applied sciences

Article

Cloud-Edge Collaboration-Based Knowledge Sharing Mechanism for Manufacturing Resources

Xixiang Wang and Jiafu Wan *

Guangdong Provincial Key Laboratory of Technique and Equipment for Macromolecular Advanced Manufacturing, School of Mechanical and Automotive Engineering, South China University of Technology, Guangzhou 510641, China; mesimonwang@mail.scut.edu.cn
* Correspondence: mejwan@scut.edu.cn; Tel.: +86-020-2223-6887

Abstract: The development of multi-variety, mixed-flow manufacturing environments is hampered by a low degree of automation in information and empirical parameters' reuse among similar processing technologies. This paper proposes a mechanism for knowledge sharing between manufacturing resources that is based on cloud-edge collaboration. The manufacturing process knowledge is coded using an ontological model, based on which the manufacturing task is refined and decomposed to the lowest-granularity concepts, i.e., knowledge primitives. On this basis, the learning process between devices is realized by effectively screening, matching, and combining the existing knowledge primitives contained in the knowledge base deployed on the cloud and the edge. The proposed method's effectiveness was verified through a comparative experiment contrasting manual configuration and knowledge sharing configuration on a multi-variety, small-batch manufacturing experiment platform.

Keywords: cloud-edge collaboration; edge computing; IoT; manufacturing resources; knowledge sharing; ontology; smart manufacturing

check for updates

Citation: Wang, X.; Wan, J. Cloud-Edge Collaboration-Based Knowledge Sharing Mechanism for Manufacturing Resources. *Appl. Sci.* **2021**, *11*, 3188. https://doi.org/10.3390/app11073188

Received: 16 March 2021
Accepted: 31 March 2021
Published: 2 April 2021

1. Introduction

In recent years, the manufacturing model has shown a shift in personalized customized production, leading to the need for manufacturing companies to complete more frequent manufacturing task adjustments in a shorter time-to-market cycle, which requires the manufacturing system to quickly complete the production line planning and scheduling layer refactoring. In the existing multi-product mixed-flow manufacturing production line, improving the production ramp-up time during the reconfiguration process will be a key factor in achieving production reconfiguration. However, traditional manufacturing resources are limited by their hardware computing capabilities, and a lot of human interventions are required in the process of complex task reconstruction, such as re-programming and commissioning. With the development of information technology, the Internet of Things [1,2], cloud computing [3,4], edge computing [5,6], big data, and AI (artificial intelligence) [7] technologies have become more and more extensive in the manufacturing field [8]. In this context, the cloud-edge collaborative architecture has given stronger computing intelligence capabilities to edge nodes such as manufacturing resources [9], making it possible to share information and learn knowledge between manufacturing resources. On this basis, the study of task decomposition and task primitive modeling to build a resource knowledge base, to realize the sharing of processing experience between equipment will be able to effectively improve the ramp-up time of production [10].

To optimize the data transmission in the intelligent manufacturing process and meet the requirements of the complex environment, Li et al. [11] proposed a solution to optimize the edge computing network through software-defined network management physical device interfaces and prioritize different data streams. Combine path differences and time limits to obtain the best routing path. Wan et al. [3] used a vertically integrated four-layer

CaSF (cloud-assisted smart factory) architecture to combine artificial intelligence and cloud computing to improve the scalability and reconfigurability of smart factories. Based on the existing cloud computing, fog computing, and edge computing, Qi et al. [12] proposed a hierarchical reference architecture for intelligent manufacturing and applied it to the digital twin workshop to better meet the needs of intelligent manufacturing applications. Aiming at the complex energy consumption problem of manufacturing clusters, Chen et al. [13] used energy perception and load balancing methods to effectively schedule equipment in smart factories. Yang [14] and others introduced an edge gateway that connects and manages plant equipment on the edge side of the cloud-side collaboration architecture and uses the edge gateway to respond to delay-sensitive applications in real-time. To realize the reuse of manufacturing process knowledge, Li et al. [15] proposed a metadata-based modeling method for manufacturing resource ontology, which solves the problem of heterogeneous description of manufacturing resource ontology in the cloud environment through the standardized description of the ontology. In terms of similar robot service configuration, Yang et al. [16] proposed an ontology-based service configuration method, which develops reusable service configuration by combining ontology and standard network interfaces. Moritz [17] researched the search of semantic environment model and map knowledge base based on the information exchange network between robots and used the mobile operation control of robots to reuse service configuration. Through the analysis of the design process and the manufacturing process, Peter et al. [18] proposed a scheme to use design experience and equipment failure experience as knowledge to realize experience reuse through ontology modeling.

Although the above literature has made certain research on edge computing and knowledge sharing, there are still some problems: (1) In the traditional cloud edge structure, the cloud carries more computing and control tasks. When the amount of data increases rapidly, it will put a lot of pressure on the transmission network, and the centralization of control and storage will increase the dependence of the entire system on the cloud; (2) the existing ontology modeling in the industrial field is mostly based on equipment self-generation, which cannot fully implement the manufacturing process; (3) the generality of the existing learning process knowledge is not strong, and most of them can only learn the same or highly similar process. Therefore, based on the processing environment of multi-variety manufacturing, this paper proposes a cloud-side collaborative manufacturing resource knowledge sharing mechanism, which provides a new solution for the mutual learning and knowledge-sharing of manufacturing resources. The contribution of this article mainly includes the following three aspects.

- Adopting an overall communication architecture based on cloud-edge collaboration, and deploying an intelligent manufacturing knowledge sharing system at the edge, realizing the combination of local learning and cloud learning between devices.
- Combining ontology to standardize, formalize and abstract the equipment attributes, manufacturing process, environmental restrictions, and equipment action execution effects involved in the field of intelligent manufacturing, and provide a fine-grained decomposition plan for the manufacturing process.
- A matching mechanism of knowledge search and reuse is proposed so that the processing experience of manufacturing resources can be reused.

The remainder of this article is organized as follows. Section 2 proposes a data communication network architecture based on cloud-edge collaboration. Section 3 defines the manufacturing resources' ontology structure and combines the production process elements to establish a semantic model enabling task decomposition. Section 4 puts forward the similarity matching algorithm for knowledge reuse based on task decomposition. Section 5 verifies the proposed knowledge sharing mechanism's effectiveness in a multi-variety mixed production line prototype platform and discusses the current problems and future research directions. Finally, Section 6 briefly summarizes the work.

2. System Architecture

As the number of communication data nodes in the manufacturing system rises, the data volume increases and the equipment description becomes more detailed. This paper adopts the concept of cloud-edge collaboration and constructs data communication network architecture in a manufacturing environment to enable the reuse of multi-variety mixed-flow manufacturing and processing experience. The edge functions include constructing the processing actions' knowledge ontology, decomposing different task execution processes' actions, orderly combining the existing knowledge action primitives to obtain complex task-oriented processing procedures, storing the local resources and knowledge, and performing the matching for knowledge reuse. The cloud functions include deploying the overall manufacturing environment knowledge base, enabling the interaction between users and managers and the system, and supplementing and timely updating the edge-side knowledge.

The system's framework is shown in Figure 1, which is composed of three parts: the manufacturing resource layer, the edge layer, and the cloud layer. The current manufacturing processes are shaped by the requirements regarding products' multi-variety, small batches, and high personalization and customization. In such multi-variety mixed-flow manufacturing systems, user needs are obtained directly through the cloud. After the user places an order on the web page, the order data is passed through the cloud, which, in turn, communicates with the edge server responsible for processing control. The edge server disassembles and refines the orders' processing tasks. After the processing procedure is determined, the specific execution plan is passed to the lowest layer (i.e., physical manufacturing resources). Once the manufacturing resource processing terminates, the processing completion status is set. The processing procedure's data is summarized, and the edge layer's and cloud layer's knowledge bases are updated. Compared with the existing cloud-edge collaboration three-tier architecture, to realize the knowledge sharing in the manufacturing environment better, we have deployed task-related functions on the edge side. In the data upload process, in addition to data preprocessing, the edge side will also decompose the processing tasks, extract reusable knowledge from the processing data and update the knowledge base. During the downward process of the control instruction, the edge side will decompose the task and find the knowledge primitives that can complete the task from the knowledge base. Effectively control the operation of manufacturing resources after the combination of knowledge primitives.

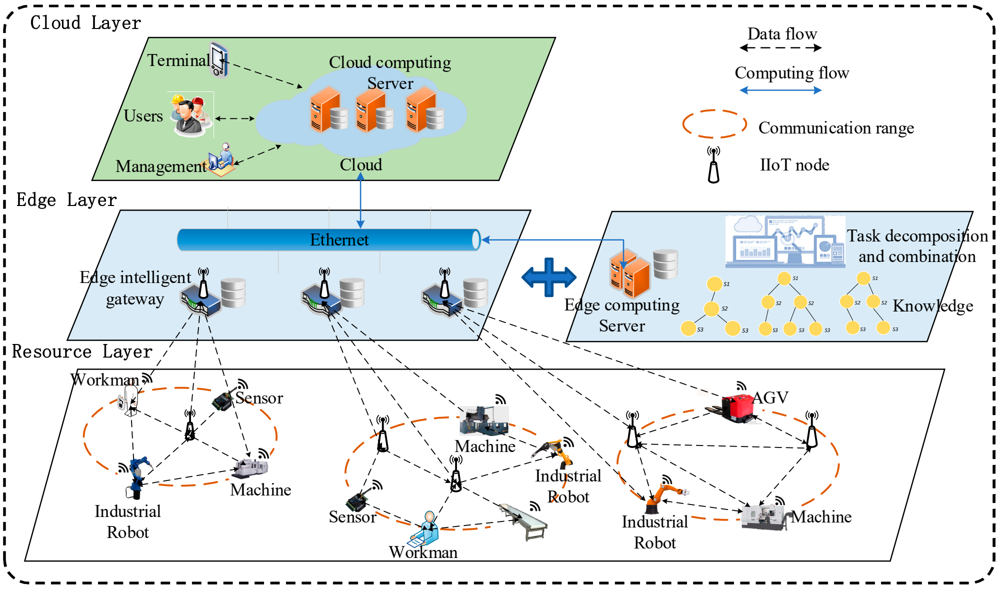


Figure 1. Data communication network architecture based on cloud-edge collaboration.

Manufacturing process data is characterized by large transmission volume [19], low latency, and high security. Using conventional cloud solutions for data communication results in high data transmission bandwidth costs and large delays. Therefore, the edge layer is deployed on the resource side to perform data preprocessing and caching, alleviating the transmission network's pressure. The edge layer is composed of two functional modules responsible for data transmission and data processing. The data transmission module includes intelligent edge gateways and wireless network cluster head nodes. The data processing module includes edge computing servers and edge knowledge bases, which perform processing tasks on the near-resource side, decompose and reorganize the equipment, and enable the equipment ontology modeling and knowledge extraction. The cloud-edge collaborative system architecture extracts knowledge that is suitable for manufacturing resources. The knowledge that is frequently used can be directly obtained on the edge side. Contrary, when the edge knowledge base does not find a matching task, the knowledge can be obtained through cloud communication for a wider search.

3. System Realization

The use of ontologies enables abstraction, standardization, and formalization of equipment attributes and their relationships within the intelligent manufacturing field. Therefore, reusable knowledge extraction and storage can be achieved through the equipment's ontological modeling and processing. Due to its ease of use and maintainability, this research utilizes the open-source software Protégé with a graphical user interface to construct the devices' ontology. Compared with the existing research on ontology modeling in the manufacturing environment, the ontology model construction method proposed in this paper is developed around the manufacturing task, not only modeling the physical resources involved in the manufacturing process, but also the surrounding environment elements related to the resource, the evaluation of the execution effect, and the matching degree of the action and the task. In the description of the manufacturing process, this article also proposes a task decomposition scheme that allows it to adopt a more fine-grained representation method, and also provides more knowledge materials for the reuse of knowledge.

The data flow of system operation is shown in Figure 2, including two parts: data uplink and data downlink. When data collection is performed, the edge server will first check the data collection node. If the local node does not have a mapping relationship with the mid-node of the ontology model, the mapping relationship will be established based on the characteristics of the data and the node type, and then enter the task decomposition steps, and collect knowledge based on the decomposed action primitives. After completing the storage of the local knowledge base, upload and update the cloud knowledge base. When reusing knowledge, the local server will first decompose the target task, and then perform similar task matching based on the action meta-model. If there is not enough knowledge available locally, it will perform a cloud knowledge query. After finding enough action primitives, the edge server will combine tasks and complete the configuration of local resources. As for the establishment of the ontology model and the task decomposition scheme, it will be described in the latter part of this section.

3.1. Establishing the Semantic Model of Manufacturing Resources

Intelligent manufacturing's physical resources include equipment used in a series of intermediate processes such as product processing, packaging, and transportation. The knowledge sharing realization requires organizing various links between manufacturing resources into a unified knowledge base to facilitate knowledge storage and recall. This section introduces the integrated ontology and semantic modeling of manufacturing resources.

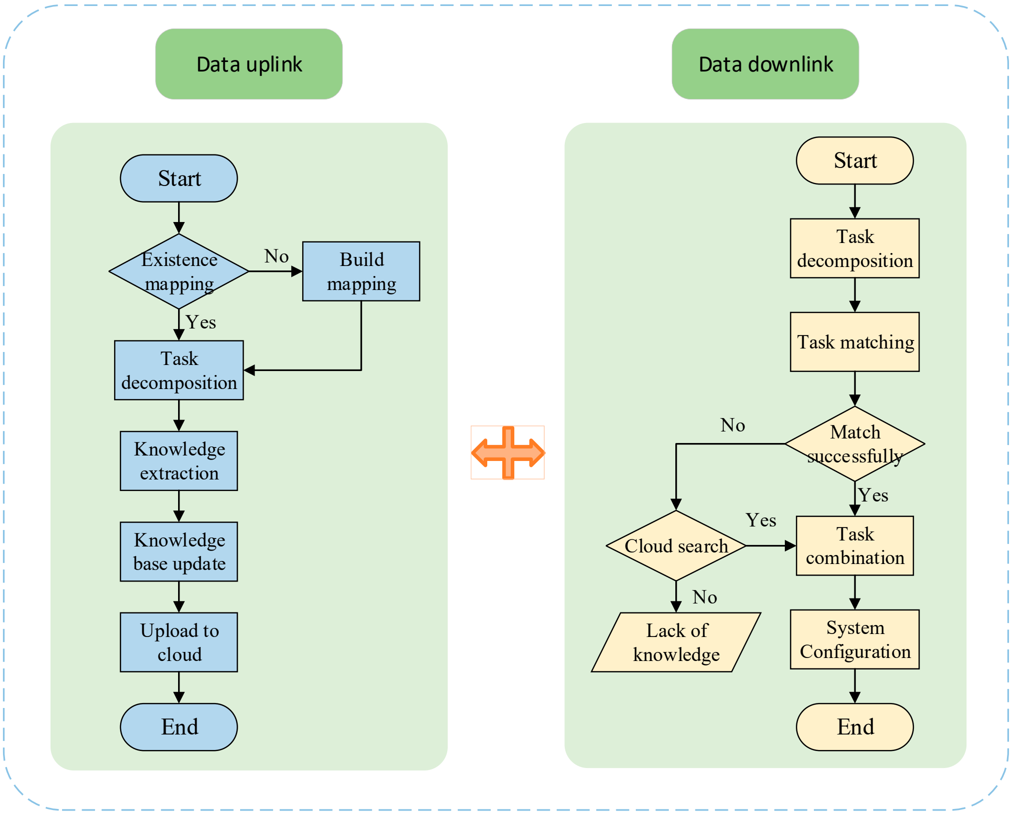


Figure 2. System operation data flow.

The manufacturing resources' ontology structure can be represented by a four-tuple {C, R, S, P} [20], where *C* is a set of ontological classes, *R* represents a set of relations between the ontological classes, *S* denotes a set of hierarchical relationships between classes (that is, the inheritance and realization of class relationships), and *P* is a set of other relationships between classes. Consider, for example, an ontology model $O_1 = \{(x_1, x_2, x_3, x_4), (r_1, r_2), ((x_1, x_2), (x_1, x_3), (x_3, x_4)), (r_1, (x_2, x_3); r_2, (x_2, x_4))\}$. Here, the classes involved are x_1, x_2, x_3, and x_4. According to the hierarchical relationship set, x_2 and x_3 are both subclasses of x_1, while x_4 is a subclass of x_3. In addition to inheritance and implementation between classes, there are two other relationships, r_1 and r_2, of which r_1 is the relationship between x_2 and x_3, and r_2 is the relationship between x_2 and x_4.

The central concept in the manufacturing resources' ontology is a manufacturing task, abstracted in accordance with the four-tuple definition. Based on this abstraction, the corresponding semantic model is constructed, as shown in Figure 3. The task-based ontology model *(Task-Based Knowledge*, *TBK)* is defined as:

$$TBK = \{Environment, Actions, PhysicalResources, ResultEvaluation\}, \tag{1}$$

and comprises the manufacturing resource objects (*PhysicalResources*), environmental factors (*Environment*), execution actions (*Actions*), and effect evaluation (*ResultEvaluation*) involved in the execution process for conceptual refinement. The objects in the manufacturing process include raw materials and manufacturing resources. The attributes used include the equipment's name, number, physical size, capabilities, and the processing operations that can be performed. Environmental factors record the manufacturing task's processing environment, such as space, position, and the manufacturing process requirements regarding, for example, temperature, light, and air quality. The execution action module includes the action level classification, the action sequence combination, and the processing procedure record, thus covering the process's sequence information and state. Finally, the effect evaluation results from the previous executions using the current knowledge and includes each action's execution time cost, product quality, and time consumed.

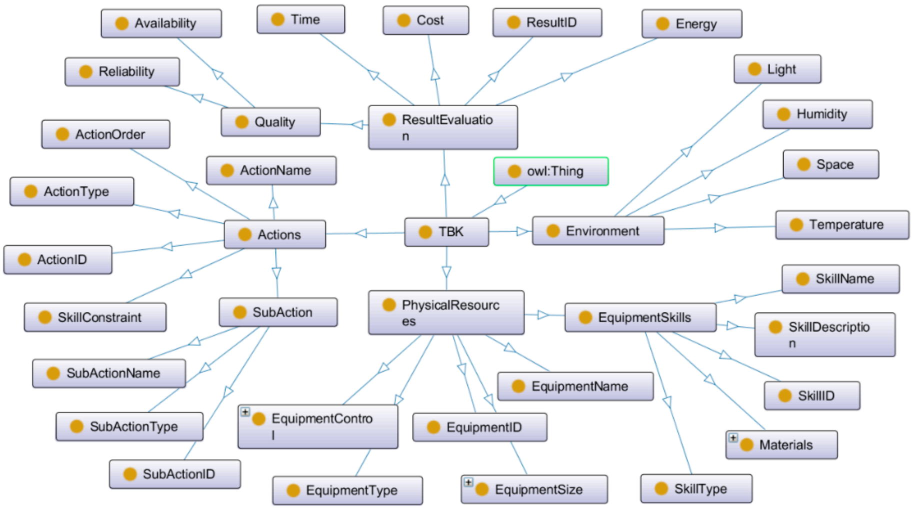


Figure 3. Task-based ontology model.

3.2. *Task Model Decomposition*

Each product's manufacturing process includes multiple manufacturing resources' action coordination. To extract and apply the manufacturing process experience, task and action models need to be decomposed, hierarchically divided, and described by a basic combination of action primitives. Additionally, the manufacturing process provides a clear hierarchical structure for subsequent knowledge reuse. This work embodies the hierarchical structure and logical relationship between the actions by ontologically modeling the manufacturing process's actions.

Specifically, as shown in Figure 4, the execution actions' levels can be divided into the following: task-based actions (*ActionsOnTask*), station-based actions (*ActionsOnStation*), and the bottommost action primitives (*PrimitiveActions*). Task-based actions state the manufactured product's purpose or a function. Station-based actions are derived from the higher-level actions according to the processing procedure and are combined with the manufacturing process's characteristics. The actions are defined for each manufacturing station, and the action primitives are used for processing. The process is refined and decomposed, and actions are decomposed and combined from the manufacturing resources' level. Reusable action control information is obtained by decomposing different manufacturing processes to the primitive level and then reorganizing the action primitives to construct the new task.

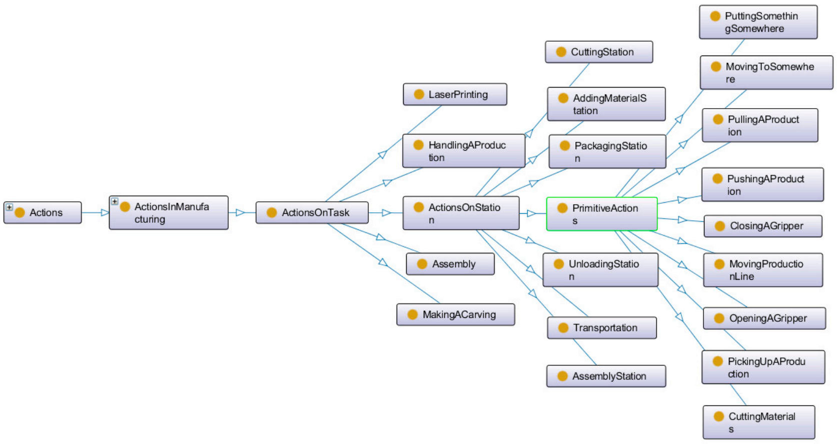


Figure 4. Action model hierarchy.

As an example, Figure 5 outlines a simplified engraving processing task. The task's execution needs to be based on the material information, the processing object's model, the processing equipment model, and the manufacturing process information. From the user's order, through a series of processes such as order arrival, loading, cutting, product packaging, and unloading, a product that meet the user's needs is obtained. These processes are engraving process's sub-tasks from the hierarchical structure perspective. Different processing procedures consist of various multi-action manufacturing resources' execution processes (e.g., the cutting procedures' subdivision). From the material's arrival to the end of the processing and returning to the pallet, it is necessary to move the manipulator and clamp the material successively. The material is obtained, placed, processed using the manufacturing tools, taken out, and finally put back into the pallet, forming a series of non-decomposable, primitive actions in the processing procedure. Thus, the complex manufacturing process involving multiple resources is decomposed into tasks and becomes a combination of action primitives, each corresponding to a single action of an individual device. Each new processing task is decomposed and refined, and the original action primitive combination is used to reuse the original manufacturing resource knowledge.

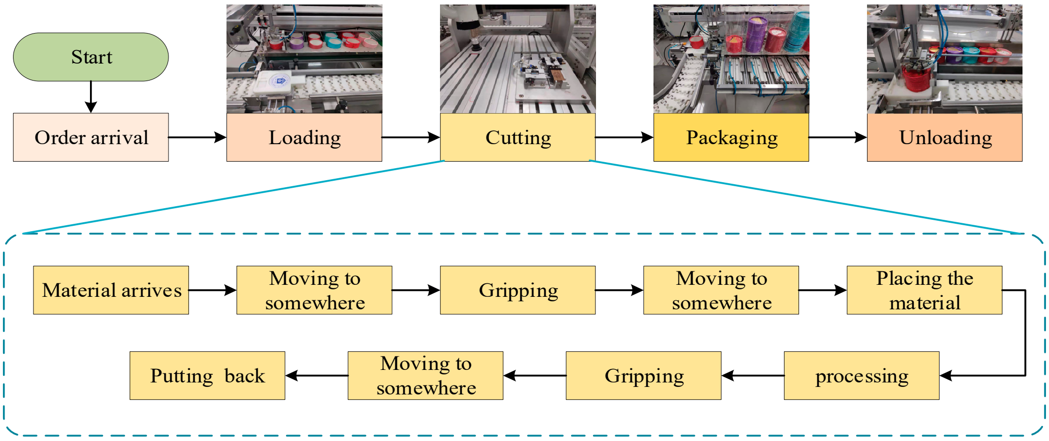


Figure 5. Simplified engraving tasks.

4. Knowledge Reuse Matching

When the intelligent manufacturing production line is altered due to equipment failure or the addition of new manufacturing resources to the process, it is necessary to study and learn the manufacturing resource processing experience. Thus, similar processing tasks in the knowledge base should be extracted, which requires establishing a task matching mechanism. The proposed task-oriented manufacturing resource ontology forms a basis for the matching mechanism that enables knowledge reuse. This section analyzes such a mechanism from the perspective of between-device knowledge exchange and finds new application scenarios for the original resource operation rules.

Complete manufacturing process matching involves considering multiple task model levels. Refining the search objects to action primitives improves the knowledge reusability and cross-domain practicability. To this end, this article adopts the breadth-first search method. Specifically, when a new manufacturing task is generated or a new resource is added to production, the mechanism first obtains the processing knowledge from the module where the resource is located. A similar task experience is then searched and, if found, used to build a direct learning data stream in the module. If no similar processing modules are found, the cloud platform is connected to search for similar tasks. The search algorithm's pseudo-code implementation is given in Algorithm 1, and its explanation is as follows.

Algorithm1. Knowledge reuse similarity matching algorithm

```
Input: Target task, set of weights {$w_1$, $w_2$, $w_3$, $w_4$, $w_5$}
Output: Task model with the highest matching degree
Start:
    for $i \leftarrow 1$ to m do
        $\vec{\mathbf{v}} = (r, s, a, e) \rightarrow \vec{\mathbf{v}}_{\mathrm{h}} = (r_h, s_h, a_h, e_h)$
        if $(r_{hi} == 0)$ then
            $w_n * r_{hi} = -w_n$
        else
            $\boldsymbol{w_n * r_{hi} = w_n}$
        end if
        $(r_h, s_h, a_h, e_h) \rightarrow (w_r r_h, w_s s_h, w_a a_h, w_e e_h)$
        $Q_n = w_r r_h + w_s s_h + w_a a_h + w_e e_h$
        if $(Q_{ni} > 0)$ then
            $S_{ni} = 1$
        else
            $S_{ni} = 0$
        end if
        for $j \leftarrow 1$ to $n$ do
            Calculate the Hamming distance
            if $\boldsymbol{Dis} \leq \mathbf{3}$ then
                $c_j \in C_l$
                Evaluate the execution effect
            end if
        end for
        Extract matching action primitives
        Update (model);
    end for
end function
```

The rapid matching mechanism for knowledge reuse assumes that the target task is composed of m action primitives. Let n denote the number of action primitive models on the edge of the knowledge base. The algorithm searches for the task model with the highest matching degree. At the algorithm's start, the edge server decomposes the target task and traverses the action primitives. While learning action primitives, the feature vectors $\vec{\mathbf{v}}$ are extracted based on action i's ontology model and include the equipment resources r, related skills s, action description a, and environmental conditions e. Thus, the vector can be expressed as $\vec{\mathbf{v}} = (r, s, a, e)$. The ontology model involves various data types. To facilitate subsequent processing and comparison, a variety of data type conversions are realized by calculating the feature vector's hash value, and the feature vector is converted into an n-bit digital signature r_h composed of zeros and ones. The feature vectors' degree of influence in knowledge learning is different, as indicated by the associated weight values. The set of weights is denoted {w_1, w_2, w_3, w_4, w_5}. If the weighting process encounters a non-zero value (i.e., a one), the hash value is multiplied by the weight w_n. Otherwise, the weight w_n is multiplied by -1. After the weighting calculation is completed, it is added vertically and merged into an n-bit sequence string signature Q_n. After dimensionality reduction, the action primitive's standardized signature S_n is obtained. The Hamming distance *Dis* between the standardized signatures is utilized to judge the similarity between the actions. Once a similar action sample set C_l is found, the knowledge reuse impact is evaluated on qualified samples. The action primitives with high similarity and good execution effect should be reused. Each action primitive's execution plan is determined and then reorganized to obtain the task-oriented execution plan's completeness. The definition of all symbols is shown in Table 1.

Table 1. Definition of Symbols.

Symbol	Definition
$\vec{v}$	Feature vector
i	Action
r	Resource
s	Skill
a	Action Description
e	Environment
r_h	Digital signature
w_n	Weight
Q_n	Serial signature
S_n	Standardized signature
C_l	Action collection

5. Experiment

This section uses a multi-variety, personalized, and customized mixed production line prototype platform to verify the feasibility of the proposed cloud-edge collaborative mechanism for knowledge sharing between manufacturing resources. The conducted experiment contrasts the newly added equipment's technological processes after learning the manufacturing and processing cycle with those of the original equipment.

5.1. Prototype Platform and Manufacturing Tasks

A prototype platform of a multi-variety, mixed-flow intelligent manufacturing production line is shown in Figure 6. The main system workflow is as follows. First, the user utilizes a web page or a mobile app to select the product's type, quantity, and personalized pattern. The user order is completed in the cloud. Once recorded, the order is issued directly to the manufacturing resource edge, generating the control flow related to the processing task. The control flow drives each manufacturing resource to perform the current processing task. The equipment utilization is optimized through a high degree of coordination and knowledge obtained in previous processing operations. The manufacturing production line's efficiency is reflected in the multi-task collaborative production and dynamic production line reconfiguration. The mixed-flow manufacturing functions incorporated in the prototype platform to date include, for example, custom wood carving crafts processing, personalized U disk customization, and Bluetooth remote control assembly. In the prototype platform, Robot1 and CNC1 are matched, as well as Robot2 and CNC2, and are responsible for the clamping and engraving process of wood carving originals. The laser machine performs laser printing according to the pattern and text selected by the user when placing the order. Loading machine loads the product packaging/box. Robot3 and Robot4 cover the processed product packaging, whereas Robot5 uses machine vision to select the required raw materials for loading operations. The final products in the packaging box are uniformly unloaded. The specific functions of manufacturing resources are shown in Table 2. The prototype platform involved in this experiment is a multi-variety mixed-flow platform, so the order of each station in the processing process cannot only satisfy a single product manufacturing process. To solve this problem, we have adopted a method of station matching. When the product passes through the station, the product manufacturing information will be read. If the product needs to be processed by this station, it will enter the corresponding production branch line, and if it is not needed, it will move directly to the next station.

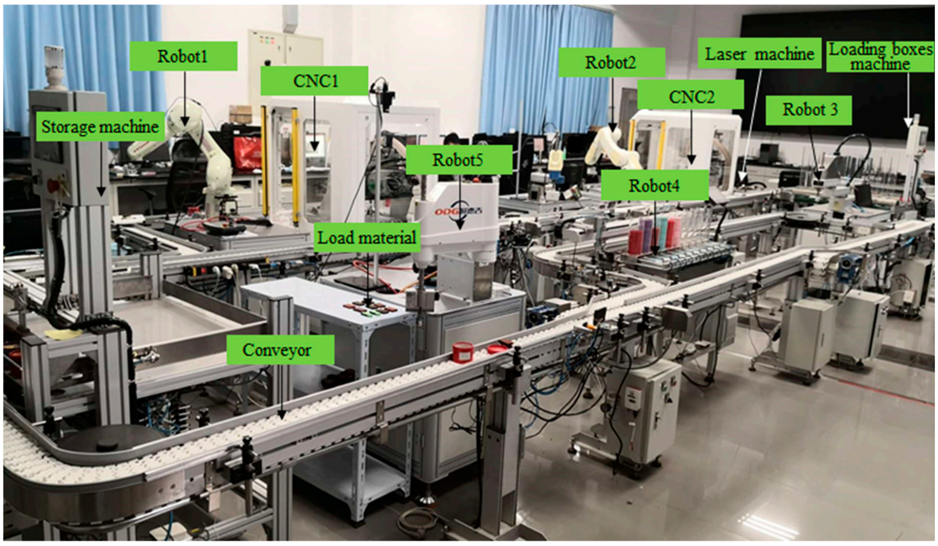


Figure 6. Prototype platform for mixed production line.

Table 2. Manufacturing resource function description.

Manufacturing Resource	Function Description
Robot1(2)	Pick up the raw materials into the CNC1(2) and put the finished products back on the conveyor belt
Robot3	Put the processed product in the box
Robot4	Select the box cover that matches the box and perform the capping operation
Robot5	Select raw materials and add to the conveyor belt
Laser machine	Laser printing personalized patterns
Loading boxes machine	Add boxes to the conveyor belt
Storage machine	Unload the packaged product
CNC1(2)	Carve the raw materials

The experiment starts by the loading machine that reads the products' order information. The loading machine writes the packaging boxes' RFID tags, including the involved process and the corresponding processing actions. The pallets carrying the packaging boxes and materials pass through various stations. The RFID tags are read at the station, and the processing actions that can be performed at the current station are executed. The relationship between the physical manufacturing resources in the prototype platform and the action primitives that can be executed is shown in Table 3. It can be seen from the table that each physical resource processing can be regarded as a collection of actions, and the action primitives it contains have a similar composition structure.

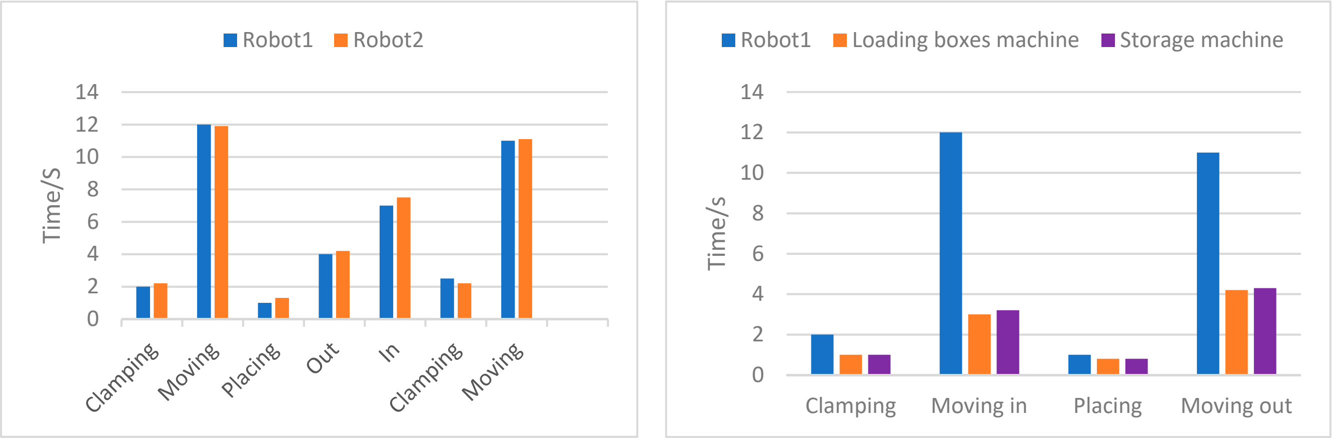
Table 3. Correspondence between manufacturing resources and action primitives ("+"means "contain").

Robot ID	Robot1	CNC1	Laser	Loading	Robot3	Robot5	Storage
Moving	+			+	+	+	+
Clamping	+	+		+	+	+	+
Placing	+			+	+	+	+
Cutting		+					
Printing			+				
Absorbing					+		

5.2. Experimental Results and Discussion

The presented multi-variety, mixed production line serves as an experimental platform. The equipment is manually configured and debugged to generate the equipment's initial processing experience (i.e., initial input). Once the ontology modeling on the manufacturing resources' edge is completed, the equipment that has not been manually debugged is configured by relying on the existing knowledge. More precisely, the existing process data model and control instructions are automatically configured. The proposed knowledge sharing mechanism's effectiveness is verified by the experimental results that compare the manually configured action cycle with the automatically configured result.

Since Robot1 and Robot2 have similar processing procedures, the two manipulators' action cycles can be read separately for comparison, as shown in Figure 7. Robot1 is manually configured and debugged, and Robot2 is automatically configured based on the knowledge sharing mechanism. The manipulators' main processing procedure can be detailed and decomposed into: conveyor belt material clamping, moving the material to the processing equipment CNC, placing the material, exiting the CNC, waiting for the CNC to enter after the processing is completed, clamping the processed material, and moving the material. The raw materials arrive at the conveyor belt pallet. Figure 7a demonstrates that there is only a slight difference between the two manipulators' action cycles, indicating that Robot2 successfully learned Robot1′s existing processing knowledge and experience. Nevertheless, certain differences in the configuration results can be found. In particular, Robot2 chose a more efficient execution plan through comparing the two actions' execution effect evaluations during the learning process. In order to reflect the reduction degree of knowledge reuse, we define the reduction degree $S_i = \frac{|T_i - T_i|}{T_i}$ of action i, where T_i represents the action execution cycle of the knowledge source, and T_i represents the action execution cycle of the device after learning the existing knowledge. The reduction degree of the whole process adopts a weighted average $\sum S_i \cdot \frac{T_i}{T_a}$, where T_a represents the total processing time of the whole process. Combining the calculation of the specific execution cycle data of each action, it can be concluded that the reduction degree of Robot2 for the same task of Robot1 is 95.8%.



(**a**) (**b**)

Figure 7. (**a**) Action cycles' comparison for similar manufacturing processes. (**b**) Multi-process action cycle comparison.

The action cycles of Robot1, loading machine, and storage machine are selected for comparison. The specific data is shown in Figure 7b. Robot1 and loading machine were manually configured and debugged, while the storage machine's process was learned through the local knowledge base. The figure shows data reading and comparison of the three devices' action primitives. The experimental results demonstrate that the storage machine's action cycle is very similar to that of the loading machine but significantly differs from Robot1′s. The storage machine and the loading machine deal with the product's outer packaging. Thus, these machines' operation accuracy is lower than the value of Robot1 when gripping the engraving materials. Consequently, Robot1 has a lower processing speed. Regarding the equipment movement, the storage machine and the loading machine are responsible for the round-trip conveyor belt and the raw material warehousing or the finished product warehousing. These machines do not involve cooperation with other equipment. Robot1 needs to place the engraving materials into the CNC processing area accurately. The CNC's airtightness further restricts Robot1′s movement. The three equipment are similar regarding the placement actions: the only difference being the processed object's size, the action cycles do not significantly differ. Following the analysis of the equipment learning process, it can be concluded that the knowledge reuse process is not a straightforward application, but a matching search for similar processing tasks to extract the knowledge that meets the similarity requirements.

Analyzing the results of the experiment, we can find that: (1) During the execution of this experiment, the collection, processing and knowledge extraction of manufacturing resource information are all completed by the intelligent gateways and servers deployed at the edge. The knowledge used in the knowledge reuse stage partly uses the knowledge extracted on the spot, and the system supplements the knowledge through the cloud for the parts that are not included in the learning samples input from the edge. The entire platform can directly carry out multi-variety mixed-flow processing after learning. The successful operation of the platform fully demonstrates the rationality of the process and the effectiveness of the cloud-edge combined knowledge reuse system used in this article. (2) It can be seen from the experimental results in Figure 6 that under the same task scenario, the knowledge reuse method we proposed has high accuracy, and the quantitative index of action reduction effect reaches 95.8%. The high reduction of the action proves that the knowledge matching algorithm we proposed is reliable. (3) In the second set of experiments, the selected equipments' functions are different, but because we have adopted the task decomposition strategy in the process of knowledge extraction and reuse, more fine-grained knowledge decomposition makes it possible to learn between different tasks. The learning of action control also confirms the superiority of our proposed task decomposition strategy.

6. Conclusions and Future Work

In the face of the current growing demand for personalized customization in the manufacturing field, the importance of dynamic reconfiguration of production lines is becoming more and more important. To solve the problem of knowledge reuse of manufacturing re-source processing experience in the production process, this paper builds on the existing cloud-edge collaboration architecture and deploys services for manufacturing process tasks on the edge side, reducing the system's dependence on the cloud; the task decomposition scheme makes knowledge reuse granular and enhances the versatility of knowledge; the knowledge matching algorithm has high accuracy and can well meet the needs of the existing manufacturing environment. In the case study, we manually configure part of the equipment on the production line to achieve the effect of learning sample input. After the knowledge extraction is completed on the edge side, the knowledge is learned and reused according to the requirements of each equipment processing task. The experimental results selected two groups of control groups for comparison. The experimental results well prove the effectiveness of our proposed scheme and reflect the accuracy of the learning process through the action reduction degree, and through the learning between different tasks, it reflects the improvement of the generality of knowledge by task decomposition.

However, there are still some noteworthy issues in this study that need to be mentioned. First of all, regarding the establishment of the manufacturing process knowledge model, this article only proposes a reference plan for specific processing tasks and scenarios. If you want to apply it to large-scale practical applications, you need to establish a unified standard for the model. Secondly, in terms of knowledge matching, the initial primitive knowledge is heavily dependent, and when there is insufficient knowledge accumulation on the cloud and edge, new knowledge primitives need to be manually input. In the follow-up research, because of the above problems, there can be a clearer research plan: combining the existing knowledge graph research results, in the process of knowledge extraction, the existing sample data information is transformed into a semantic model and a knowledge graph. When a new data source is added, the mapping relationship of data nodes can be established based on the information on the existing knowledge graph, and a more complete knowledge system can be formed through the combination of ontology modeling and knowledge graphs. In terms of knowledge derivation, the method of entity alignment is adopted to quickly reuse knowledge and transfer knowledge between different fields and enhance the universality of knowledge.

Author Contributions: Conceptualization, X.W. and J.W.; methodology, J.W.; software, X.W.; validation, X.W. and J.W.; formal analysis, X.W.; investigation, X.W.; resources, X.W.; data curation, X.W.; writing—original draft preparation, X.W.; writing—review and editing, X.W.; visualization, J.W.; supervision, J.W.; project administration, J.W.; funding acquisition, J.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Joint Fund of the National Natural Science Foundation of China and Guangdong Province (No. U1801264), and the Natural Science Foundation of Guangdong Province, China (No. 2021A1515011946).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Acknowledgments: The authors thank Guangdong Provincial Key Laboratory of Technique and Equipment for Macromolecular Advanced Manufacturing.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Al-Fuqaha, A.I.; Guizani, M.; Mohammadi, M.; Aledhari, M.; Ayyash, M. Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 2347–2376. [CrossRef]
2. Atzori, L.; Iera, A.; Morabito, G. The internet of things: A survey. *Comput. Netw.* **2010**, *54*, 2787–2805. [CrossRef]
3. Wan, J.; Yang, J.; Wang, Z.; Hua, Q. Artificial Intelligence for Cloud-Assisted Smart Factory. *IEEE Access* **2018**, *6*, 55419–55430. [CrossRef]
4. Dillon, T.; Wu, C.; Chang, E. Cloud computing: Issues and challenges. In Proceedings of the 2010 24th IEEE International Conference on Advanced Information Networking and Applications, Perth, WA, Australia, 20–23 April 2010; pp. 27–33.
5. Shi, W.; Cao, J.; Zhang, Q.; Li, Y.; Xu, L. Edge Computing: Vision and Challenges. *IEEE Internet Things J.* **2016**, *3*, 637–646. [CrossRef]
6. Salman, O.; Elhajj, I.; Kayssi, A.I.; Chehab, A. Edge computing enabling the Internet of Things. In Proceedings of the 2015 IEEE 2nd World Forum on Internet of Things (WF-IoT), Milan, Italy, 14–16 December 2015; Institute of Electrical and Electronics Engineers (IEEE): Greenvile, SC, USA, 2015; pp. 603–608.
7. Wan, J.; Li, X.; Dai, H.-N.; Kusiak, A.; Martinez-Garcia, M.; Li, D. Artificial-Intelligence-Driven Customized Manufacturing Factory: Key Technologies, Applications, and Challenges. *Proc. IEEE* **2021**, *109*, 377–398. [CrossRef]
8. Suganuma, T.; Oide, T.; Kitagami, S.; Sugawara, K.; Shiratori, N. Multiagent-Based Flexible Edge Computing Architecture for IoT. *IEEE Netw.* **2018**, *32*, 16–23. [CrossRef]
9. Jiang, C.; Wan, J. A Thing-Edge-Cloud Collaborative Computing Decision-Making Method for Personalized Customization Production. *IEEE Access* **2021**, *9*, 10962–10973. [CrossRef]
10. Surbier, L.N.; Alpan, G.l.N.; Blanco, E. A comparative study on production ramp-up: State-of-the-art and new challenges. *Prod. Plan. Control* **2014**, *25*, 1264–1286. [CrossRef]
11. Li, X.; Li, D.; Wan, J.; Liu, C.; Imran, M. Adaptive Transmission Optimization in SDN-Based Industrial Internet of Things With Edge Computing. *IEEE Internet Things J.* **2018**, *5*, 1351–1360. [CrossRef]
12. Qi, Q.; Tao, F. A Smart Manufacturing Service System Based on Edge Computing, Fog Computing, and Cloud Computing. *IEEE Access* **2019**, *7*, 86769–86777. [CrossRef]
13. Wan, J.; Chen, B.; Wang, S.; Xia, M.; Li, D.; Liu, C. Fog Computing for Energy-Aware Load Balancing and Scheduling in Smart Factory. *IEEE Trans. Ind. Inform.* **2018**, *14*, 4548–4556. [CrossRef]
14. Yang, C.; Lan, S.; Wang, L.; Shen, W.; Huang, G.G.Q. Big Data Driven Edge-Cloud Collaboration Architecture for Cloud Manufacturing: A Software Defined Perspective. *IEEE Access* **2020**, *8*, 45938–45950. [CrossRef]

15. Li, X.; Zhuang, P.; Yin, C. A metadata based manufacturing resource ontology modeling in cloud manufacturing systems. *J. Ambient. Intell. Humaniz. Comput.* **2018**, *10*, 1039–1047. [CrossRef]
16. Yang, T.-H.; Ko, C.-S.; Huang, J.-Y.; Lee, W.-P. Configuring reusable robot services in a cloud environment. In Proceedings of the 2015 IEEE International Conference on Robotics and Biomimetics (ROBIO), Zhuhai, China, 6–9 December 2015; Institute of Electrical and Electronics Engineers (IEEE): Greenvile, SC, USA, 2015; pp. 225–230.
17. Tenorth, M.; Beetz, M. Exchanging action-related information among autonomous robots. In *Intelligent Autonomous Systems 12*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 467–476.
18. Chhim, P.; Chinnam, R.B.; Sadawi, N. Product design and manufacturing process based ontology for manu-facturing knowledge reuse. *J. Intell. Manuf.* **2019**, *30*, 905–916. [CrossRef]
19. Wan, J.; Li, J.; Imran, M.; Li, D.; Amin, F.-E.-A. Blockchain-Based Solution for Enhancing Security and Privacy in Smart Factory. *IEEE Trans. Ind. Inform.* **2019**, *15*, 3652–3660. [CrossRef]
20. Knoblock, C.A.; Szekely, P.; Ambite, J.L.; Goel, A.; Gupta, S.; Lerman, K.; Muslea, M.; Taheriyan, M.; Mallick, P. Semi-automatically mapping structured sources into the semantic web. In *Extended Semantic Web Conference*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 375–390.