information

MDPI

Article

A Novel Approach of Resource Allocation for Distributed Digital Twin Shop-Floor

Haijun Zhang, Qiong Yan *, Yan Qin, Shengwei Chen and Guohui Zhang

School of Aerospace Engineering, Zhengzhou University of Aeronautics, Zhengzhou 450015, China
* Correspondence: zhanghjy@gmail.com

Abstract: Facing global market competition and supply chain risks, many production companies are leaning towards distributed manufacturing because of their ability to utilize a network of manufacturing resources located around the world. Deriving from information and communication technologies and artificial intelligence, the digital twin shop-floor (DTS) has received great attention from academia and industry. DTS is a virtual shop-floor that is almost identical to the physical shop-floor. Therefore, multiple physical shop-floors located in different places can easily be interconnected to realize a DT that is a distributed digital twin shop-floor (D2TS). However, some challenges still hinder effective and efficient resource allocation among D2TSs. In order to attempt to address the issues, firstly, this paper proposes an information architecture for D2TSs based on cloud–fog computing; secondly, a novel mechanism of D2TS resource allocation (D2TSRA) is designed. The proposed mechanism both makes full use of a digital twin to support dynamic allocation of geographic resources and avoids the centralized solutions of the digital twin which lead to a heavy burden on the network bandwidth; thirdly, the optimization problem in D2TSRA is solved by a BP neural network algorithm and an improved genetic algorithm; fourthly, a case study for distributed collaborative manufacturing of aero-engine casing is employed to validate the effectiveness and efficiency of the proposed method of resource allocation for D2TS; finally, the paper is summarized and the relevant research directions are prospected.

Keywords: digital twin; distributed manufacturing; resource optimal allocation; cloud–fog–edge computing; shop-floor scheduling

Citation: Zhang, H.; Yan, Q.; Qin, Y.; Chen, S.; Zhang, G. A Novel Approach of Resource Allocation for Distributed Digital Twin Shop-Floor. *Information* **2023**, *14*, 458. https://doi.org/10.3390/info14080458

Academic Editors: Vasco N. G. J. Soares, João M. L. P. Caldeira, Bruno Bogaz Zarpelão and Jaime Galán-Jiménez

Received: 13 July 2023
Revised: 31 July 2023
Accepted: 4 August 2023
Published: 13 August 2023

1. Introduction

With the increasing challenges of global economy, companies have to continue to make the switch to more flexible, agile, and intelligent manufacturing paradigms in order to meet the changing market conditions—including responses to the coronavirus pandemic, low-volume customized products, global supply chain issues, pressure to reduce costs, and demand fluctuations. Distributed manufacturing is apprehended as the ideal manufacturing approach in the field of production science [1,2] by leveraging and coordinating localized production facilities (e.g., shop-floor) that are connected to each other via advanced information and communication technologies. With a DM model, manufacturers can spread out discrete manufacturing components across geographically dispersed facilities instead at a single shop-floor, enabling them to produce closer to target customer, low labor cost, or raw materials areas. Nomenclature lists all abbreviations used in the paper to enhance reader comprehension and maintain conciseness.

Despite the considerable research to understand DM along with the associated advantages and implementation barriers, the resource allocation research oriented to DM is still lacking dynamic adaptation and reconfiguration methods during runtime (e.g., addition or removal of machines, reconfiguration of existing machines) to accommodate the continuously changing demand in production or due to failures [3]. Existing research on resource allocation in distributed manufacturing systems has the following two shortcomings:

(1) In terms of resource sharing, traditional distributed manufacturing systems encapsulate manufacturing resources digitally to provide consumers with transparent, on-demand service. These distributed manufacturing platforms failed to consider the requirement of real-time interaction between consumers and manufacturing resources.
(2) In terms of resource allocation, a large amount of research on resource allocation focused on multi-objective optimization models and algorithms. However, manufacturing resources in the research were assumed to be in a static state with immutable properties for the proposed models and algorithms.

Digital twin (DT) is an integrated multi-physics, multi-scale, probabilistic simulation of a vehicle or system that uses the best available physical models, sensor updates, fleet history, etc. to mirror the life of its corresponding flying twin [4]. DT has been recognized as a technology that can seamlessly connect the physical space and cyber space of resources [5]. Grieves [6] (University of Michigan) proposed a concept similar to DT, the Mirrored Space Model, in a Product Life Cycle Management course in 2003. In October 2011, Tuegel et al. [7] published a paper on digital twin, focusing on health diagnosis and prediction of the transportation vehicles required for future deep space exploration. Tao [8] (Beihang University) first proposed digital twin shop-floor (DTS), which was an advanced data-driven manufacturing model with a high-fidelity iterative simulation process integrating physical production facilities and their real-time digital replica in 2017. Distributed digital twin shop-floor (D2TS) consists of some geographically dispersed DTSs that are conveniently connected through the corresponding digital shop-floor in the virtual space. D2TS is ideally suited for the development of DM.

DM needs modern resource allocation for small, flexible, and scalable production units in decentralized production networks [1]. DTS can obviously meet the above requirement of dynamic capabilities for resource allocation in DM. Therefore, this paper proposes a novel approach of resource allocation based on digital twin shop-floor oriented to DM. More specifically, some research questions are of our interest: (1) how to design an effective information architecture for real-time connection of geographically dispersed shop-floors, which is necessary to enable interaction between manufacturing resource providers and consumers, as well as to monitor the manufacturing process. (2) How to realize dynamic resource allocation among geographically dispersed shop-floors, making full use of the characteristics of iterative interaction optimization between physical and virtual spaces in DTS. (3) How to utilize the twin data of a manufacturing resource to estimate the working hours of each operation, select a suitable manufacturing resource to complete each operation, and determine the order of operations to be executed on this resource.

The remainder of this paper is organized as follows. Section 2 reviews related works about distributed manufacturing and digital twin shop-floor. Section 3 designs an information architecture for D2TS. Section 4 proposes an optimization mechanism of resource allocation for D2TS. Section 5 illustrates resource allocation problem modeling and algorithms for working hours estimation and scheduling optimization, which are some of the key technologies for resource optimization allocation. A case study is demonstrated in Section 6 to validate the proposed method. Contributions and future work are described in Section 7.

2. Literature Review

This section reviews existing related research work on D2TS, especially cloud manufacturing and DTS, and describes the research gaps that motivated this research work.

2.1. Cloud Manufacturing

In the last decade, a lot of research and applications have been carried out on distributed manufacturing, such as agile manufacturing [9], green manufacturing [10], and especially cloud manufacturing [11]. The research content of cloud manufacturing can be roughly divided into the following three categories: (1) theory and framework system of cloud manufacturing; (2) key technologies of cloud manufacturing, including encapsulation modeling, discovery matching, optimization selection, exception handling, and so on; (3) research and application of cloud manufacturing platforms.

Li et al. [12] proposed the concept of cloud manufacturing and discussed the characteristics, important definitions, and system architecture of cloud manufacturing. Cheng et al. [13] put forward a hypernetwork-based model introducing the idea of graph coloring and an artificial bee colony algorithm-based method for distributed and collaborative manufacturing service scheduling towards smart manufacturing. Möncha and Shen [14] proposed a greedy randomized adaptive search framework for a parallel machine scheduling with the total weighted delivery time in distributed manufacturing. Carlucci et al. [15] proposed an intelligent decision-making model based on a minority game for resource allocation in cloud manufacturing. Delaram et al. [16] proposed a utility-based matching mechanism for stable and optimal resource allocation in cloud manufacturing platforms using a deferred acceptance algorithm. Yin et al. [17] proposed a device selection method based on constraint characteristics for networked collaborative manufacturing equipment optimization. Wu et al. [18] proposed a data-driven real-time scheduling method for a shop-floor based on a BP neural network. Zhou et al. [19] presented a unified knowledge graph-driven production resource allocation approach for discrete manufacturing workshops, where the knowledge graph model was designed to integrate the engineering semantic information in the machining shop-floor. Szaller et al. [20] introduced a distributed collaboration framework of manufacturing agents, where agents considered trustfulness during the selection from resource-offering agents' proposals and made decisions considering subjective trust and public reputation values. Liu et al. [21] developed an artificial intelligence-assisted distributed system for manufacturing plant-wide predictive maintenance applications, which enabled the data to be processed near the sensors, requiring fewer data to be transmitted to the central cloud server, reducing network delay and delivering more accurate results.

2.2. Digital Twin Shop-Floor

DT promotes a new breed of manufacturing systems comprising automated industrial machines extended with additional sensors and actuators in sensor and/or actuator nodes (SANs). Industrial machines and SANs in DT are governed by their virtual models executed on embedded computers, interconnected via industrial networks or high-level dedicated protocols [3]. The virtual model in DT realizes the functional services and application requirements of design optimization, performance improvement, and predictive maintenance for physical objects by multi-dimensional characterization of the actual behavior of physical entities [22]. Currently, the DT technology has been explored and applied in more than 50 directions in over 10 fields, including industry, agriculture, education, healthcare, transportation, energy, and more [23].

In the research field of DTS, Zhang et al. [24] proposed a modeling and online training method for DTS, aiming at the difficulties in modeling, simulation, and verification. Zhang et al. [25] constructed a software and hardware integrated configuration model of a digital twin manufacturing cell (DTMC) and established a smart contract-based edge–cloud collaborative operation and intelligent control mechanism of the DTMC. Qian et al. [26] proposed a mathematical method for verifying the DTS twin model. Liu et al. [27] proposed an online prediction method of the operation status of the DTS based on real-time data. Hong et al. [28] proposed a real-time detection approach for DTS production action. Liu et al. [29] explored an architecture of the cloud–edge interplay workshop and proposed a blockchain-based data interactive approach to form the peer-to-peer data exchange mechanism between digital twin-based manufacturing systems. Wang et al. [30] presented a real-time digital twin flexible job shop scheduling method with edge computing and an improved Hungarian algorithm. Wang et al. [31] proposed a knowledge graph-based multi-domain model integration method for DTS. Bellavista et al. [32] proposed the application-driven digital twin networking middleware, which can simplify the interaction among heterogeneous devices and dynamically manage network resources in edge industrial environments. Vyskocil et al. [33] explicated an architecture of a distributed manufacturing execution system that is capable of autonomously composing, verifying, interpreting, and executing production plans using digital twins and symbolic planning methods. Liu et al. [34] proposed a novel DT-enabled collaborative data management framework for metal additive manufacturing systems, where a cloud DT communicates with distributed edge DTs in different product lifecycle stages.

2.3. Research Gaps

Manufacturing platforms have changed from a localized and centralized (private) mode to a globalized and decentralized (public) mode [16]. The summary of the above research is presented in Table 1. Many of the prior research papers look to resource allocation as an optimization problem. There is less research on the collaborative relationship between manufacturing participants, the dynamic manufacturing resource scheduling decision, and the utilization of complex information data in this process. There are three major research gaps:

(1) To adapt to dynamic manufacturing processes, solving resource allocation problems in DM needs to ensure real-time interaction between consumers and key geographically dispersed facilities (providers), especially in the typical form of distributed manufacturing—cloud manufacturing. Manufacturing processes are different from computing processes, and many machining resource capabilities cannot be fully encapsulated as digitized, fully automated manufacturing services. Manufacturing services like remote and transparent computing services are not suitable for all machining tasks.

(2) In terms of data transmission, although cloud manufacturing provides an environment for remote resource sharing, there is a lack of real-time and efficient transmission mechanisms for twin data in DTS. The conventional centralized cloud computing-based IoT solutions always lead to a heavy burden on the network bandwidth due to the large amount of sensor data collected frequently that have to be transmitted to the central server and this leads to poor response time for the distributed manufacturing system [21]. Twin data require not only high real-time requirements but also the integration of various factors' data in the shop-floor, as well as the fusion of physical and virtual space data. Therefore, if all manufacturing resources are interconnected and interact through digital twin in a distributed manufacturing platform, there will be a problem of bandwidth competition and severe delay through cloud computing.

(3) In terms of resource allocation, previous research always assumed that the resource allocation systems have all the permissions for manufacturing data, including historical and real-time data of resources (even factory data or shop-floor data). This situation may be possible among shop-floors under the same group, but it is not realistic under different groups. This is because production big data are an important asset and sensitive data of the factory. Although the blockchain technology can ensure the reliability and security of data sharing, one key and difficult point for resource allocation in DM is how to integrate data and when and with whom to share data in order to achieve seamless information flow and decision coordination [35].

The main contribution of this paper is to propose D2TS, which adapts to distributed manufacturing by connecting geographically dispersed DTSs through cloud computing and fog computing technologies to form a self-aware, self-adaptive collaborative network. Fog computing adds storage-enabled servers between the edge layer and the cloud layer to process sensitive data that are not easy to upload to the cloud computing center. Non-critical equipment digital twins use loose links, while critical equipment digital twins use real-time links. The distributed manufacturing collaborative network should allow interoperability between distributed business applications and human operators, so it is necessary to capture complex dynamic situations in the production process in real time and use data for production monitoring, process control, and optimization. This research, combined with cloud–edge technology, can fulfil the scalability of the distributed manufacturing collaborative network and is suitable for distributed manufacturing applications.

Table 1. Summary of literature review.

Ref.	Human–Equipment Interaction	Data Transmission	Resource Allocation
Li et al. [12] (2012) Szaller et al. [20] (2019)	Non-real-time	Distributed	Static
Delaram et al. [16] (2021)	Non-real-time	Centralized	Static
Cheng et al. [13] (2018) Wu et al. [18] (2020) Zhang et al. [24] (2023) Qian et al. [26] (2021) Liu et al. [27] (2021) Wang et al. [31] (2023)	Real-time	Centralized	Dynamic
Hong et al. [28] (2021)	Real-time	Centralized	N/A
Möncha and Shen [14] (2020) Yin et al. [17] (2018)	N/A	Distributed	Static
Carlucci et al. [15] (2020) Zhou et al. [19] (2021)	N/A	Centralized	Static
Zhang et al. [25] (2023) Liu et al. [34] (2022)	Real-time	Distributed	N/A
Liu et al. [21] (2022) Liu et al. [29] (2023) Wang et al. [30] (2023) Bellavista et al. [32] (2021) Vyskocil et al. [33] (2023)	Real-time	Distributed	Dynamic
This paper	Real-time	Distributed	Dynamic

"N/A" indicates that it is not mentioned in the paper.

3. Information Architecture for D2TS

To support distributed manufacturing engineering applications, this paper proposes an information architecture for D2TS based on cloud and fog computing, which mainly includes three layers: resource layer, cloud–fog layer, and application layer, as shown in Figure 1.

Resource Layer: It mainly includes physical manufacturing resources such as equipment, personnel, materials, software, and processes used for product design, machining, and production management. These resources can be either fixed or mobile, hardware or software, and can be operated either in real time or offline. This is the fundamental reason why distributed manufacturing is much more complex and difficult than cloud and fog computing. Manufacturing resources are usually distributed in multiple DTSs located in different geographical locations, but they form thousands of edge nodes through the Industrial Internet of Things (IIoT). They are physical entities of digital twins (including cloud digital twin, edge digital twin) in the application layer, and they are also objects for optimized resource allocation based on digital twin.

Cloud–Fog Layer: It includes cloud nodes and fog nodes. Cloud nodes are composed of high-performance server clusters with powerful computing and storage capabilities, responsible for global resource optimization allocation, processing tasks with high computational complexity and low time sensitivity and big data tasks. Fog nodes include broker node (BN), computing node (CN), and repository node (RN). It should be noted that fog nodes can be customized and constructed according to the scale of manufacturing tasks. BN supports the background (backend) processing of all resource application programs under the fog nodes, such as monitoring the running status of edge nodes' resources (resource running status, resource consumption, service time, etc.). When necessary, they can communicate and interact with other fog nodes or cloud centers on behalf of the fog node. CN is used to process the computing tasks within the fog node, accessed through BN, and collaborate with computing nodes in the fog network to greatly reduce the processing delay of tasks in distributed manufacturing. Based on security policies, not all BNs can accept access requests from other fog nodes. RNs are mainly distributed databases that provide sharing, replication, recovery, and secure storage of data for edge nodes and the fog node; they provide real-time access and analysis of historical data interfaces for local edge nodes and other fog nodes.

Application Layer: It mainly consists of a digital twin-based intelligent manufacturing platform (IMP) that is oriented towards both resource providers and resource consumers. Resource providers are owners of the resources in the resource layer, while resource consumers are the ones who submit production and processing tasks. IMP is the manager responsible for optimizing and configuring the manufacturing resources in D2TS. Digital twin is the bridge among providers, consumers, and IMP. Digital twin in D2TS is divided into two different levels: edge digital twin and cloud digital twin, which are deployed in fog nodes and cloud nodes, respectively. The cloud digital twin is used for global optimization in IMP (e.g., resource allocation in Section 3), while the edge digital twin is used for local simulation optimization and real-time monitoring of resources (e.g., equipment failure prediction). The cloud digital twin is a lightweight and data-simplified version of the edge digital twin and does not require separate development, making DTS easy to plug and play into IMP.

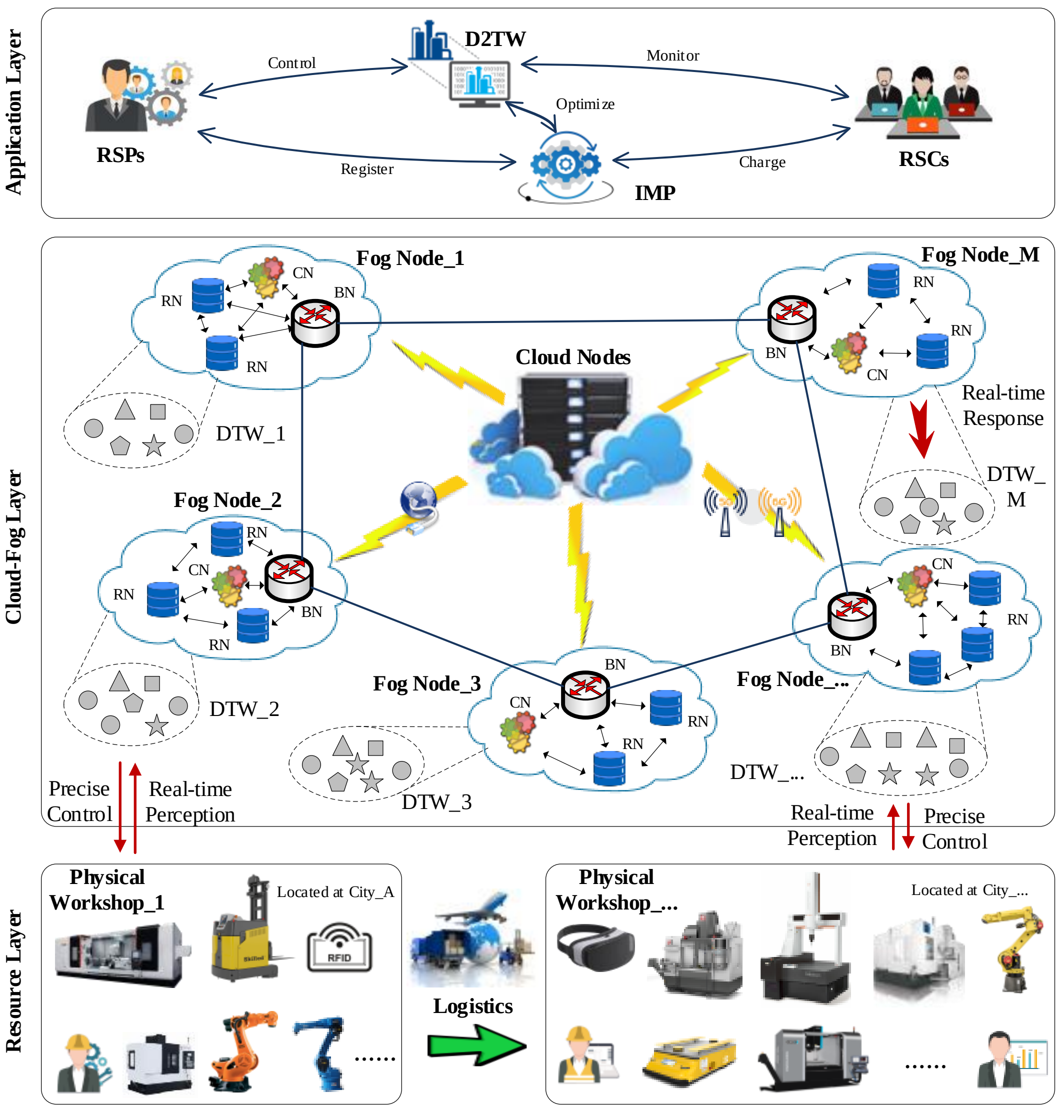

Figure 1. Information architecture for D2TS.

4. Resource Allocation Mechanism for D2TS

With the support of the above information architecture, this section designs a resource allocation mechanism for D2TS. The mechanism mainly involves the cloud computing space (CCS), fog computing space (FCS), and physical resource space (PRS). Data transmission between the three spaces is achieved through IIoT, including industrial fieldbus, industrial Ethernet, Ethernet, fiber optics, time-sensitive networking (TSN), wireless networks (e.g., Zigbee, WIAPA, Wireless HART, ISAI 00.11a, NB-IoT, 5G, 6G), IPv6, etc. Refer to Figure 2 for details, and the UML sequence diagram is shown in Figure 3.

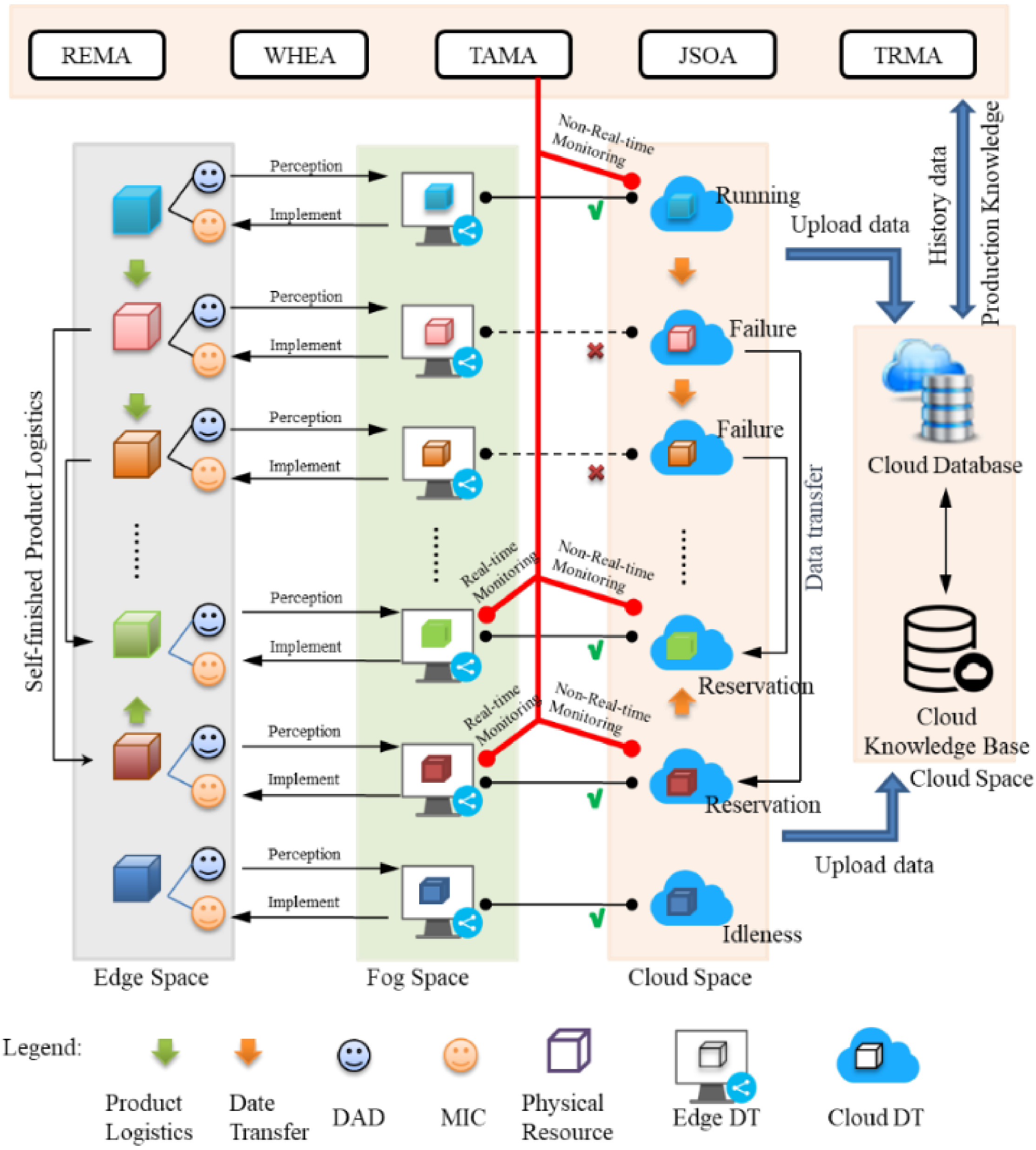

Figure 2. Mechanism of resource allocation for D2TS.

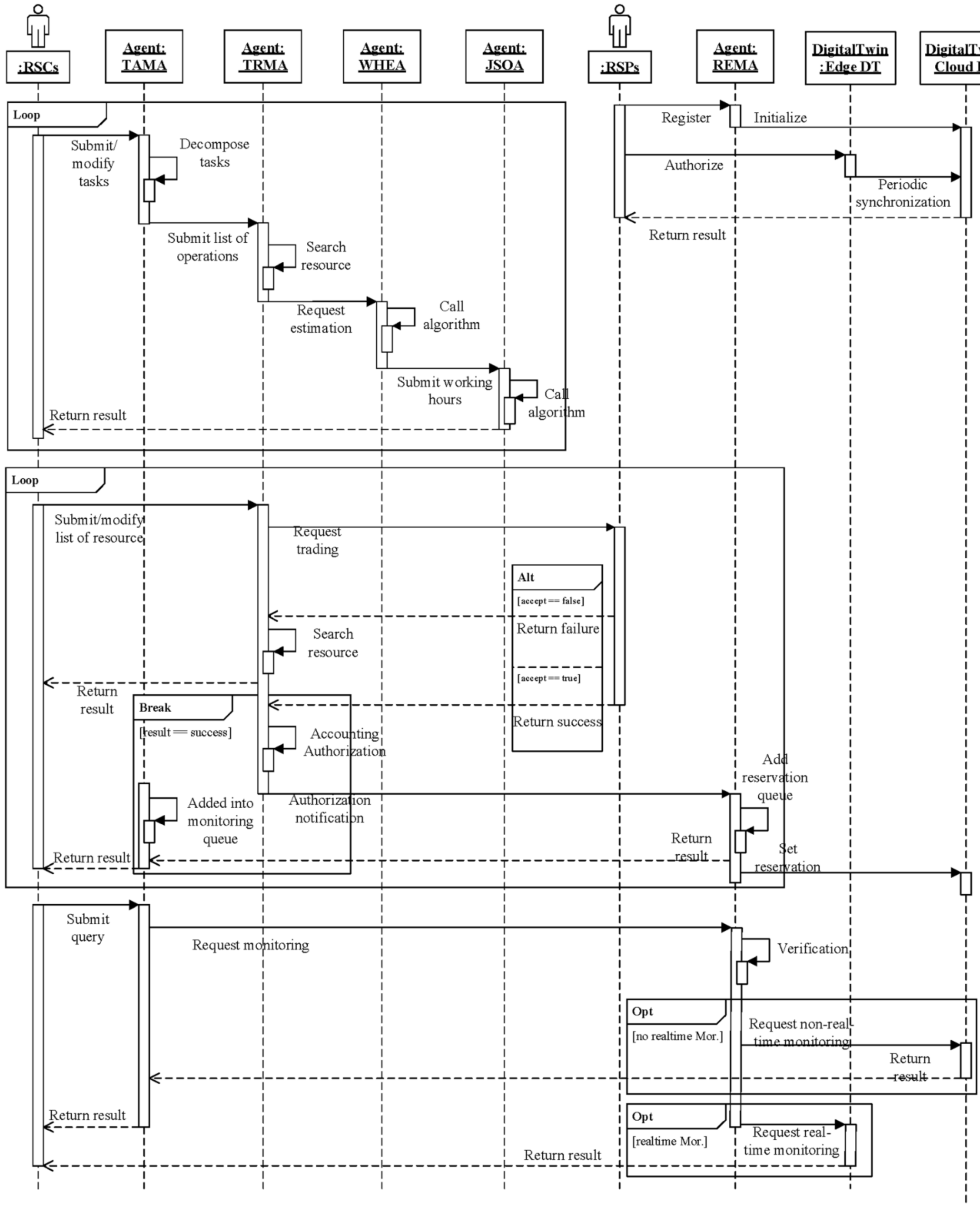

Figure 3. UML sequence diagram for resource allocation for D2TS.

The CCS consists of three parts: cloud digital twin function area (CDTFA), cloud data management function area (CDMFA), and resource optimization allocation function area (ROAFA).

(1) CDTFA: Its main function is to link the corresponding edge digital twin at a customized frequency, in order to obtain the status information of physical resources or production progress information, and upload relevant important data to the cloud

database. The cloud digital twin is deployed in the CCS, which is a simplified and lightweight version of the edge digital twin. It is an image of the manufacturing resources that are not fully informed and does not have the ability to control physical manufacturing resources. Its main function is to provide cloud platforms with resource search, matching, scheduling, billing, credit, big data, and other services.

(2) CDMFA: It includes both the cloud database and cloud knowledge base [36]. This implementation involves partitioning selection, storage, cataloging, and indexing of massive industrial data. With the help of distributed processing architectures, it meets the batch processing requirements of the CDTFA's massive data and the ROAFA's demand for predictive models, intelligent algorithms, process planning, etc., as well as the demand for non-real-time or historical data such as resource trading, credit, quality, cost, and status. The raw data obtained by the cloud digital twin are cleaned to provide high-quality data sources for subsequent storage, management, and analysis.

(3) ROAFA includes some multi-agent modules, such as resource management agent (REMA), task management agent (TAMA), trading management agent (TRMA), working hours estimation agent (WHEA), job scheduling optimization agent (JSOA), etc.

- REMA: It is responsible for providing an entry point for producers to configure the cloud digital twin of manufacturing resources, such as publishing the manufacturing resources they own through semantic description and encapsulation.
- TAMA: It is responsible for the development and modification of process regulations based on cloud computer-aided process planning, as well as the decomposition and combination of complex machining tasks before task execution. During task execution, it provides real-time production monitoring (such as quality) and non-real-time monitoring (such as production progress).
- WHEA: It is responsible for receiving requests from TRMA, predicting the required working hours based on a back propagation neural network (BPNN) model (obtained from the cloud knowledge base), and submitting the results to the JSOA for scheduling optimization.
- JSOA: It is responsible for receiving requests from TAMA. Based on the optimization goals set by the consumers, it calls on the historical transaction data and intelligent optimization algorithms in the cloud database and cloud knowledge base.
- TRMA: It is responsible for resource searching and matching based on the basic task list provided by TAMA before task execution, as well as signing blockchain cooperation agreements. In case of resource failure during task execution, TRMA transfers the processing information to the cloud digital twin node of the replacement resource. After task execution, TRMA settles the order amount with the cloud digital twin that executed the task.

FCS contains a large number and diverse types of edge digital twins of resources that are distributed in different locations. These edge digital twins are located close to the physical manufacturing resources and provide a complete, real-time, digital mirror image of the resources. They can not only perceive information in real time from the physical manufacturing resources but also control the corresponding physical manufacturing resources accurately after simulation, prediction, and optimization analysis. In the physical space, various manufacturing resources are sensed in real time through data acquisition devices such as sensors and servo drivers and precisely controlled through micro industry controllers such as PLCs, DCSs, and FCSs.

5. Problem Modeling and Algorithms of Resource Allocation for D2TS

5.1. Problem Formulation

Suppose a factory receives a small batch of orders with multiple varieties $\{J_1, J_2, \ldots\ldots\}$ but the products in these orders are new prototypes. The factory already owns most of the production equipment but lacks some key processing equipment and cannot complete all processing steps by itself. The factory intends to employ some key processing equipment from other factories through the proposed D2TS method, which is a typical resource allocation problem in the field of distributed manufacturing. This paper will attempt to address the following two issues of resource allocation for D2TS: first, how to estimate the working hours of each operation on a candidate resource for scheduling; second, how to select the appropriate manufacturing resource to complete each operation and determine the order of operations to be executed on this resource.

5.2. Resource Allocation Route

Generally, before processing this batch of orders, the process engineer will develop corresponding process routes $\{(O'_{11}, O'_{12}, O'_{13}, O'_{14}, O'_{15}, O'_{16}), (O'_{21}, O'_{22}, O'_{23}), \ldots\ldots\}$, which serve as the basis for quality inspection, cost accounting, production planning, worker operations, and production preparation (such as procurement or manufacturing of raw materials, purchase or modification of equipment) during the production process. However, such process routes are not suitable for resource allocation in D2TS, such as when certain continuous processes can be completed by the same equipment. For example, CNC machining centers can perform multiple operations, such as turning, milling, and drilling. Therefore, multiple operations $O'_{12}, O'_{13}, O'_{14}, O'_{15}$ can be combined into one operation O_{12} (as shown in Figure 4a). In addition, due to the non-uniqueness of process routes, an operation can also be divided into multiple operations according to actual needs. For example, an operation O'_{22} can be divided into two operations O_{22}, O_{23}, as shown in Figure 4b.

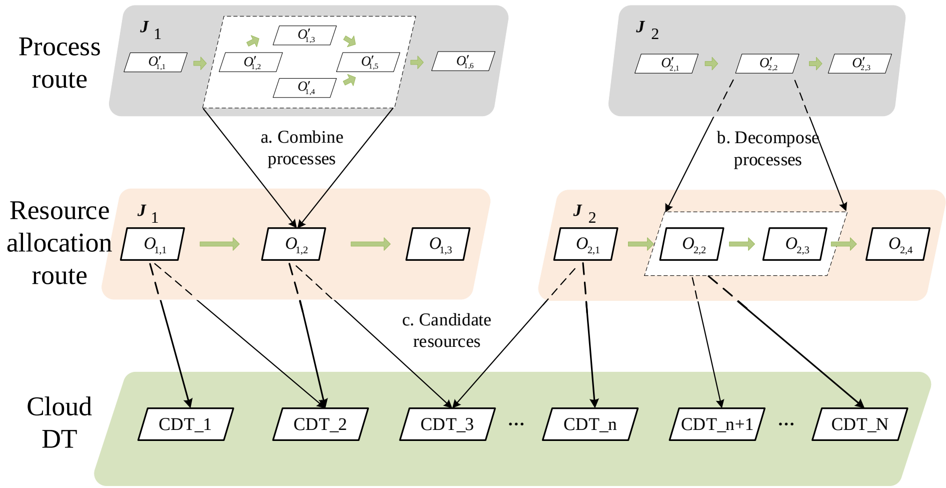

Figure 4. Example of resource allocation route for D2TS.

In this paper, these modified routes are referred to as resource allocation routes. For each operation in the resource allocation route, the TAMA searches and matches cloud digital twins of manufacturing resources and identifies candidate manufacturing resources for each operation, as shown in Figure 4c. It should be noted that each operation may have multiple candidate manufacturing resources that can be used. For example, lathes located in City_A and City_B can both complete turning parts' rough castings with a diameter within 20CM.

5.3. Mathematical Modeling

Without loss of generality, the following assumptions are made for the resource allocation problem in D2TS:

(1) When the cloud digital twin corresponding to the physical manufacturing resource is in a state of failure, it cannot be used as a candidate resource. When the current state is reservation or working, it can be used as a candidate resource, but it is necessary to further obtain the available time slot in real time.

(2) All operations for all tasks are assumed to be serial. This is because in actual production, parallel and reentrant operations can be transformed into serial ones, and equivalent processing operations can be performed with job scheduling and mathematical modeling.
(3) Each machine can only process one workpiece at a time, and phenomena such as equipment abnormalities, defective products during processing, and material shortages are not considered.
(4) Processing time, preparation time, and transportation time may vary depending on the resources used. If two adjacent operations use the same equipment, the preparation time and transportation time are assumed to be zero.
(5) Transportation time between different resources is considered on a per-workpiece (per-processing task) basis.

For convenience of description, the following symbols are defined in Table 2.

Objectives:

$$f_1 = min\left(\underset{1\le i\le NJ}{max}\left(te_{i,NO_i}\right)\right) \tag{1}$$

$$f_2 = min\left(\sum_{i=1}^{NJ}\sum_{j=1}^{NO_i}\sum_{r=1}^{NR} x_{i,j,r}TM_{i,j,r}^2\right) \tag{2}$$

$$f_3 = min\left(\sum_{i=1}^{NJ}\sum_{j=1}^{NO_i-1}\sum_{r=1}^{NR}\sum_{s=1}^{NR} x_{i,j,r}x_{i,j+1,s}TL_{r,s}\right) \tag{3}$$

Subject to:

$$ts_{i,j} + \sum_{r=1}^{NR}\sum_{k=1}^{2} x_{i,j,r}TM_{i,j,r}^k = te_{i,j}$$

where

$$\forall i \in [1, NJ], \forall j \in [1, NO_i] \tag{4}$$

$$te_{i,j} + \sum_{r=1}^{NR}\sum_{s=1}^{NR} x_{i,j,r}x_{i,j+1,s}TL_{r,s} \le ts_{i,j+1}$$

where

$$\forall i \in [1, NJ], \forall j \in [1, NO_i - 1] \tag{5}$$

$$\sum_{r=1}^{NR} x_{i,j,r} = 1$$

where

$$\forall i \in [1, NJ], \forall j \in [1, NO_i] \tag{6}$$

$$te_{i,j} \le ts_{i',j'}$$

where

$$x_{i,j,r} = 1, x_{i',j',r} = 1, y_{i',j',r} = y_{i,j,r} + 1 \tag{7}$$

Formula (1) represents the scheduling optimization objective of minimizing the completion time of all jobs; Formula (2) represents the scheduling optimization objective of minimizing the setup time and processing time; Formula (3) represents the scheduling optimization objective of minimizing the logistics time between process transitions; Formula (4) represents that the process scheduling needs to include setup time and processing time; Formula (5) represents that the process scheduling needs to include transportation time; Formula (6) represents that each process can only use one manufacturing resource at a time;

and Formula (7) represents that each manufacturing resource cannot complete multiple operations at the same time.

Table 2. Notations.

Notations	Description
$NJ = 1,2,3,\ldots$	The total number of manufacturing jobs
$NR = 1,2,3\ldots$	The total number of candidate manufacturing resources
$J = \{J_i \mid i = 1,2,\ldots,NJ\}$	The set of manufacturing jobs to be scheduled, where J_i represents the i-th manufacturing job
$O = \{O_{i,j} \mid i = 1,2,\ldots,NJ;\ j = 1,2,\ldots,NO_i\}$	The set of operations for all jobs to be scheduled, where $O_{i,j}$ represents the j-th operation of manufacturing job J_i and NO_i represents the number of operations for the job J_i
$M = \{M_r \mid r = 1,2,\ldots,NR\}$	The set of candidate manufacturing resources, where M_r represents the r-th manufacturing resource
$TM_{i,j,r}^k(k = 1,2; i = 1,2,\ldots,NJ; j = 1,2,\ldots,NO_i; r = 1,2,\ldots,NR)$	The k-th type of time required for operation $O_{i,j}$ to be completed using manufacturing resource M_r, when $k = 1$ it represents preparation time, and when $k = 2$ it represents processing time
$TL_{r,s}(r,s = 1,2,\ldots,NR)$	The logistics time required for the manufacturing job to be transported from manufacturing resource M_r to manufacturing resource M_s
$ts_{i,j}(i = 1,2,\ldots,NJ;\ j = 1,2,\ldots,NO_i)$	Time variable, start time of operation $O_{i,j}$
$te_{i,j}(i = 1,2,\ldots,NJ;\ j = 1,2,\ldots,NO_i)$	Time variable, completion time of operation $O_{i,j}$
$x_{i,j,r}(i = 1,2,\ldots,NJ;\ j = 1,2,\ldots,NO_i;\ r = 1,2,\ldots,NR)$	Decision variable. It is 1 if operation $O_{i,j}$ is completed on manufacturing resource M_r and 0 otherwise
$y_{i,j,r}(i = 1,2,\ldots,NJ;\ j = 1,2,\ldots,NO_i;\ r = 1,2,\ldots,NR)$	Decision variable, the processing sequence of the operation $O_{i,j}$ on the manufacturing resource M_r

5.4. Working Hours Estimation Based on BPNN

There are many factors that affect working hours, and the relationships between these factors are complex. Working hours estimation is a typical highly non-linear problem. The back propagation neural network (BPNN) can adjust structural parameters based on sample characteristics, demonstrating strong adaptability. In conjunction with the resource allocation mechanism for D2TS, this section proposes a working hours estimation method based on BPNN (as shown in Figure 5). The specific steps are as follows:

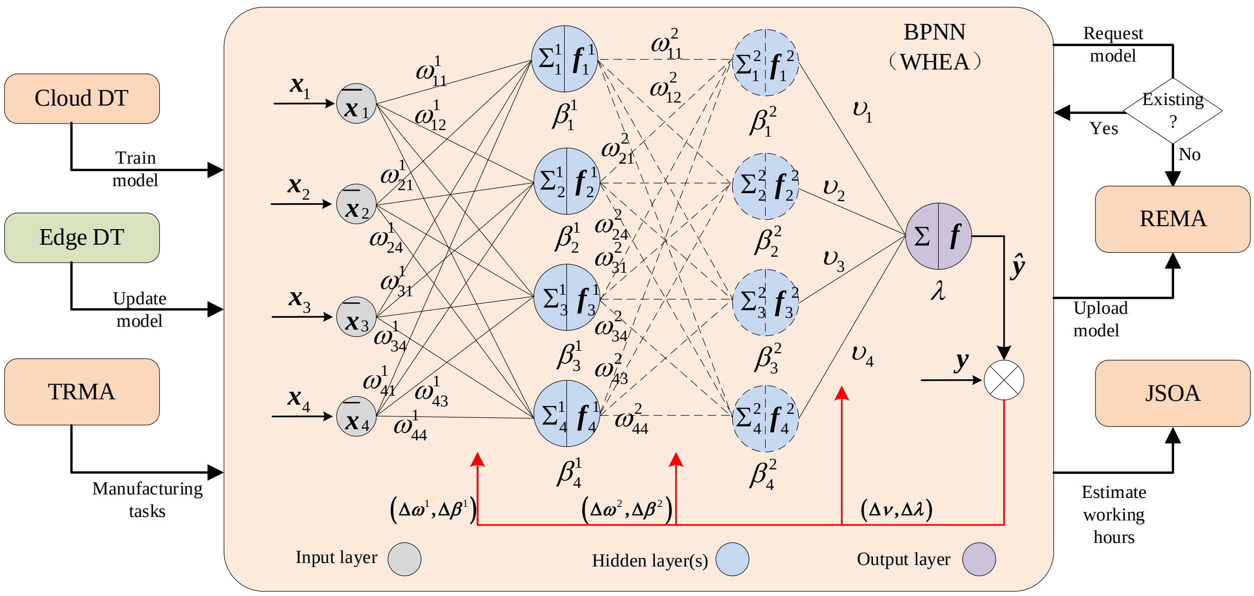

Figure 5. Working hours estimation method based on BPNN.

Step 1: First, determine the feature factors that affect working hours, then obtain the actual production data of historical tasks from the cloud digital twin as input samples for the BPNN. The samples are divided into training, validation, and testing samples at a certain proportion, and the working hours prediction model is trained in the cloud computing center. If there is no corresponding cloud digital twin for the resources, the sample data will be requested from the edge digital twin based on the "task–resource" pairs provided by the TRMA, and the model is trained in the fog node.

Step 2: Due to the existence of noise and singularity in the digital twin data, it is necessary to first clean and organize the data. Then, linear normalization is performed on the input and output variables, transforming them into values within the range of [0, 1] or [−1, 1] to meet the requirements of the network for input and output and to avoid saturation of neurons.

Step 3: Determine the topology of the BPNN, including the number of hidden layers and the number of nodes (neurons) in each hidden layer. This can be carried out using various methods, such as Kolmogorov's theorem or empirical formulas.

Step 4: Set the parameters of the BPNN, including initial weights, initial thresholds, learning rate, momentum factor, target error, and maximum number of training epochs, performance function, transferring function, training function, learning function.

Step 5: Input each training sample into the BPNN and modify the network's weights and thresholds through learning.

Step 6: Determine whether the performance function has reached the preset target error or the maximum number of training epochs has been reached. If not, return to Step 5 and enter the next round of training. If so, end the algorithm and output the prediction model.

Step 7: If the sample data are trained by the cloud digital twin, the trained model will be uploaded to the cloud knowledge base by the cloud digital twin. If the sample data are trained by the edge digital twin and authorized by the resource provider, the model will be uploaded to the cloud knowledge base through the REMA, and the provider will obtain the corresponding revenue.

Step 8: Based on the factors that affect working hours, the basic information of the candidate resource set and processing tasks will be input into the trained prediction model. After obtaining the output data of the neural network, inverse normalization will be performed to obtain the final working hours.

5.5. Job Scheduling Based on IGA

Most scheduling research focuses on the processing time of jobs, without taking into account the preparation time and transportation time required between distributed digital twin shop-floors to complete the jobs. Based on the previous research [37], this paper proposes a scheduling method using an improved genetic algorithm (IGA), as shown in Figure 6. The specific steps are as follows:

Step 1: Prepare the basic data of job scheduling. The complex machining jobs will be decomposed into basic operations by TAMA, that is, resource allocation routes. According to these basic operations, the candidate manufacturing resources capable of completing the machining operations will be searched and matched by the TRMA. The WHEA provides the working hours required to complete the operations, including processing time, preparation time, and transportation time.

Step 2: Prepare the mathematical model and the encoding method for the genetic algorithm. Based on the basic data provided in Step 1 and the method described in Section 5.3, a mathematical model for the optimization problem will be established, employing three scheduling optimization objectives f_1, f_2, f_3. This paper adopts an integer encoding method, dividing the chromosomes into two parts: the machine selection part and the process sorting part.

Step 3: Initialize the population. To ensure the quality of individuals in the initial population, three initialization methods are used to generate the initial population P(t), where P represents the population size and t represents the current generation, with t = 0 for the initial population.

Step 4: Evaluate the population. Combined with the basic data provided in Step 1, it is decoded using certain methods to calculate the objective functions of each individual in the population P(t). Then, the three objective functions are normalized, summed up, and converted into a fitness value F.

Step 5: Check if the algorithm has reached the stopping criterion. The termination criterion is reaching the maximum number of iterations in this paper.

Step 6: Selection operation. This paper adopts the tournament selection method, because compared with other methods (such as roulette wheel selection and random selection), this method has a better balance between convergence and computational efficiency.

Step 7: Crossover operation. The artificial pairing mechanism proposed in the previous research [37] is adopted, and a multi-point crossover operation is used for the machine selection part, while a preserving order-based crossover is used for the job sorting part.

Step 8: Mutation operation. Roulette wheel selection is used for the machine selection part, and adaptive neighborhood search is used for the job sorting part.

Step 9: Greedy operation. Then, multiple rules are used for repair, such as shortest processing time (SPT), earliest start (ES), earliest finish (EF), least utilized machine (LUM), minimum idle time (MIT), and minimum gap per job (MGJ), and finally the optimal individual is retained in the repaired population.

Step 10: Elite operation. Record and preserve individuals in the population that are not dominated, in order to avoid losing potentially better individuals during the previous operations.

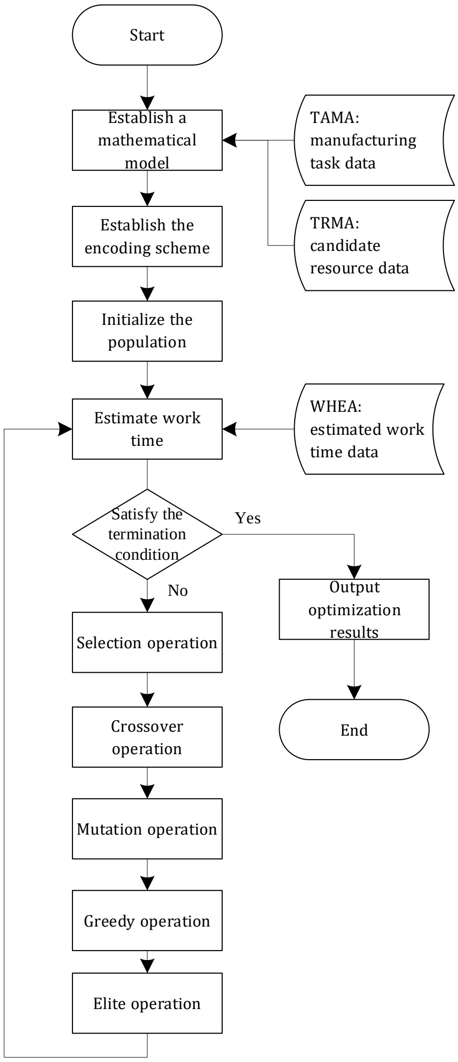

Figure 6. Job scheduling method based on IGA.

6. Cases Study

The casing is an important supporting and load-bearing component in aviation engines, which has typical features of complex shape, thin wall, difficult-to-process materials, high material removal rate, and high dimensional accuracy. The quality of its processing has an important impact on improving the overall performance of the engine. With the continuous improvement of the design structure and performance of aviation engines, there are more and more types of casings, and the processing difficulty is increasing. It requires a large number of ordinary machining equipment such as lathes, boring machines, and milling machines, as well as special processing equipment such as high-performance multi-axis CNC machining centers and high-pressure vacuum electron beam welding equipment. Against the background of DM, the production of aviation engine components adopts a distributed manufacturing model, fully utilizing the resource collaboration and processing of various small and medium-sized enterprise factories.

6.1. Case of Working Hours Estimation

This section takes the welding process of the inlet casing assembly of an aircraft engine as an example to verify the effectiveness and feasibility of the working hours estimation method described in Section 5.4. The inlet casing consists of several components, including the inner casing, support plate, outer casing, and support plate head, and is made of TC4 titanium alloy. The components are connected using high-energy electron beam welding, as shown in Figure 7. Due to the rarity of high-pressure vacuum electron beam welding equipment, consumers need to find welding equipment capable of completing this process on the cloud platform. Four factors related to working hours, including support plate thickness, welding rod diameter, weld thickness, and weld length, are selected as input features based on the processing characteristics of the component. By querying historical transaction data, 29 sets of raw data for this resource are available from the cloud digital twin, as shown in Table A1. That is, there are 29 samples in total. Columns 2 to 5 of the table represent the input factors of the training or testing samples, and the output factor is the total labor hours. Nineteen randomly selected sets are used as training data to train the BP neural network model, and the remaining ten sets are used as testing data to evaluate the effectiveness of the BP neural network model. The Matlab software (version: R2019b) is used to build the BPNN model for simulation experiments.

Figure 7. High-energy beam electron welding equipment and air inlet casing part.

The normalization formula is $\overline{x} = \frac{2(x - x_{min})}{x_{max} - x_{min}} - 1$, where x is the sample data of the factor influencing working hours, $\overline{x}$ is the normalized data, x_{max} and x_{min} are the maximum and minimum values in the sample data.

Due to Robert Hecht-Nielsen's proof that any continuous function on a closed interval can be approximated by a BP network with a single hidden layer, the number of hidden layers for this network model is determined to be 1. The number of hidden nodes is determined based on Kolmogorov's theorem and empirical formulas based on the least squares method:

Empirical formula based on the least squares method:

$$h = \left(0.43mn + 2.54m + 0.77n + 0.35 + 0.12n^2\right)^{1/2} + 0.5$$

Kolmogorov's theorem:

$$h = 2m + 1$$

In the above formulas, h represents the number of hidden nodes and m and n represent the number of input layer nodes and output layer nodes, respectively. In this case, $m = 4$ and $n = 1$, so according to the above formulas, the number of hidden nodes is 4 or 9. In this experiment, positive integers in the interval [3,10] are chosen as the number of hidden nodes.

In this experiment, the training function is "Trainlm", the learning function is "Learngdm", the transferring function between the input layer and the hidden layer is "Tansig", the transferring function between the hidden layer and the output layer is "Purelin", the maximum number of training epochs is 3000, the target error is 1.0×10^{-6}, and the learning rate is set to 0.1, 0.5, and 0.9. To eliminate the influence of randomness of the BPNN model, each parameter configuration was tested 10 times to obtain the average value, as shown in Table 3. Nine sets of test samples were input into the network for calculation, and the results are shown in Table 4. It can be seen that when the number of hidden nodes is 3, the average error between the calculated results and the actual processing time is within 10%, indicating good calculation performance. The optimal result was found when the number of hidden nodes is 3 and the learning rate is 0.1, with an average error of 3.57%. A line chart was generated to more intuitively show the comparison between the calculated values and the actual values, as shown in Figure 8, which verifies the effectiveness of the proposed method for estimating working hours.

Table 3. Experimental results for estimating working hours.

Index	Number of Hidden Nodes	Learning Rate	Number of Iterations for Convergence	MAE	MSE	RMSE
1	3	0.1	180	0.09	45.45	5.88
2	3	0.5	146	0.06	7.68	2.71
3	3	0.9	1292	0.12	99.44	7.02
4	4	0.1	1738	0.42	499.98	17.41
5	4	0.5	1912	1.03	23,953.11	75.98
6	4	0.9	735	0.19	151.29	10.41
7	5	0.1	403	0.32	504.25	17.34
8	5	0.5	365	0.23	213.76	12.45
9	5	0.9	491	0.33	896.47	19.32
10	6	0.1	259	0.26	391.11	17.15
11	6	0.5	176	0.28	215.13	13.62
12	6	0.9	309	0.18	66.84	7.37
13	7	0.1	223	0.22	344.33	15.55
14	7	0.5	216	0.22	289.06	13.08
15	7	0.9	200	0.26	207.60	13.75
16	8	0.1	139	0.31	282.64	14.93
17	8	0.5	140	0.25	274.59	14.51
18	8	0.9	181	0.19	164.34	11.66
19	9	0.1	153	0.24	253.58	13.88
20	9	0.5	115	0.29	299.27	15.67
21	9	0.9	122	0.17	132.30	11.13
22	10	0.1	128	0.21	212.96	12.42
23	10	0.5	121	0.24	178.11	11.84
24	10	0.9	135	0.24	111.48	9.78

Table 4. Average Optimal Result of Working Hours Estimation.

Index	1	2	3	4	5	6	7	8	9
True value	13.2	18.7	27.5	33	40.7	52.8	67.1	82.5	118.8
Estimated value	12.66	18.52	24.59	31.40	40.94	52.50	65.89	80.63	111.09
Error rate	4.06%	0.95%	10.57%	4.85%	0.60%	0.57%	1.81%	2.27%	6.49%

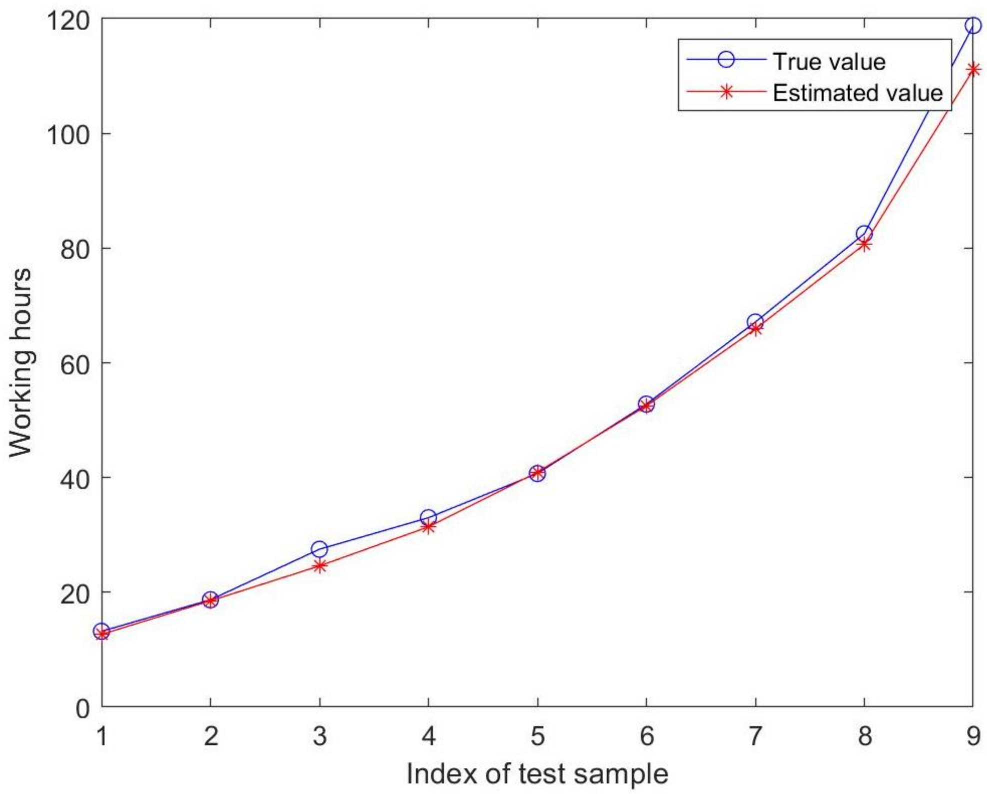

Figure 8. The curve graph of the average optimal results.

6.2. Case of Job Scheduling

In this section, the effectiveness and feasibility of the job scheduling method for D2TS described in Section 5.5 are verified using the machining tasks of multi-model aircraft engine casings as an example. The casing is the base of the entire engine and is distributed throughout various parts of the engine. It is equipped with a spindle, blades, and various connecting accessories. For example, in a turbofan engine, the casing can be divided into an intake casing, a fan casing, an intermediate casing, a compressor casing, a combustion chamber casing, a turbine casing, a bearing casing, a central transmission casing, and an accessory casing. Eight types of casing machining tasks are selected $\{J_1, J_2, \ldots, J_8\}$, and three to four key operations are taken as production operations in each machining task $\{(O_{11}, O_{12}, O_{13}), (O_{21}, O_{22}, O_{23}, O_{24}), \ldots\}$. The TRMA searches and matches candidate manufacturing resources for the above 27 operations, and a total of eight candidate manufacturing resources are obtained $\{M_1, M_2, \ldots, M_8\}$. The processing time, setup time, and transportation time required for scheduling optimization are estimated using the method described in Section 5.4, as shown in Tables A2 and A3.

The parameters required for the proposed IGA are shown in Table 5. The proposed IGA in Section 5.5 was implemented using Matlab software (Version: R2019b) and tested on a personal computer (CPU: 3.6GHz, RAM: 8 GB). To eliminate the randomness of the IGA, it was run ten times and the best result was selected.

Table 5. Parameters of IGA.

Parameters	Value
Number of individuals	100
Maximum number of iterations	100
Initial solution proportion for random generation method	0.4
Initial solution proportion for maximum priority selection method for remaining processing time	0.3
Initial solution proportion for uniform distribution method	0.3
Crossover probability	0.8
Minimum mutation probability	0.8
Maximum mutation probability	0.2

For the scheduling problem described in this paper, the proposed algorithm found 14 non-dominated solutions, as shown in Table 6. "Makespan" represents the production order completion cycle, "setup time" indicates the preparation time required when switching processing equipment, and "transportation time" denotes the logistics transportation time between geographically distributed devices. From Table 6, it can be observed that as transportation time increases, makespan gradually decreases, indicating that fully utilizing the distributed digital twin manufacturing approach proposed in this paper can shorten the production cycle and improve production efficiency. The variation in setup time is not significant because regardless of whether distributed manufacturing is adopted or not, this preparation time needs to be added whenever processing equipment is changed. A Gantt chart is drawn for Solution 3, which has the smallest makespan compared with all solutions and the least transportation time compared with Solutions 1 and 2, as shown in Figure 9. In Solution 14, although the transportation time is minimized, the makespan and setup time are maximized. This is because the production tasks are concentrated on a few pieces of processing equipment, leading to longer waiting times at individual bottleneck devices. Additionally, frequent switching of processing equipment significantly increases the preparation time before actual processing begins.

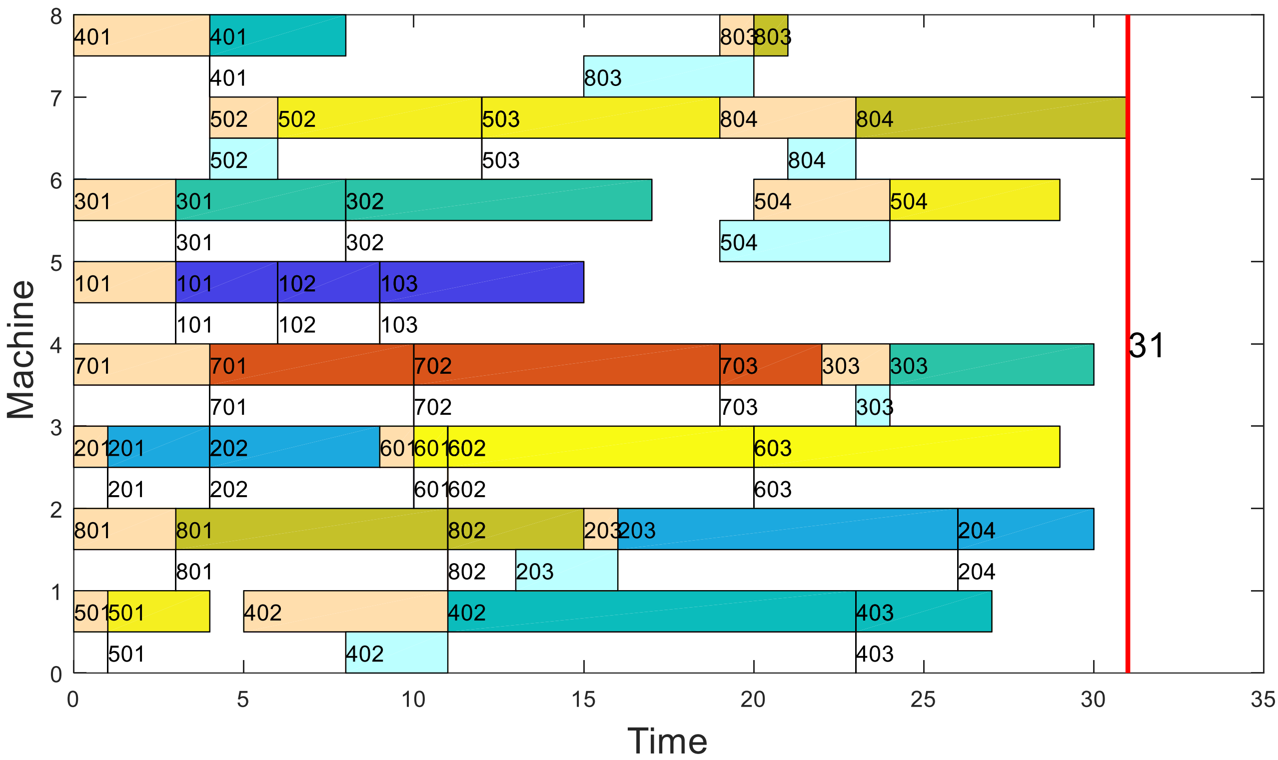

Figure 9. Gantt chart for Solution 3 with minimum transportation time.

Table 6. Non-dominated Solutions for the Instance.

	Makespan	Setup Time	Transportation Time
Solution 1	31	35	26
Solution 2	31	38	23
Solution 3	31	40	21
Solution 4	32	36	24
Solution 5	33	33	21
Solution 6	33	35	11
Solution 7	34	29	14
Solution 8	34	31	13
Solution 9	34	32	12
Solution 10	34	34	11
Solution 11	35	26	16
Solution 12	35	27	8
Solution 13	36	24	6
Solution 14	54	40	5

This paper compares the proposed IGA with three other GA-related improved algorithms, namely Elitism GA (ELGA) [38], Enhanced GA (ENGA) [39], and MOGWO [40]. Four metrics are used to evaluate the optimization capabilities of the four algorithms. “NNS” represents the number of non-dominant solutions obtained by each algorithm, “BMS” denotes the best makespan among all solutions, “BST” denotes the sum of the shortest setup time among all solutions, and “BTT” denotes the sum of the shortest transportation time among all solutions. As shown in Table 7, ELGA finds six optimal non-dominant solutions, ENGA finds five non-dominant solutions, MOGWO finds twenty-three optimal solutions, and IGA finds fourteen optimal solutions. It is evident that IGA outperforms ELGA and ENGA. While MOGWO finds the highest number of optimal solutions, IGA excels in finding the best setup time and transportation time among non-dominant solutions, such as the shortest BST and BTT. This indicates that the proposed IGA demonstrates a strong ability to escape local optima and provides more diverse distributed manufacturing solutions with superior optimization results.

Table 7. The result comparison.

	ELGA	ENGA	MOGWO	IGA
NNS	6	5	23	14
BMS	39	38	31	31
BST	36	33	28	24
BTT	16	7	7	5

7. Conclusions and Future Work

Motivated by distributed manufacturing and digital twin technologies, a distributed digital twin shop-floor is proposed to achieve dynamic sharing of large-scale manufacturing resources and capabilities. By employing the digital twin shop-floor model, the business can be contracted or expanded according to demand and the company can remain agile in the face of fluctuations. This paper attempts to use digital twin and the BPNN and an improved genetic algorithm to tackle the dynamic resource allocation problem in distributed manufacturing.

The major contributions of this paper are summarized as follows. Firstly, an information architecture of D2TS based on cloud–fog computing is proposed, which mainly includes three layers: the resource layer, the cloud–fog layer, and the application layer. Taking full advantage of the characteristics of easy interconnection and intercommunication of the digital twin shop-floor, geographically dispersed resources are dynamically connected through cloud computing and fog computing technologies to achieve resource sharing. Secondly, a resource allocation mechanism for D2TS is proposed. Based on multi-agent technology, it provides resource search, matching, scheduling, billing, credit, big data, and other functions. The innovation lies in linking the manufacturing platform with cloud digital twins or edge digital twins to save bandwidth resources and achieve reliable dynamic and real-time monitoring of critical resources, while ensuring the security of sensitive resource information. Finally, a combined approach using the BPNN algorithm and improved genetic algorithm is proposed to solve the resource scheduling problem in D2TS. The BPNN algorithm is used to estimate the job working hours, especially for new products or SMEs. The improved genetic algorithm is applied to optimize the scheduling of candidate resources, taking into account factors such as logistics transportation time required for distributed manufacturing. By combining these two algorithms, the utilization and productivity of resources can be effectively improved.

In future research, the optimization objective of scheduling needs to consider manufacturing costs, including machine processing costs, fixture design and manufacturing costs, logistics costs, labor costs, etc. By combining technologies such as digital twins, big data, and blockchain, the credibility of candidate resources in resource allocation and the prediction of equipment resource failures during task execution can be improved to enable preventive scheduling optimization and improve the robustness of scheduling solutions. Combining the multi-attribute decision-making theory, this paper uses the three-way decision method and the improved TOPSIS method will be employed to study the decision-making problem of multiple non-dominated scheduling schemes more suitable for the D2TS environment, in order to improve decision-making efficiency and reduce decision-making costs.

Author Contributions: Conceptualization, H.Z.; Methodology, H.Z. and Q.Y.; Formal Analysis, H.Z. and Q.Y.; Writing—Original Draft Preparation, Y.Q. and S.C.; Writing—Review and Editing, H.Z., Q.Y. and G.Z.; Funding Acquisition, H.Z. and G.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by the National Natural Science Foundation of China (Grant No. 51705472); in part by the Humanities and Social Sciences Project of the Ministry of Education of China (Grant No. 22YJAZH144); in part by the Scientific and Technological Research Projects in Henan Province (Grant No. 232102320063); in part by the Program for Innovative Research Teams (in Science and Technology) at the University of Henan Province (Grant No. 21IRTSTHN018); and in part by the Scientific Research Team Plan of Zhengzhou University of Aeronautics (Grant No. 23ZHTD01014).

Data Availability Statement: Not applicable.

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

Nomenclature

DT	Digital twin
DTS	Digital twin shop-floor
D2TS	Distributed digital twin shop-floor
D2TSRA	Distributed digital twin shop-floor resource allocation
BP	Back propagation
DM	Distributed manufacturing
ICTs	Information and communication technologies
SMEs	Small and medium-sized enterprises
SANs	Sensor and/or actuator nodes
DTMC	Digital twin manufacturing cell
CPS	Cyber-physical system
CNC	Computer numerical control
BN	Broker node
CN	Computing node
RN	Repository node
IMP	Intelligent manufacturing platform

CCS	Cloud computing space
FCS	Fog computing space
PRS	Physical resource space
CDTFA	Cloud digital twin function area
CDMFA	Cloud data management function area
ROAFA	Resource optimization allocation function area
REMA	Resource management agent
TAMA	Task management agent
TRMA	Trading management agent
WHEA	Working hours estimation agent
JSOA	Job scheduling optimization agent
CAPP	Computer-aided process planning
BPNN	Back propagation neural network
IGA	Improved genetic algorithm
MAE	Mean absolute error
MSE	Mean squared error
RMSE	Root mean square error

Appendix A

Table A1. Experimental sample data.

Index	Plate Thickness/mm	Welding Rod Diameter/mm	Weld Bead Thickness/mm	Weld Bead Length/m	Working Hours/min
1	2	2.5	2	0.5	5
2	2.5	2.5	2.5	0.9	8.1
3	3	2.5	3	1.4	11.2
4	24	5	24	1.1	118.8
5	5	3.2	5	2	20
6	6	4	6	1	12
7	8	4	8	0.7	11.9
8	10	4	10	1.6	40
9	12	5	12	1	30
10	14	5	14	1.3	48.1
11	16	5	16	0.5	24
12	16	5	16	1.9	91.2
13	18	5	18	1	61
14	20	5	20	0.7	52.5
15	2	2.5	2	1.1	11
16	2.5	2.5	2.5	1.1	8.8
17	3	2.5	3	1.1	9.6
18	4	3.2	4	1.1	9.9
19	5	3.2	5	1.1	11
20	5	4	6	1.1	13.2
21	8	4	8	1.1	18.7
22	10	4	10	1.1	27.5
23	12	5	12	1.1	33
24	14	5	14	1.1	40.7
25	16	5	16	1.1	52.8
26	18	5	18	1.1	67.1
27	20	5	20	1.1	82.5
28	4	3.2	4	0.7	6.3
29	28	5.8	28	1.1	136.4

Table A2. Processing time and setup time for the instance.

Job	Oper.	Processing Time/Setup Time							
		M1	M2	M3	M4	M5	M6	M7	M8
J1	O1,1	5/1	3/1	5/1	3/3	3/3	–	10/1	9/7
	O1,2	10/4	–	5/2	8/6	3/3	9/3	9/3	6/1
	O1,3	–	10/2	–	5/2	6/3	2/1	4/1	5/4
J2	O2,1	5/4	7/2	3/1	9/5	8/1	–	9/5	–
	O2,2	–	8/2	5/5	2/1	6/3	7/7	10/8	9/4
	O2,3	–	10/1	–	5/2	6/3	4/3	1/1	7/2
	O2,4	10/1	8/3	9/3	6/3	4/3	7/5	–	–
J3	O3,1	10/3	–	–	7/1	6/5	5/3	2/2	4/2
	O3,2	–	10/1	6/2	4/4	8/6	9/3	10/8	–
	O3,3	1/1	4/3	5/4	6/2	–	10/7	–	7/3
J4	O4,1	3/2	1/1	6/1	5/5	9/7	7/6	8/4	4/4
	O4,2	12/6	11/7	7/1	8/3	10/6	5/5	6/5	9/3
	O4,3	4/2	6/3	2/1	10/9	3/2	9/3	5/4	7/6
J5	O5,1	3/1	6/5	7/6	8/6	9/8	–	10/6	–
	O5,2	10/7	–	7/1	4/2	9/1	8/5	6/2	–
	O5,3	–	9/6	8/5	7/2	4/3	2/2	7/6	–
	O5,4	11/1	9/4	–	6/6	7/2	5/4	3/2	6/6
J6	O6,1	6/2	7/6	1/1	4/3	6/1	9/2	–	10/3
	O6,2	11/4	–	9/8	9/6	9/6	7/1	6/6	4/3
	O6,3	10/5	5/5	9/2	10/2	11/8	–	10/4	–
J7	O7,1	5/3	4/4	2/1	6/4	7/7	–	10/9	–
	O7,2	–	9/8	–	9/1	11/4	9/1	10/4	5/5
	O7,3	–	8/1	9/6	3/1	8/4	6/1	–	10/8
J8	O8,1	2/1	8/3	5/5	9/3	–	4/2	–	10/8
	O8,2	7/1	4/4	7/4	8/1	9/7	–	10/10	–
	O8,3	9/2	9/8	–	8/2	5/4	6/3	7/2	1/1
	O8,4	9/7	–	3/2	7/6	1/1	5/4	8/4	–

Note: "–" means the device does not support the corresponding operation for machining.

Table A3. Transportation time for the instance.

Candidate Manuf. Resources	Transportation Time							
	M1	M2	M3	M4	M5	M6	M7	M8
M1	0	4	2	1	5	1	2	1
M2	1	0	5	1	5	2	5	5
M3	1	3	0	2	3	4	5	5
M4	3	1	3	0	1	4	3	3
M5	4	2	5	3	0	1	4	1
M6	5	5	1	1	2	0	3	2
M7	1	1	2	5	2	5	0	3
M8	3	5	1	5	5	4	2	0

References

1. Matt, D.T.; Rauch, E.; Dallasega, P. Trends towards distributed manufacturing systems modern forms for their design. *Procedia CIRP* **2015**, *33*, 185–190. [CrossRef]
2. Rauch, E.; Dallasega, P. Sustainability in manufacturing and supply chains through distributed manufacturing systems and networks. In *Encyclopedia of Sustainable Technologies*; Elsevier: Oxford, UK, 2017.
3. Atmojo, U.D.; Salcic, Z.; Wang, K.I.-K.; Vyatkin, V. A Service-Oriented Programming Approach for Dynamic Distributed Manufacturing Systems. *IEEE Trans. Industr. Inform.* **2020**, *16*, 151–160. [CrossRef]
4. Shafto, M.; Conroy, M.; Doyle, R.; Glaessgen, E.; Wang, L. *Modeling, Simulation, Information Technology and Processing Roadmap*; National Aeronautics and Space Administration: Washington, DC, USA, 2012. Available online: http://www.nasa.gov (accessed on 22 June 2023).
5. Zhang, W.; Zhang, G.; Guo, X.; Zhang, Y.F. Digital twin-driven service model and optimal allocation of manufacturing resources in shared manufacturing. *J. Manuf. Syst.* **2021**, *59*, 165–179. [CrossRef]

6. Grieves, M. Origins of the Digital Twin Concept. 2016. Available online: https://www.researchgate.net/publication/307509727_Origins_of_the_Digital_Twin_Concept (accessed on 20 May 2023).
7. Tuegel, E.J.; Ingraffea, A.R.; Eason, T.G.; Spottswood, S.M. Reengineering aircraft structural life prediction using a digital twin. *Int. J. Aerosp. Eng.* **2011**, *7*, 154798. [CrossRef]
8. Tao, F.; Zhang, M. Digital Twin Shop-Floor. A New Shop-Floor Paradigm Towards Smart Manufacturing. *IEEE Access* **2017**, *5*, 20418–20427. [CrossRef]
9. Sanchez, L.M.; Nagi, R. A review of agile manufacturing systems. International. *Int. J. Prod. Res.* **2001**, *39*, 3561–3600. [CrossRef]
10. Rusinko, C. Green manufacturing: An evaluation of environmentally sustainable manufacturing practices and their impact on competitive outcomes. *IEEE Trans. Eng. Manag.* **2007**, *54*, 445–454. [CrossRef]
11. Ren, L.; Zhang, L.; Wang, L.H.; Tao, F.; Chai, X.D. Cloud manufacturing: Key characteristics and applications. *Int. J. Ccomput. Integr. Mang.* **2017**, *30*, 501–515. [CrossRef]
12. Li, B.H.; Zhang, L.; Ren, L.; Chai, X.D.; Tao, F.; Wang, Y.Z.; Yin, C.; Huang, P.; Zhao, X.P.; Zhou, Z.D. Typical features, key technologies, and applications of cloud manufacturing. *Comput. Integr. Manuf. Syst.* **2012**, *18*, 1345–1356.
13. Cheng, Y.; Bi, L.N.; Tao, F.; Ji, P. Hypernetwork-based manufacturing service scheduling for distributed and collaborative manufacturing operations towards smart manufacturing. *J. Intell. Manuf.* **2020**, *2020*, 1707–1720. [CrossRef]
14. Moench, L.; Shen, L.J. Parallel machine scheduling with the total weighted delivery time performance measure in distributed manufacturing. *Comput. Oper. Res.* **2021**, *127*, 105126. [CrossRef]
15. Carlucci, D.; Renna, P.; Materia, S.; Schiuma, G. Intelligent decision-making model based on minority game for resource allocation in cloud manufacturing. *Manag. Decis.* **2019**, *58*, 2305–2325. [CrossRef]
16. Delaram, J.; Houshamand, M.; Ashtiani, F.; Valilai, O.F. A utility-based matching mechanism for stable and optimal resource allocation in cloud manufacturing platforms using deferred acceptance algorithm. *J. Manuf. Syst.* **2021**, *60*, 569–584. [CrossRef]
17. Yin, C.; Xiao, X.; Qiu, L. Constraint characteristics of networked collaborative machining equipment and an optimization selection method. *J. Mech. Eng.* **2018**, *54*, 192–201. [CrossRef]
18. Wu, X.; Sun, L. Real time scheduling of workshop based on data-driven intelligent manufacturing system. *Cont. Des.* **2020**, *35*, 523–535. [CrossRef]
19. Zhou, B.; Bao, J.S.; Li, J.; Lu, Y.Q.; Liu, T.Y.; Zhang, Q. A novel knowledge graph-based optimization approach for resource allocation in discrete manufacturing workshops. *Robot. Comput. Integr. Manuf.* **2021**, *71*, 102160. [CrossRef]
20. Adam, S.; Peter, E.; Botond, K. Trust-based resource sharing mechanism in distributed manufacturing. *Int. J. Comput. Integr. Manuf.* **2020**, *33*, 1–21. [CrossRef]
21. Liu, Y.H.; Yu, W.J.; Dillon, T.; Rahayu, W.; Li, M. Empowering IOT predictive maintenance solutions with AI: A Distributed System for Manufacturing Plant-Wide Monitoring. *IEEE Trans. Industr. Inform.* **2022**, *18*, 1345–1354. [CrossRef]
22. Zhang, H.J.; Zhang, G.H.; Yan, Q. Digital twin-driven cyber-physical production system towards smart shop-floor. *J. Ambient Intell. Hum. Comput.* **2019**, *10*, 4439–4453. [CrossRef]
23. Wang, H.; Lv, L.; Li, X.; Li, H.; Leng, J.; Zhang, Y.; Thomson, V.; Liu, G.; Wen, X.; Sun, C. A safety management approach for Industry 5.0′s human-centered manufacturing based on digital twin. *J. Manuf. Syst.* **2023**, *66*, 1–12. [CrossRef]
24. Zhang, L.T.; Guo, Y.; Qian, W.W.; Wang, W.L.; Liu, D.Y.; Liu, S. Modelling and online training method for digital twin workshop. *Int. J. Prod. Res.* **2021**, *68*, 3943–3962. [CrossRef]
25. Zhang, C.; Zhou, H.; Xiao, J. Multidimensional and multi-scale modeling of digital twin manufacturing cells and edge cloud collaborative configuration. *Comput. Integr. Manuf. Syst.* **2023**, *29*, 355–371. [CrossRef]
26. Qian, W.W.; Guo, Y.; Cui, K.; Wu, P.X.; Fang, W.G.; Liu, D.Y. Multidimensional data modeling and model validation for digital twin workshop. *J. Comput. Inf. Sci. Eng.* **2021**, *21*, 031005. [CrossRef]
27. Liu, J.; Zhuang, C.B.; Liu, J.H.; Miao, T.; Wang, J.Q. Online prediction technology of workshop operating status based on digital twin. *Comput. Integr. Manuf. Syst.* **2021**, *27*, 467–477. [CrossRef]
28. Hong, Q.; Sun, Y.F.; Liu, T.Y.; Fu, L.; Xie, Y.F. TAD-Net: An approach for real-time action detection based on temporal convolution network and graph convolution network in digital twin shop-floor. *Digit. Twin* **2023**, *1*, 39–56. [CrossRef]
29. Liu, S.M.; Lu, Y.Q.; Li, J.; Shen, X.W.; Sun, X.M.; Bao, J.S. A blockchain-based interactive approach between digital twin-based manufacturing systems. *Comput. Ind. Eng.* **2023**, *175*, 108827. [CrossRef]
30. Wang, J.; Liu, Y.; Ren, S.; Wang, C.; Ma, S.Y. Edge computing-based real-time scheduling for digital twin flexible job shop with variable time window. *Robot Cim-Int. Manuf.* **2023**, *79*, 108827. [CrossRef]
31. Wang, X.D.; Hu, X.F.; Ren, Z.J.; Tian, T.C.; Wan, J.F. Knowledge-graph based multi-domain model integration method for digital-twin workshops. *Intern. J. Adv. Manuf. Technol.* **2023**, *128*, 405–421. [CrossRef]
32. Bellavista, P.; Giannelli, C.; Mamei, M.; Mendula, M.; Picone, M. Application-driven network-aware digital twin management in industrial edge environments. *IEEE Trans. Industr. Inform.* **2021**, *17*, 7791–7801. [CrossRef]
33. Vyskocil, J.; Douda, P.; Novak, P.; Wally, B. A digital twin-based distributed manufacturing execution system for industry 4.0 with ai-powered on-the-fly replanning capabilities. *Sustainability* **2023**, *15*, 6251. [CrossRef]
34. Liu, C.; Le Roux, L.; Korner, C.; Tabaste, O.; Lacan, F.; Bigot, S. Digital twin-enabled collaborative data management for metal additive manufacturing systems. *J. Manuf. Syst.* **2023**, *62*, 857–874. [CrossRef]
35. Timothy, S.; Michael, S.; William, Z.B.; Michael, P.B.; Moneer, H.; Thomas, H. Integrated operations management for distributed manufacturing. *IFAC-PapersOnLine* **2020**, *52*, 1820–1824. [CrossRef]

36. Zhang, H.J.; Yan, Q.; Zhang, G.H.; Li, Q.Y.; Yu, J. Dynamic optimization decision of manufacturing resources based on digital twins. *Comput. Integr. Manuf. Syst.* **2021**, *27*, 521–535. [CrossRef]
37. Zhang, G.H.; Hu, Y.F.; Sun, J.H.; Zhang, W.Q. An improved genetic algorithm for the flexible job shop scheduling problem with multiple time constraints. *Swarm Evol. Comput.* **2020**, *54*, 100664. [CrossRef]
38. Xie, N.M.; Chen, N.L. Flexible job shop scheduling problem with interval grey processing time. *Appl. Soft Comput.* **2018**, *70*, 513–524. [CrossRef]
39. Dai, M.; Tang, D.B.; Giret, A.; Salido, M.A. Multi-objective optimization for energy-efficient flexible job shop scheduling problem with transportation constraints. Robot. *Comput. Integr. Manuf.* **2019**, *59*, 143–157. [CrossRef]
40. Luo, S.; Zhang, L.X.; Fan, Y.S. Energy-efficient scheduling for multi-objective flexible job shops with variable processing speeds by grey wolf optimization. *J. Clean. Prod.* **2019**, *234*, 1365–1384. [CrossRef]