Design of device adapter enabling plug & produce for intelligent manufacturing production line

Xia’nan Yao¹, Di Li¹, Hao Tang¹, Xixiang Wang¹ and Yongchao Luo²*

¹School of Mechanical & Automotive Engineering, South China University of Technology, Guangzhou, Guangdong, 510640, China
²School of Electrical Engineering, Guangzhou College of South China University of Technology, Guangzhou, Guangdong, 510800, China
*Corresponding author’s e-mail: luoyc@gcu.edu.cn

Abstract. Aiming at the problem that traditional manufacturing production lines are difficult to meet the increasingly customized product requirements due to their poor flexibility and adaptability, a design framework and implementation method for the intelligent device adapters that followed the model-driven concept were proposed. Firstly, the importance of model-driven approach based on the requirements of production line for customization was explained; then a variety of mainstream hardware interfaces and software communication protocols of devices in the modern manufacturing field were analysed. By this means, the software and hardware systems of the device adapters were designed in a modular manner, so that the ability to automatically perceive the surrounding environment and communicate externally with the standard service interfaces of OPC UA was achieved. Finally, the results of the functional verification carried out on the production line prototype platform show that the intelligent device adapters can automatically register the metadata of devices based on the self-description model to the service catalog, and the communication protocols and data description formats of devices can also be uniformly adapted to OPC UA, so as to realize the capability of plug & produce without manual intervention.

1. Introduction

In recent years, consumers’ demands for products have shown an increasing trend of diversification and customization [1]. This trend of consumption has caused serious impact and challenge to the traditional production pattern which is used to produce large quantities of products with low varieties. In order to maintain a continuous competitive advantage in manufacturing, manufacturers are forced to comply with consumers’ demands while ensuring the ability to deliver quickly [2]. With the emergence and development of emerging technologies such as big data [3] and cloud computing [4], the Cyber-Physical System (CPS) [5] is considered as a key technology to promote the transformation and upgrading of a production line which is a typical collaborative manufacturing system towards a more intelligent direction.

Since the production line for customization covers many fields such as machinery, electric and communication, there exists plenty of tools for engineering design and operation control, and the processing devices also face great uncertainty. If relying solely on manual rearrangement and configuration of devices according to the specific products, it will undoubtedly consume a lot of time and labour costs, and the process is complicated with low fault tolerance. Therefore, devices for
intelligent production lines are required to have the features of self-organization and self-adaption based on the models to improve the efficiency of integration and collaboration.

In this context, scholars all over the world have also conducted lots of researches in terms of theory and practice respectively. For example, the openMOS project [6-7] has proposed an implementation approach of device adapters using AutomationML and OPC UA, while Ye, X. et al [8] have proposed a plug-and-produce approach for devices based on the asset administration shell (AAS). However, both of them mainly concentrate on the realization of interoperability between devices with a unified service interface, and rarely mention how to automatically adapt heterogeneous devices and to endow them with dynamic operational characteristics on the basis of self-describing models. Therefore, there is a low degree of automation during the integration of the production line. From the perspective of the requirements of intelligent production lines for customization, the importance of model-driven approach is explained, and a design and implementation method of the device adapter that follows the concept of plug-and-produce is proposed.

2. Requirements of intelligent production lines

Since the intelligent production line is used for the manufacture of customized products, there is frequent switching and recombination of devices and changes in process parameters during the execution of different processing tasks. This will introduce a lot of ineffective engineering time to the manufacturing system. If the function and performance of the production line can be predicted through technologies such as virtual commission before actual processing, the implementation efficiency of the production line will be greatly improved. In view of this, the use of virtual digital models (which are an abstract and objective description of the specific fields of interest to users throughout the product development) and the organic combination with their physical entities can achieve interoperability of the systems during engineering and integration [9]. This is in line with the idea of model-driven architecture proposed by the Object Management Group (OMG) in 2001 [10].

Figure 1. Model-driven process of intelligent production line.

Focusing on the specific production factor of processing devices and referring to the concept of Industry 4.0 components [11], the device model is a digital mirror of the device object in the physical space. Because the two have a mutual mapping relationship, the different objectives of the production line throughout the whole lifecycle can be achieved by substituting or interacting with each other. As shown in figure 1, when the customer issues the product order, the mutual substitution relationship can be used to realize the virtual commission of the production line through the device models; and after the reachability is proven, the interaction relationship can be used to form a dynamic closed-loop system between the control signal of the model and the operating status of the physical device [12].

3. Design framework of device adapter based on model-driven

Since the intelligent production lines need to dynamically switch and combine various types of processing and transportation devices according to specific objects, thus forming a product-specific manufacturing process, it is necessary to decouple various types of devices in the surroundings into a set of discrete single systems that can be freely combined and organized, so as to improve the
flexibility and adaptability of production lines. The utilization of the cloud manufacturing concept [13] can satisfy the purpose of reorganizing and sharing of devices on demand through the network. In this context, to empower production lines to be easily reconfigured, each distributed device needs to be developed independently, and then connected to the manufacturing system in the form of service interfaces, so that the collaborative production can be achieved through dynamic organization and data exchange of devices in the cloud.

Considering that heterogeneous devices from different manufacturers often do not provide specific communication interfaces, leading to difficulties in integration during collaborative manufacturing in the plug-and-produce pattern, as well as that some legacy devices does not have any network interfaces except for only semaphores triggered by I/O, this paper proposes a design framework of an device adapter based on model-driven concept, which is regarded as a distributed intelligent node and directly deployed at the edge of the device, as shown in figure 2.

In this case, the device model is described using the AutomationML data format which enables the storage and interaction of different engineering data by connecting heterogeneous information from different engineering tools and integrating semantics and knowledge across disciplines [14]. Based on the device model, the functions of the intelligent device adapter are developed in a modular manner which has strong stability and is convenient for later iteration and maintenance.

4. Implementation of device adapter

4.1. Software system

The software design of the intelligent device adapter is mainly used to support the plug-and-produce ability based on the model-driven concept. This section describes the implementation of each software module of the device adapter in the architecture shown in figure 2.

Figure 2. Design framework of device adapter.

4.1.1. Model converter.

As mentioned above, the information model of devices for intelligent production lines is described in the AutomationML data format. However, it only integrates relevant concepts from different engineering domains, and does not regulate the way of data interaction in the operation phase. Therefore, when realizing dynamic behaviour characteristics at the operation phase, the standard OPC UA protocol [15] is adopted in this paper as the unified communication interface of the device adapter. Due to its information modelling of complex objects on a semantic basis and the ability to communicate in a service-oriented manner across platforms, it is accepted by more and more domains and enterprises.

Before exposing it to the public as an OPC UA service interface, the device model needs to be mapped to the OPC UA information model and associate it with the production operation data. Since the manufacturing process is closely related to the operation of capabilities provided by the devices,
the model conversion rule proposed in literature [16] is needed to be further expanded, so as to realize the mapping of the PLCopen XML interface defined in the AutomationML model to the method node in the OPC UA address space.

The implementation flow of the model converter is shown as in figure 3. As seen from the figure, in addition to mapping the information model, the model converter also implements automatic code generation of callback functions for variable nodes and method nodes so that the corresponding interaction logic can be automatically triggered when other production factors access device data or perform the process capabilities of the device. Finally, the OPC UA server is implemented based on the open-source library open62541.

4.1.2. Protocol mapping and Logic mapping.

In order to present the data source and operation control of heterogeneous devices to the public with a unified OPC UA service interface, a protocol mapping module and a logic mapping module were developed in the intelligent device adapter respectively. The protocol mapping module is mainly used to handle the download and upload of the configuration parameters or the running status feedback, and then map to the corresponding OPC UA variable nodes according to the binding rules of heterogeneous data sources. This module integrates a variety of mainstream communication protocols and interfaces (such as TCP/IP, Modbus, GPIO, serial port, etc.) in current industrial manufacturing sites. By resolving the data source attributes of the data elements defined in the AutomationML model of the device, it can dynamically switch to the corresponding underlying communication protocols and map the real-time data to the OPC UA address space, thus realizing the automatic adaptation function of different communication protocols.

4.1.3. Service Discovery and Registration.

Because of the unpredictable dynamic addition and removal of devices on a production line, a service discovery and registration module is configured in the device adapter to implement the plug-and-produce ability without downtime and to empower the cloud to dynamically monitor and manage available devices. Service discovery and registration is such a process where a new device can automatically discover the service catalog in the environment and then register relevant information
once it is just plugged into the production line, thus enabling the manufacturing field to be informed of all currently available devices and invoke their relevant processing capabilities.

The service discovery and registration function is implemented by the Consul service grid, which can complete the service discovery and configuration of distributed systems, and can query the description information and operation status of distributed service. There is a prerequisite that all nodes on the production line (including private clouds and device adapters) need to be included in the Consul service grid. Therefore, it is necessary to deploy the Consul server as a service catalog on the private cloud in advance, and deploy the corresponding Consul proxy in the device adapters. During the operation phase of the intelligent production line, once a new device is dynamically plugged in, the device adapter will request the local address of the service catalog from the domain name system (DNS) server, and then register the type, name, identifier, Internet Protocol (IP), access port and process capabilities that the device can provide to the service catalog, as shown in figure 5.

Figure 5. Procedure of service registration.  Figure 6. Intelligent device adapter.

4.2. Hardware system

In order to meet the needs of various industrial scenarios and heterogeneous devices, the hardware system of intelligent device adapters needs to support common industrial communication interfaces, such as GPIO, RS-232, RS-485, RJ45, etc. In this paper, Raspberry Pi 3B+ is selected as the main control unit of the device adapter, which has 40 GPIO pins, one RJ45 Gigabit Ethernet interface, four USB interfaces, and supports convenient program development and deployment via SSH under Linux operating system. By extending the USB to RS-232/RS-485 serial port connection and configuring the mapping rules of hardware interface, the Raspberry Pi also has the capability of RS-232/RS-485 serial communication to connect those supporting Modbus-RTU and other protocols. The intelligent device adapter designed in this paper is shown in figure 6.

5. Case Study

For functional validation, the design and implementation method of intelligent device adapters proposed in this paper is applied on the production line prototype platform built by Smart Manufacturing Laboratory of South China University of Technology. As shown in figure 7, the prototype platform is mainly composed of an upload-box station, an upload-workpiece station, two computerized numerical control (CNC) stations, a laser-labeling station, a packaging station, an upload-cover station, an unload-product station, two automated guided vehicles (AGVs) and several conveyors, responsible for the production of customized U-disks and wooden souvenirs.

Here a conveyor is used as an example to verify whether the device adapter can achieve the plug-and-produce ability based on a model-driven concept. The conveyor is controlled by a programmable logic controller (PLC), which is physically connected to the device adapter via an RS-485 interface and interacts via the Modbus-RTU protocol at the software level. The AutomationML model of the conveyor is also deployed on the device adapter in advance. As mentioned above, when the conveyor and its device adapter are connected to the production line, the adapter will automatically perform the model conversion to present it externally with the standard OPC UA service interface, and find the service catalog through DNS and register relevant information of the conveyor with it. As shown in figure 8, the monitor of the Consul service catalog in the cloud indicates that the conveyor has been
successfully registered to the cloud, and provides device metadata such as IP and port for communication. Figure 9 shows the connectivity of the IP and port of this conveyor through a third-party OPC UA client. As seen in the figure, the conveyor information model can be correctly mapped to the OPC UA address space, the belt speed can be initialized through the configuration of the variable node, and the running status can be performed by invoking the method node. This shows that the model converter and the protocol mapping and logic mapping modules have good versatility and feasibility. By continuous processing of 200 customized products and recording the memory status of the device adapter every half an hour, the statistical results show that the average memory consumption is 48.13MB, and there is no unexpected interruption during the experiment, which further proves the reliability of the device adapter.

6. Conclusion

Aiming at the problem that traditional manufacturing production lines are difficult to meet the increasingly customized product requirements due to their purpose for large quantities of products with low varieties, this paper follows the model-driven concept, proposes a design framework of the intelligent device adapters for edge-cloud collaboration and an implementation method of modular functions resulting from the requirements of production lines for customization. Experiments are conducted on the production line prototype platform eventually, and the results show that the proposed device adapter can uniformly adapt the data access and process execution interface of underlying heterogeneous devices in the OPC UA standard service, and can empower the production line to monitor and manage the available devices in real time by means of service discovery and registration. The concept of plug-and-produce is of great theoretical and practical significance for production lines that require frequent reconfiguration.
Acknowledgments
This work was supported in part by the National Key R&D Program of China (Grant No. 2018YFB1700500) and Science and Technology Planning Project of Guangdong Province (Grant No. 2017B090914002).

References
[1] Zhou, J., Yao, X., Liu, M., et al. (2017) State-of-Art review on new emerging intelligent manufacturing paradigms. Computer Integrated Manufacturing Systems, 23(03): 624-639 (in Chinese).
[2] Herrmann, C., Schmidt, C., Kurle, D., et al. (2014). Sustainability in manufacturing and factories of the future. International Journal of precision engineering and manufacturing-green technology, 1(4): 283-292.
[3] Chen, M., Mao, S., Liu, Y. (2014). Big data: A survey. Mobile networks and applications, 19(2): 171-209.
[4] Xu, X. (2012). From cloud computing to cloud manufacturing. Robotics and computer-integrated manufacturing, 28(1): 75-86.
[5] Fu, J. (2014) Development status and trend of intelligent manufacturing equipment. Journal of Mechanical & Electrical Engineering, 31(08): 959-962 (in Chinese).
[6] Dorofeev, K., Cheng, C. H., Guedes, M., et al. (2017). Device adapter concept towards enabling plug&produce production environments. In: 2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA). Limassol. pp. 1-8.
[7] Miranda, F., Martins, R., Dorofeev, K., et al. (2018). Towards a common manufacturing service bus to enable flexible plug-and-produce automation. In: 2018 50th International Symposium on Robotics (ISR). Munich. pp. 1-8.
[8] Ye, X., Jiang, J., Lee, C., et al. (2020). Toward the Plug-and-Produce Capability for Industry 4.0: An Asset Administration Shell Approach. IEEE Industrial Electronics Magazine, 14(4): 146-157.
[9] Sun, H., Xiao, T., Fan, W., et al. (2012) Novel model-driven joining technology for cloud manufacturing federations. Computer Integrated Manufacturing Systems, 18(07): 1536-1546 (in Chinese).
[10] Frankel, D.S. (2003) Model Driven Architecture: Applying MDA to Enterprise Computing. Wiley, New York.
[11] Ye, X., Hong, S. H. (2019). Toward Industry 4.0 components: Insights into and implementation of asset administration shells. IEEE Industrial Electronics Magazine, 13(1): 13-25.
[12] Jiao, H., Lu, J. (2018) Summary of smart factory and its key technologies. Journal of Mechanical & Electrical Engineering, 35(12): 1249-1258 (in Chinese).
[13] Yi, L., Tu, J., Liang, X. (2016) Cloud manufacturing resource planning and its coordination mechanism based on supply sub-chain. Journal of Mechanical & Electrical Engineering, 33(12): 1442-1447 (in Chinese).
[14] Drath, R., Luder, A., Peschke, J., et al. (2008). AutomationML-the glue for seamless automation engineering. In: 2008 IEEE International Conference on Emerging Technologies and Factory Automation. Hamburg. pp. 616-623.
[15] Mahnke, W., Leitner, S.H., Damm, M. (2009) OPC Unified Architecture. Springer, Berlin.
[16] Henssens, R., Schleipen, M. (2014). Interoperability between OPC UA and AutomationML. Procedia Cirp, 25: 297-304.