# An Interoperable Cloud Environment of Manufacturing Control System

**Yulia Yadgarova and Victor Taratukhin**

**Abstract** In presented study a new integrated framework for building cloud-based manufacturing environment is described. This approach allows develop future production systems as the class of adaptive distributed systems with virtual cloud model and simulation. Dependency model of equipment interaction is defined and the method of detailed production plan scheduling is introduced. As the result of the investigation architecture and prototype of the system is presented.

**Keywords** Industrial internet architecture · Learning systems · Cloud manufacturing · Multi-agent systems · Simulation

## 1 Introduction

Presently the concepts of Industrial Internet, M2M-interaction and Industry 4.0 [1] play a significant role in the process of creation Factories of the Future [2]. Complex industrial production in space and aerospace industries, collaboration of different manufacturers and rapid development of customized products imply changes in the production process. The modern manufacturing lines should be customizable, easy controlled and intelligent to meet complexity demands and to increase quality of the production [3].

Roadmap to the Industry 4.0 concept [1] implies transition of traditional control production process model to the distributed system with collaborative protocols on the different levels. On a business level such systems implement Single information space (with the concept of Virtual Enterprise). Virtual production line is the idea of

Y. Yadgarova (✉)
Bauman Moscow State Technical University, Moscow, Russia
e-mail: y.yadgarova@bmstu.ru

V. Taratukhin
European Research Center for Information Systems (ERCIS), Muenster, Germany

V. Taratukhin
Stanford University, Stanford, CA, USA

 
K. Mertins et al. (eds.), *Enterprise Interoperability VII*,
Proceedings of the I-ESA Conferences 8, DOI 10.1007/978-3-319-30957-6_1

geographically distributed production units and transparent technological process through these units [4, 5].

Research and development process of virtual enterprise concept was complex and intensive. There are a lot of research groups all over the world investigating this problem. The greatest ones are in USA, for example Advanced Manufacturing partnership [6] that focuses on the Advanced Manufacturing Enterprise concept. Another one is Intelligent machines (IM) which develops smart sensors and controllers also intelligence machines for the FMS (Flexible manufacturing line). Mentioning Europe activities, there is necessary to note global and local projects such as “Factories of the Future” (global), “Industry 4.0” (Germany), “Usine du Futur” (France). According to the “Factories of the Future” [2], classification there are three main concepts:

1. **Digital Factory** concept infers improving quality of the design and production processes as the decreasing expense of modeling and product knowledge management within the production process.
2. **Smart Factory** concept means creation of flexible, customizable production with computer automation systems and robotics.
3. **Virtual Enterprise** concept enforces value added creation by supply chain management and control of distributed manufacturing lines.
4. Also the Digital Factory is the base notion for creation of the Smart and Virtual Enterprises.

On (Fig. 1) detailed description of each type is presented.

The purpose of our investigation is to create framework, methods and prototype platform for the Smart Factory concept.

Industrial internet consortium initiative [7] defines common principles of building reference architectures of the interacted devices. Some principles (reliability, scalability, usability, maintainability, portability and so on) were taken as a

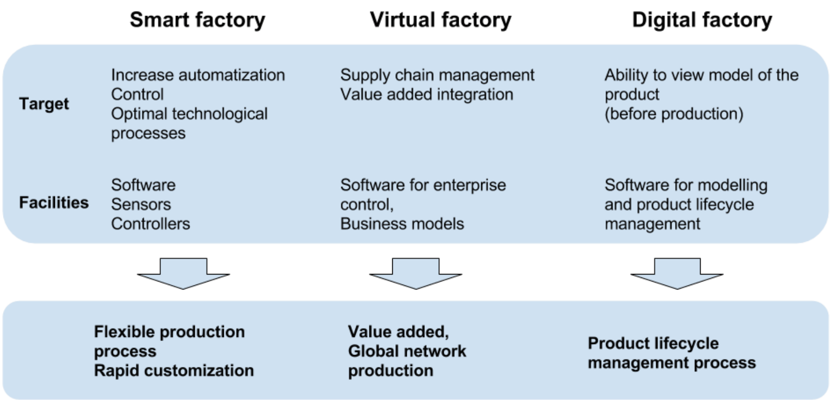


**Fig. 1** Concepts of the factory-of-the-future

basis of our architecture. Moreover there are several critical characteristics that system should implement alongside the standard [8].

Now the cloud systems are being integrated in plant landscapes. In the [9] researchers provide system architecture of cloud robotics system based on Machine-to-Machine and Machine-to-Cloud communications. At that work several cloud architectures have been compared (Peer-based, Proxy-based and Clone-based) also robustness, interoperability and mobility of such design solutions have been evaluated. Although the final option of result architecture depends on the application of the system. In the cloud robotics model we have several robots and ad-hock network between them [10]. During the production process there are different states when each unit of equipment can be critically important. Because of that in the current research the clone-based architecture of the system is presented.

Another challenge at constructing the cloud-based environment of production line is the integration of Information technologies domain (IT) and operation technologies (OT) [7]. In Information Technologies domain all system characteristics are represented by digital world. Symbol-grounding problem is the main problem of the information technology domain- representing a meaning of the symbols to machine. On the other case in Operation technology domain (OT) "control" should be applied directly to the physical process without any attempting to the models. In the IoT (Internet of Things) concept (and especially in Industrial Internet domain) it's vital to use the link between IT-domain and OT-domain. One of the main instruments for minimizing the difference between OT-state and IT-model is protocol of interaction between real-time devices and computer world. The first part of the study highlights the formal model of interaction of devices. In the second part the implementation of cloud system based on test equipment and stack of Java technologies is described.

## 2 Cloud Manufacturing Framework

A set of activities to create the cloud-based environment is complex. The whole investigation process was divided into several stages. Firstly the problem was to create clone of the real-world system with real-time interaction and minimization of the latency. The initial architecture of the physical system is presented on Fig. 5. The Flexible manufacturing line consisted of several devices: miller and lathe tools, conveyor, two robots, storage and robocara. All presented devices had to be exposed into cloud. The research project implied decreasing structure differences between real device and it's model clone. Secondly the thing was to provide interaction and self-organization of clones in the cloud. Achieving this goal would give strong evidence for feasibility of such architectures and in perspective would bring a big impact in building unmanned production environment. In fact the first issue is described in detail at [9] while the second have not properly investigated yet.

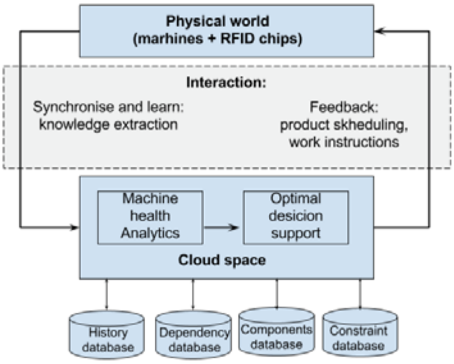


**Fig. 2** The cloud manufacturing framework

As the solution for the first issue we tie a RFID chip with work instructions information on the manufacturing part. Also each device and machine tool had a link with cloud environment through driver's software. As a result it can be stated that the stroke model of the production system in the cloud had been obtained. Now each physical device has its own representation model (clone) based on signals and driver's status. Meanwhile the part arrives to the shop floor information about its components is extracted from process instructions in the chip. After these activities the process of runtime configuration starts. The runtime schedule based on dependencies and constraints of the devices begins configuring (Fig. 2). During this process (described below) additional devices (that are needed for the operation) are extracted from the dependency database. Another (Constraint) database stores process attached rules (time constraints and sequences). Moreover, the essential part at self-organization process is history database, where analytics for all devices are stored. Process can be automatically rescheduled according to historical data.

Dependency database consists of rules that link two or more devices. To start processing on the machine tool, part should be transported on the work space by the robot. Meanwhile to grab and move the part robot has to perceive it in the processing zone. So, robot need conveyor to transport part in that zone.

Based on dependency database information, relation model of shop floor constraints of the equipment links, history the concrete work instructions and time schedule are arrived on the virtual clones in the cloud model (Fig. 3).

Constraint database are used after Dependency storage to indicate limitations between devices. Overall schema of the production plan scheduling in flexible manufacturing system is presented at Fig. 4.

Next step of the cloud control process is simulation. The main impacts of the modelling are to ensure that there are no gaps between devices interaction and to predict system failures and troubles. In case of time/structure gap the system asks master of the shop floor and offers a several variants. Below the most essential branches of the research are described in detail.

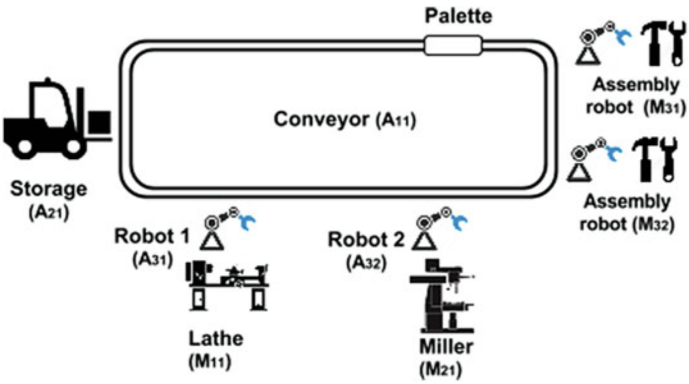


**Fig. 3** Schema of the flexible manufacturing line

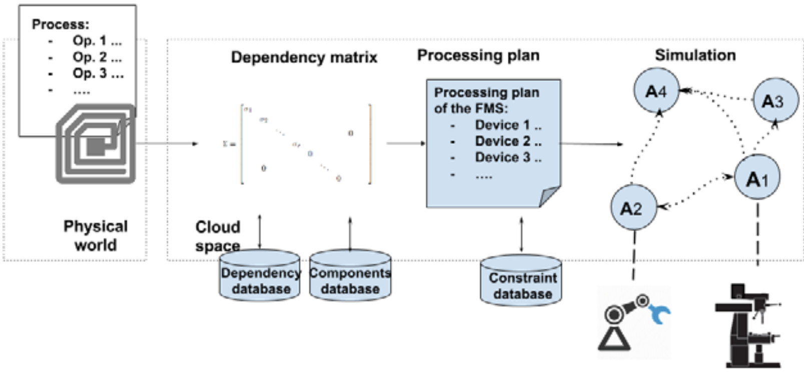


**Fig. 4** Sequence of the processing

## *2.1 Architecture of Clone-Based Model*

The target system was developed based on clone-based architecture [9] and implements twin for each equipment unit. At once each clone was represented by the actor [11] that could be able to cooperate with other actors and to solve scheduling and operation problems at the shop.

The final environment consists of static Actor-Dependency model, presented above. Each equipment unit have a virtual clone (Ai, actor), that have a dependency with other units (Fig. 5). These dependences are defined by the user during developing the cloud structure of the manufacturing line.

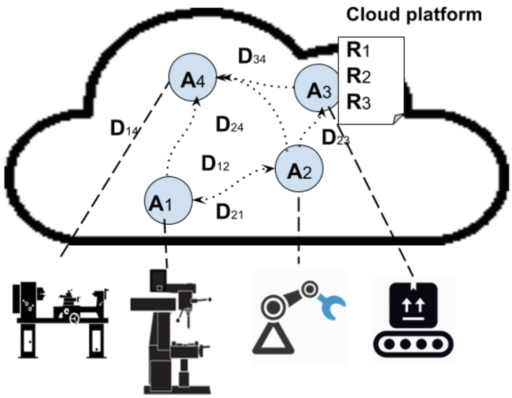


**Fig. 5** Dependencies and components

## 2.2 Operation Processing Plan Generation

The cloud model represents different equipment units. In this system each equipment unit is presented as actor with dependencies (Fig. 6).

The production process starts when the task arrives to the workshop. At the time point $t$ states of the system's units are presented by the equation:

$$O(t) = \{M_{jk}(t)/j = 1,\ldots,J;\ k = 1,\ldots,K_j, \\ A_{mn}(t)/m = 1,\ldots,M; \quad n = 1,\ldots,N_m \tag{1}$$

$M_{jk}$ The main equipment (machine tools, assembly robots, work centers);
$A_{mn}$ Auxiliary equipment (manipulators, conveyors, robocars, storage);
$J$ Number of types of the main equipment;
$K_j$ Number of units of the main equipment ($j$—type);
$M$ Number of types of the auxiliary equipment;
$N_m$ Number of units of the auxiliary equipment ($m$—type);

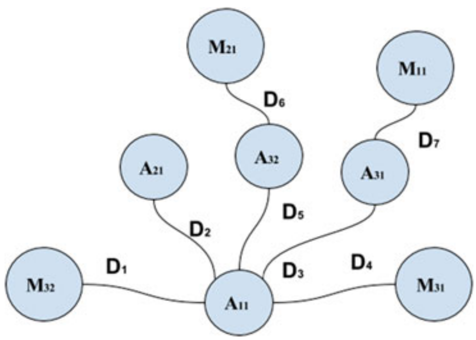


**Fig. 6** Dependency graph of the equipment

$$F^D = \begin{array}{c|cccccccc} & M_{11} & M_{21} & M_{31} & M_{32} & A_{11} & A_{21} & A_{31} & A_{32} \\ \hline M_{11} & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ M_{21} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ M_{31} & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ M_{32} & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ A_{11} & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ A_{21} & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ A_{31} & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ A_{32} & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \end{array} \tag{1.2}$$

**Fig. 7** Dependency matrix for the flexible manufacturing line

The *k*—unit of main equipment of the *j*—type $M_{jk}(t)$ is characterized by several parameters (properties and restrictions).

The production process is defined by the dependency graph of the equipment (Fig. 6) as well as Dependency matrix $F^D$ (Fig. 7).

Below the matrix for production line presented at (Fig. 3) is defined.

Dependency graph for relevant flexible manufacturing system is presented on the Fig. 6.

There are two types of devices are presented in the matrix: main and auxiliary. Indeed each device of particular type has restrictions that help system to build a correct operation plan. Restrictions resemble rules in this case. For instance, all auxiliary devices' dependencies of type A→M (dependency from Auxiliary to the main type works from one way).

Dependency matrix is static and forming while user of the system defines structure of the production line.

*Case:*

Order incoming to the shop floor is presented as:

$$Z_i(t) = \langle M_i, \alpha_j, t_i, T_i^t \rangle \tag{3}$$

where

$M_i$ Sequence of the main equipment unit types (route map);
$\alpha_j$ Processing conditions (time of the processing, machine tools' parameters, CNC-program name etc.);
$t_i$ Time of the order income;
$T_i$ Planned time of issue;

Auxiliary devices dependencies are calculated by applying sequence of processing (input parameter). For instance, if the sequence of processing consists of $M_1$ and $M_3$ then the first step is to define equipment units of this type to meet target function.

As a target functions there are several types that can meet production requirements. The target function can be minimization of processing time or cost, also minimization of equipment downtime, etc. It depends on the final target and can also be multipurpose.

At the further steps of defining operation sequence auxiliary dependencies should be obtained. In this case, dependencies $A_{11}$ and $A_{31}$ was found by using Dependency matrix. According to matrix data and restrictions, the overall operation production sequence presented as:

$$A_{11} \rightarrow A_{31} \rightarrow M_{11} \rightarrow A_{31} \rightarrow A_{11} \rightarrow M_{31} \quad (4)$$

## 3 Prototype of the System

Prototype of the system was developed using Flexible Manufacturing line in research lab (Fig. 3).

### *3.1 Architecture of the System*

The Cloud-based manufacturing model consists of two parts: local part including device drivers (connectors), communication server, reactive planner (to perform real-time actions) and cloud part based on web technologies (Fig. 8).

Capabilities of local part include control physical device and communication with the cloud server. All functions that driver can manage are transmitted to the cloud, as with monitored statuses. Protocol of interaction between cloud and device is based on Web-Sockets [12]. The presented architecture guarantees fast delivery of the messages indeed if device would be disconnected from control system.

The cloud part of the system consists of several architecture blocks (Fig. 8):

- communication web-socket server
- presentation and settings model
- process planer and modeler
- databases (dependency, components, constraint)
- decision support system
- analytics model

During the device control process the cloud component of the system performs simulation within the server; on the next steps commands will be transmitted to the physical system.

Characteristics of the clone model (functions, properties and dependencies) are defined by the user and stored in the particular database. Interaction between several agents is rested on the properties and dependencies.

At Fig. 9 user interface of the system is presented. There are conveyor dependencies definition process and tracking conveyor state.

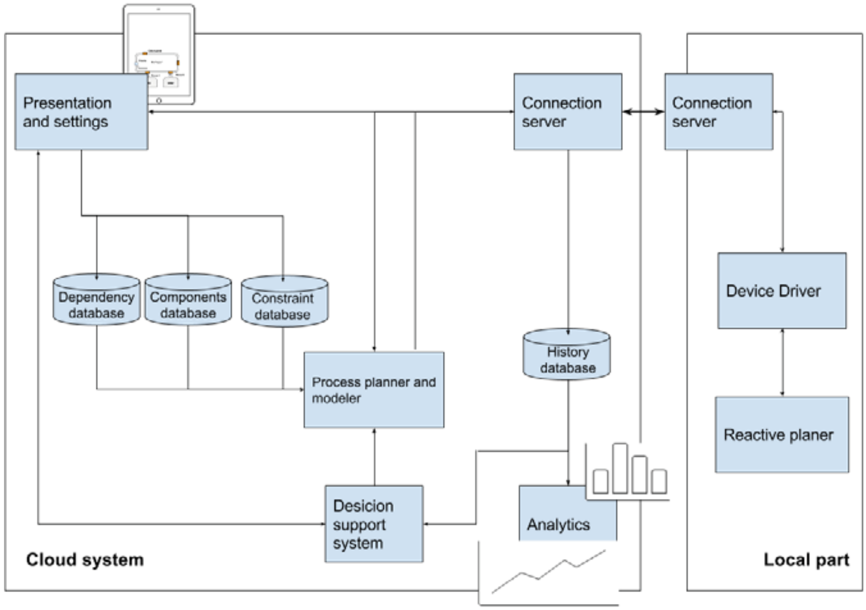


**Fig. 8** Architecture of the system

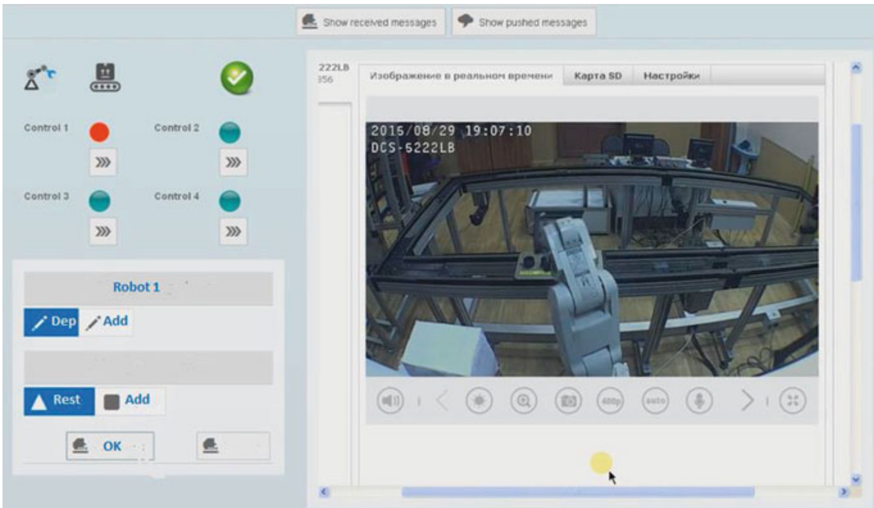


**Fig. 9** Interface of the system

## 4 Conclusions

At research study the integrated framework for cloud-based manufacturing process is described. The method of developing detailed production plan schedule based on the dependency model is described. Presented investigation provides evidence for efficiency of such architectures.

We hope that obtained results can be the basis for design future production systems as the class of adaptive distributed systems.

## References


1. Industry 4.0 Smart Manufacturing for the Future. http://www.gtai.de/GTAI/Content/EN/Invest/_SharedDocs/Downloads/GTAI/Brochures/Industries/industrie4.0-smart-manufacturing-for-the-future-en.pdf. Accessed October 3, 2015.
2. Factories of the Future PPP: Towards Competitive EU Manufacturing. http://ec.europa.eu/research/press/2013/pdf/ppp/fof_factsheet.pdf. Accessed October 3, 2015.
3. Brettel, M., et al. (2014). How virtualization, decentralization and network building change the manufacturing landscape: An industry 4.0 perspective. *International Journal of Science, Engineering and Technology* 8(1), 37–44.
4. Lee, J., Kao, H.-A., & Yang, S. (2014). Service innovation and smart analytics for industry 4.0 and big data environment. *Procedia CIRP*, *16*, 3–8, ISSN 2212-8271.
5. Yadgarova, Y., Taratoukhine, V., & Skachko, E. (2015). Multi-agent product life cycle environment. Interoperability issues. In: Lecture Notes in Business Information Processing, Berlin, Heidelberg: Springer.
6. Report to the President on Ensuring American Leadership in Advanced Manufacturing. (2011, June). *President's Council of advisors on science and technology*. http://www.whitehouse.gov/sites/default/files/microsites/ostp/pcast-advanced-manufacturing-june2011.pdf. Accessed October 3, 2015.
7. Industrial Internet Reference Architecture. http://www.iiconsortium.org/IIRA.htm. Accessed October 4, 2015.
8. Taratoukhine, V., & Bechkoum, K. (1999). Towards a consistent distributed design: A multi-agent approach. In: *Proceedings of Information Visualisation.*
9. Hu, G., Tay, W. P., & Wen, Y. (2012). Cloud robotics: Architecture, challenges and applications. *IEEE Network, 26*(3), 21–28.
10. Chibani, A., et al. (2013). Ubiquitous robotics: Recent challenges and future trends. *Robotics and Autonomous Systems, 61*(11), 1162–1172.
11. Hewitt, C., Bishop, P., & Steiger, R. (1973). A universal modular ACTOR formalism for artificial intelligence. *Proceedings of the 3rd International Joint Conference on Artificial Intelligence (IJCAI'73)* (pp. 235–245). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.
12. The WebSocket Protocol. https://tools.ietf.org/html/rfc6455. Accessed October 10, 2015.