitioning of manufacturing capability into different levels, but it may also be challenged. Using a low-level capability description approach enables identification of numerous matching resources for a manufacturing task, and from the perspective of resource allocation flexibility, smaller granularity levels would be preferred. Searching for a machining resource to perform some rudimentary machining operations would then result in an immense set of possible resources. In contrast, the search for a car engine manufacturing resource would probably return a fairly restricted set of resources. A manufacturing feature-level capability partitioning could facilitate and speed up the search for suitable resources.

The number of matching resources for a unique manufacturing request would be drastically reduced if a manufacturing resources' many functional capabilities were matched to a limited number of different pre-defined product MfgFs. The concept of manufacturing feature-level capability can also be used to describe the capabilities of combinations of resources and their functionalities, e.g., assembly operations realised through the combination of a robot and a gripper tool.

From a resource perspective, a MfgF is a combination of a manufacturing resources' capabilities, and describes the ability to complete a unique manufacturing product feature, within a certain manufacturing domain (machining, assembly, etc.). A feature-level capability model for the robotic assembly domain is presented in Fig. 8. Typical manufacturing properties and atomic manufacturing operations for matching against different categories of MfgFs can be identified from conceptual capabilities.

Four manufacturing resources and their associated capabilities, described by functional properties and their defining parameters and constraints are defined at a conceptual AF capability level. (For reasons of clarity, all possible relations have not been graphically represented, and only a subset of available properties are included.). This modeling approach represents capability as MfgFs defined by specific parameters and constraints. Here, a manufacturing operation is the detailing of a property with related parameters within existing constraints. The feature(s) related to a specific resource and its properties can be explicitly defined by the resource provider or implicitly inferred from an ontological description [33]. For defining features, standardised feature templates can be used, describing the required set of unique operations for each feature, e.g. *Insert*, Fig. 2. Using this capability model and granularity level,

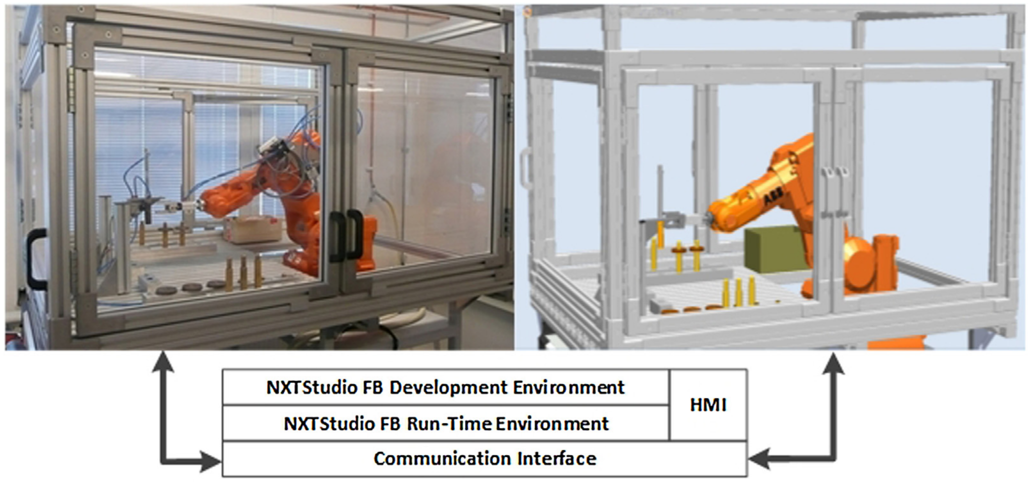

Fig. 9. CPS Robot Demonstrator System.

a quicker assembly task request and resource matching can be performed at an AF-level. As a first step in the matching procedure, task decomposition and parameter-feature identification are used, followed by a feature-level resource search. The feature-enriched product data model will be the primary defining source for setting the search conditions in this process. When available resources for the requested manufacturing task have been found, a second search among these can be performed, now focusing on resource capabilities other than the functional ones, e.g., requirements regarding cost, delivery, quality, etc. The outcome would provide the set of resources which best matches the manufacturing request, and from which the final resource selection can be made.

For effective search and inference capabilities, a multi-layer ontology is an effective language to implement the proposed information framework, and is outlined in the next section.

5.3. Semantic ontology information representation

A CPS environment may comprise of a large variety of different distributed and heterogeneous manufacturing resources, each of which may hold unique manufacturing capabilities. To improve the quality and efficiency of discovery and matching, Semantic Web and ontology technologies can be used to describe both resources and manufacturing tasks. Ontologies provide a solid mechanism to represent real-world objects and are particularly important for providing computers with explicit definitions of relevant domain knowledge representations, as well as capabilities for reviewing knowledge, in order to infer additional information [34–36].

An alternative method for establishing the proposed resource and product information models, in case they do not already exist, is to use an ontology-based information representation and retrieval approach, as described by [33]. Product and resource data is often handled by heterogeneous information systems and stored in different systems at the facilities of providers and those requesting a task. Using an ontology-based approach, the contents of different data sources can be retrieved and represented semantically. With ontology extraction, information regarding resources or products can be structured and formalised into local ontologies. In addition, through the use of semantic augmentation and querying, as well as matching to global manufacturing ontologies, new information may be inferred. It is not feasible to develop a single, complete CPS ontology, since different companies have different requirements. A multi-layer ontology structure for the creation of dedicated ontologies for different levels of manufacturing applications and situations is therefore realistic. When developing an ontology that covers all aspects of a certain manufacturing domain, the major problem is to obtain everybody's approval of a shared understanding, interpretation and denomination of all objects, entities and their relationships. In reality, there are often different interpretations and definitions for many typical and common manufacturing concepts. Ontology development is therefore not really an information technology problem, but rather more of a domain consensus issue. To support the implementation of the proposed information models, a multi-layer ontology structure, including ontologies for manufacturing resources, products and features, is required.

6. System concept demonstration

A CPS demonstrator system has been developed to demonstrate the feature-based FB-control approach. The operation planning and execution control has been implemented and verified in a robot cell (Minicell) and its virtual copy (Fig. 9).

Minicell is a robot cell equipped with an ABB 140-robot with a double gripper tool for assembling different variants of washer and shaft components. The purpose of the demonstrator system was to evaluate the ability to verify the correct robotic assembly behaviour in a virtual environment, before the AF-FB control structure was used to control the real robot. A set of virtual sensors was used in the virtual robot cell to test for adaptive system behaviour. These sensors detected the varying locations of shafts and washers to be assembled, which could appear at random locations within the working environment of the robot cell. The component location information was used as a control instruction robot-target parameter, and relayed via the FB data inputs to the FB algorithms, which, when triggered by appropriate input events, could dynamically create the correct control instructions for the robot system. After verification of the correct assembly functionality in the virtual system, the control structure could be used to control the real robot locally or in a distributed environment over an Ethernet connection. This is possible since the same RAPID (ABBs robot language) control instructions are used for both virtual and real ABB robot controllers. A run-time generated control such as this enables assembly plans to be dynamically adjusted in order to handle different variations. This means that even though the assembly task is started, components for assembly can be moved away from their original locations, and the robot system will still be able to successfully complete the assembly task. Reconfiguration or exchange of AF-FBs will adjust the control system's behaviour to suit other assembly scenarios. To completely automate the control, the required AF-FBs need to be

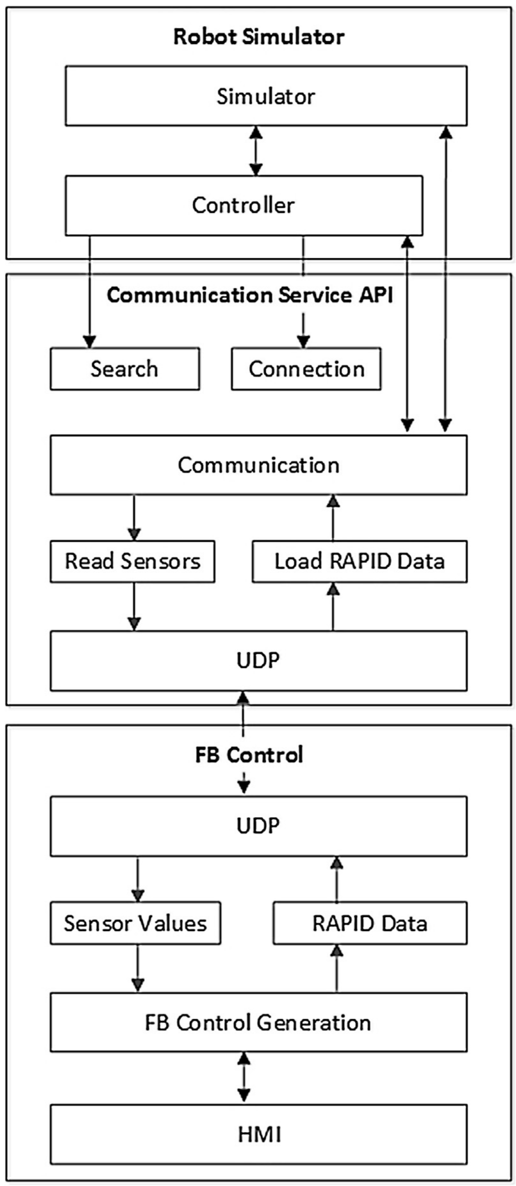

Fig. 10. FB control system for robot CPS.

identified and properly sequenced into an APP, and input to the system. Thus far, a major obstacle with using FBs for control generation is that controllers of most manufacturing equipment are legacy controllers, which means they cannot read and execute these FBs. In the demonstrator system, a front-end computer was used to overcome this obstacle.

The control system is constructed from three cooperating modules (Fig. 10): *FB Control*, *Communication Service* and *Robot Simulator*.

6.1. FB control

Through an interface (HMI) with a set of basic commands for controlling the application, it is possible to search for available robot systems, real or virtual, either directly connected to the front-end computer or available on a network, and to choose which one to connect to. Different robotic assembly tasks can be selected and started. The commercial software nxtSTUDIO, offering both FB development and run-time environments, was used for AF-FB and FB control structure development, as well as for assembly execution control and HMI construction.

6.2. Communication service

An ABB Application Programming Interface (API) was developed (Visual Studio/C#) to establish a two-way robot communication interface between the FB run-time environment and the robot controller. It established a connection for reading robot system status and sensor values, and transferring RAPID instructions to the controller.

6.3. Robot simulator

The virtual robot cell was created in ABBs offline programming and simulation software RobotStudio (RS). It is a virtual copy of the real Minicell, with virtual robot, tool and assembly components. RS Smart Component sensors are used to detect component locations.

7. Conclusions

The formalised control of manufacturing equipment operations within distributed CPS environments is a technological and management issue which needs to be addressed, as support for this issue is weak, especially if traditional process planning methods are used. The proposed MfgF-FB control approach has many promising benefits for use within CPS environments. One major advantage, compared to traditional control, is that the required control is created at run-time, by a group of FB-embedded algorithms, triggered to execute at the right time by events in the manufacturing environment. No pre-determined equipment specific control programs need to be prepared and sent in advance to the equipment controllers. Other important advantages are the distributable property of IEC 61499 FBs, as well as portability of functionality between different manufacturing resources. The results of the CPS demonstrator system have verified the adaptive behaviour of the MfgF-FB control approach. Robot control for adapting to specific changes in the assembly scenario is dynamically generated and programming-free. Compared to manually generating the correct control instructions when a change occurs, the amount of time saved is difficult to measure or estimate, but may be considered significant.

Following the IEC 61499 standard, control applications of varying levels of complexity and sophistication may be created for distributed control in CPSs. Being event-driven FBs, their dynamic behaviour may be controlled, and the ability to implement the performance of the desired functionality, through algorithms/programs, makes them very versatile. In contrast to Basic and Composite FBs, which are intended to be developed by an application developer, FBs for interfacing manufacturing equipment (Service Interface FBs) need to be provided by vendors of the corresponding equipment (controllers, fieldbus, remote input/output modules, intelligent sensors, etc.). The biggest obstacle to taking full advantage of the MfgF-FB control approach in CPSs is the possible presence of proprietary controlled manufacturing equipment, using controllers with native control languages. These cannot be directly controlled by FBs, so intermediate solutions such as front-end computers performing code translations and FB generation of machine-specific control instructions would have to be used. To take full advantage, controllers that can interface, interpret and execute FBs directly are ideally required.

The availability of FB-based equipment controllers, with predefined sets of parametrised feature FBs would close the loop of feature-based manufacturing. The feature-based information framework could then retrieve the appropriate resource feature information directly from the FB controller specifications, further facilitating discovery, matching and control.

Conflict of interests

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

Acknowledgement

The authors are thankful to the Swedish Knowledge Foundation [grant number 20130303] for providing support to accomplish the research work presented.

References

[1] Ren L, et al. Cloud manufacturing: from concept to practice. Enterp Inf Syst 2013;9(2):186–209.
[2] Ren L, et al. Cloud manufacturing: key characteristics and applications. Int J Comput Integr Manuf 2014:1–15.
[3] Zhou K, Taigang L, Lifeng Z. Industry 4.0: towards future industrial opportunities and challenges. 2015 12th international conference on fuzzy systems and knowledge discovery (FSKD) 2015.
[4] Wang L, Törngren M, Onori M. Current status and advancement of cyber-physical systems in manufacturing. J Manuf Syst 2015;37:517–27.
[5] Monostori L. Cyber-physical production systems: roots, expectations and R&D challenges. Procedia CIRP 2014;17:9–13.
[6] Harrison R, Vera D, Ahmad B. Engineering methods and tools for cyber–physical automation systems. Proc IEEE 2016;104(5):973–85.
[7] Guan X, et al. A comprehensive overview of cyber-physical systems: from perspective of feedback system. IEEE/CAA J Autom Sin 2016;3(1):1–14.
[8] Hu F, et al. Robust cyber–physical systems: concept, models, and implementation. Future Gener Comput Syst 2016;56:449–75.
[9] Xu X, Wang L, Newman ST. Computer-aided process planning–a critical review of recent developments and future trends. Int J Comput Integr Manuf 2011;24(1):1–31.
[10] Monostori L, et al. Towards adaptive and digital manufacturing. Annu Rev Control 2010;34(1):118–28.
[11] Herterich MM, Uebernickel F, Brenner W. The impact of cyber-physical systems on industrial services in manufacturing. Procedia CIRP 2015;30:323–8.
[12] Wu L, Yang C. A solution of manufacturing resources sharing in cloud computing environment. In: Cooperative design, visualization, and engineering. Springer; 2010. p. 247–52.
[13] Shah JJ, Rogers M. Expert form feature modelling shell. Comput-Aided Des 1988;20(9):515–24.
[14] Van Holland W, Bronsvoort WF. Assembly features in modeling and planning. Rob Comput-Integr Manuf 2000;16(4):277–94.
[15] Wang L, Jin W, Feng H-Y. Embedding machining features in function blocks for distributed process planning. Int J Comput Integr Manuf 2006;19(5):443–52.
[16] Tapoglou N, et al. Cloud-based platform for optimal machining parameter selection based on function blocks and real-time monitoring. J Manuf Sci Eng 2015;137(4):040909.
[17] Wang L, Keshavarzmanesh S, Feng H-Y. A function block based approach for increasing adaptability of assembly planning and control. Int J Prod Res 2011;49(16):4903–24.
[18] Iec I. 61499-1, function blocks-part 1: architecture. International Standard, 1, first edition Geneva: International Electrotechnical Commission; 2005. p. 2005.
[19] Vyatkin V. IEC 61499 as enabler of distributed and intelligent automation: state-of-the-art review. IEEE Trans Ind Inf 2011;7(4):768–81.
[20] Adamson G, et al. Adaptive assembly feature based function block control of robotic assembly operations. In: 13th Mechatronics forum international conference. 2012 [TRAUNER Verlag].
[21] Adamson G, et al. Function block approach for adaptive robotic control in virtual and real environments. In: Proceedings of the 14th mechatronics forum international conference. 2014.
[22] Wu D, et al. Cloud-based design and manufacturing: a new paradigm in digital manufacturing and design innovation. Comput-Aided Des 2015;59:1–14.
[23] Xu X. From cloud computing to cloud manufacturing. Rob Comput-Integr Manuf 2012;28(1):75–86.
[24] Tao F, et al. Cloud manufacturing: a computing and service-oriented manufacturing model. Proc Inst Mech Eng Part B: J Eng Manuf 2011;225(10):1969–76.
[25] Zhang L, et al. Cloud manufacturing: a new manufacturing paradigm. Enterp Inf Syst 2012;8(2):167–87.
[26] Wu D, Terpenny J, Schaefer D. Digital design and manufacturing on the cloud: a review of software and services. In: Artificial intelligence for engineering design, analysis and manufacturing. Cambridge University Press; 2016. p. 1–15.
[27] Liu X, et al. A feature based method for product-Oriented representation to manufacturing resources in cloud manufacturing.pdf>. In: ASME 2014 International manufacturing science and engineering conference. 2014 [p. V001T04A017].
[28] Xu W, et al. Dynamic modeling of manufacturing equipment capability using condition information in cloud manufacturing. J Manuf Sci Eng 2015;137(4):040907.
[29] Kulvatunyou B, et al. A framework to canonicalize manufacturing service capability models. Comput Ind Eng 2015;83:39–60.
[30] Negri E, et al. Requirements and languages for the semantic representation of manufacturing systems. Comput Ind 2016;81:55–66.
[31] Zhang Y, et al. Task-driven manufacturing cloud service proactive discovery and optimal configuration method. Int J Adv Manuf Technol 2015:1–17.
[32] Ameri F, Urbanovsky C, McArthur C. A systematic approach to developing ontologies for manufacturing service modeling. In: Proc. 7th International conference on formal ontology in information systems (FOIS 2012). 2012 [Citeseer].
[33] Xia K, et al. A semantic information services framework for sustainable WEEE management toward cloud-Based remanufacturing. J Manuf Sci Eng 2015;137(6):061011.
[34] Staab S, Studer R. Handbook on ontologies. Springer Science & Business Media; 2013.
[35] Guarino N. Formal ontology in information systems. Proceedings of the first international conference (FOIS'98), vol. 46. 1998 [IOS press].
[36] Yen IL, et al. Rapid service composition reasoning for agile cyber physical systems. 2016 IEEE symposium on service-oriented system engineering (SOSE) 2016.