

## Article

# Leveraging Blockchain to Support Collaborative Distributed Manufacturing Scheduling

Veera Babu Ramakurthi <sup>1</sup>, Vijaya Kumar Manupati <sup>2</sup>, Leonilde Varela <sup>3,\*</sup>  and Goran Putnik <sup>3</sup><sup>1</sup> Department of Mechanical Engineering, National Institute of Technology, Warangal 506004, India<sup>2</sup> Operations and Supply Chain Division, NITIE, Mumbai 400087, India<sup>3</sup> ALGORITMI Research Centre/LASI, Department of Production and Systems, School of Engineering, University of Minho, 4804-533 Guimarães, Portugal

\* Correspondence: leonilde@dps.uminho.pt

**Abstract:** The recent trend in collaborative distributed manufacturing scheduling (CDMS) has gained significant importance in extended, networked, and virtual manufacturing environments due to its adaptability and integration potential. In a distributed manufacturing environment, CDMS can occur within a single factory or across multiple companies in a dynamic and variable extended or virtual organization. For effective collaboration, the CDMS system must be secure, transparent, and trustworthy. This paper proposes a blockchain-based model for CDMS and discusses its implementation in the processing of manufacturing functions, specifically joint process planning and scheduling. An illustrative example is used to demonstrate the application of the proposed approach and its potential to enhance the management processes of CDMS enterprises. The results of the analysis indicate that the proposed blockchain approach can effectively facilitate communication and integration among CDMS enterprises. Additionally, the approach can be expanded to more complex manufacturing environments under different conditions.

**Keywords:** blockchain; collaborative distributed manufacturing scheduling; smart contract; Ethereum



**Citation:** Ramakurthi, V.B.; Manupati, V.K.; Varela, L.; Putnik, G. Leveraging Blockchain to Support Collaborative Distributed Manufacturing Scheduling. *Sustainability* **2023**, *15*, 3283. <https://doi.org/10.3390/su15043283>

Academic Editor: Mirco Peron

Received: 7 December 2022

Revised: 4 February 2023

Accepted: 6 February 2023

Published: 10 February 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Collaborative distributed manufacturing scheduling (CDMS) involves multiple entities working together and sharing resources to achieve their individual and collective goals [1,2]. This type of scheduling can occur in various manufacturing settings, including extended manufacturing environments (EME) and virtual enterprises (VE) [3,4]. In the current complex and rapidly changing manufacturing environments [5], organizations such as EME or VE require collaborative distributed manufacturing scheduling (CDMS) to meet the demands of Industry 4.0 [6]. CDMS involves two or more entities collaborating and sharing resources and processes to reach their individual and joint goals [1,2]. Scheduling problems that occur in distributed environments [3,7] are complex, but CDMS can help companies tackle them effectively.

The scheduling problems in CDMS [4,8] are becoming increasingly complex due to the growing number of entities and resources involved. These entities and resources may be geographically dispersed and often involve combinatorial optimization problems [9]. The scheduling process in CDMS is not only complex in and of itself but also becomes more challenging to handle as the number of participating entities increases.

Blockchain-based collaborative distributed manufacturing scheduling (BCDMS) offers a solution to the complex scheduling problems faced in modern I4.0-era manufacturing environments. The proposed model in this paper aims to support BCDMS by providing a novel approach based on Jackson's model for classical job shop environments. This approach can be applied to more complex manufacturing scenarios, involving two or more work centers or machines, in a classical job shop or in extended and virtual enterprises through a distributed manufacturing environment. The model extends Jackson's model to

facilitate collaboration between multiple factories and ensure efficient execution of jobs in a distributed manufacturing environment [10,11].

The proposed BCDMS model offers a comprehensive solution for a range of manufacturing environments, including traditional workshops and distributed manufacturing setups. Unlike existing solutions, which are specific to a particular production environment or programming method, the BCDMS model makes a novel contribution to the field by integrating various techniques such as mathematical optimization, metaheuristics, machine learning, and agent-based approaches to solve basic scheduling problems. The BCDMS model's flexibility enables it to adapt to the specific needs of different manufacturing environments [12–18].

This paper makes a contribution to the field of collaborative management in Industry 4.0 and distributed manufacturing scheduling. The paper is organized into six sections to properly present the contribution. Section 2 provides a brief literature review on the topic, including the use of blockchain technology and distributed manufacturing scheduling approaches. Section 3 summarizes the proposed blockchain and smart contract technology for distributed manufacturing scheduling. The use of the proposed scheduling approach is illustrated in Section 4 through an application example. The main results, including the benefits and limitations of the proposed scheduling approach, are presented in Section 5. Finally, Section 6 concludes the paper and proposes future work.

## 2. Literature Review

The advancement of Industry 4.0 has brought significant changes to the manufacturing industry, particularly in terms of collaboration and scheduling processes. Collaborative management and distributed manufacturing scheduling have become essential in ensuring efficient production in today's I4.0 era. In this literature review, we will examine the current state of research on collaborative management in Industry 4.0 and the use of blockchain technology and distributed manufacturing scheduling approaches. By outlining the current state of the field, this review will provide a foundation for the proposed solution presented in this paper.

### 2.1. Collaborative Management and Industry 4.0 Technology

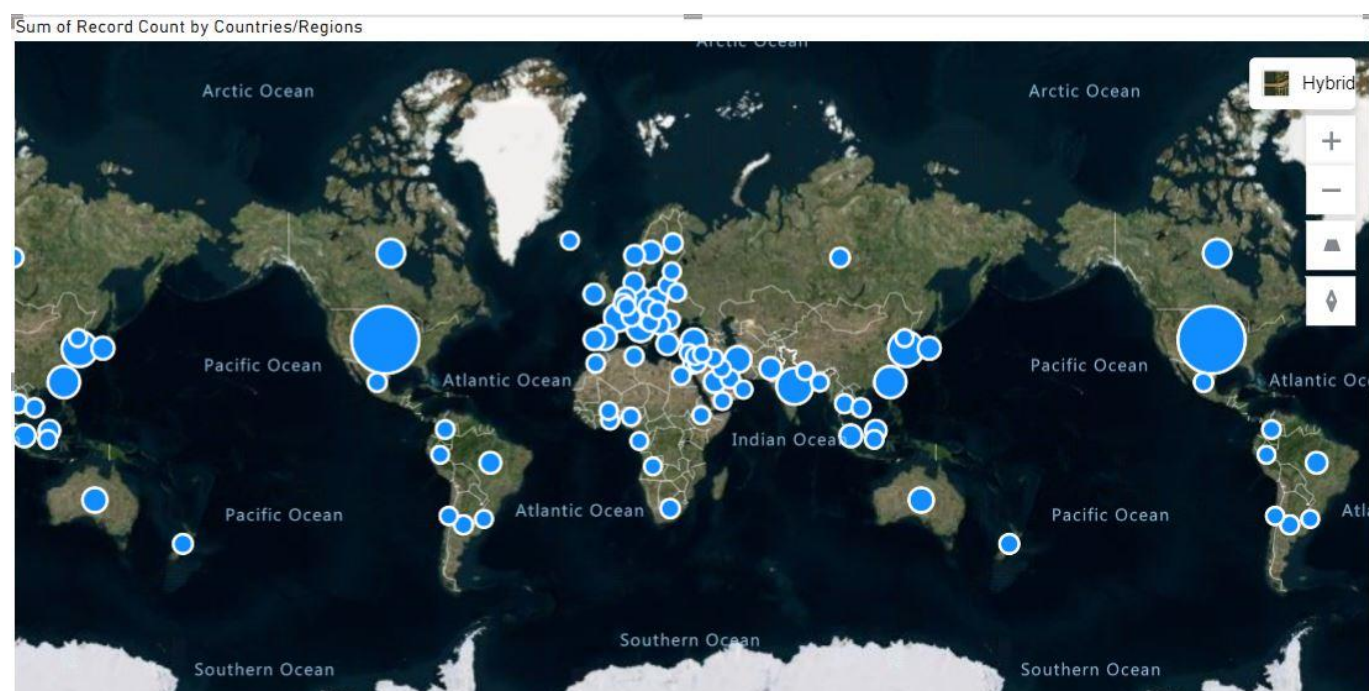
The role of collaborative management in Industry 4.0 has become increasingly important in recent years [19]. Blockchain technology, with its decentralized and distributed digital registry system, has seen widespread use in various industries over the past 15 years, including production planning, where it enhances efficiency and transparency through integration with management systems [20–24]. Table 1 summarizes information on articles published in various journals that focus on the use of blockchain technology in the context of manufacturing and process planning and scheduling.

This section of the literature review focuses on the study of process planning and scheduling in manufacturing and the application of blockchain technology in this context. Table 1 presents data on articles published in various journals on these two topics over the last 15 years. The table showcases the volume of work carried out in the field of process planning and scheduling in manufacturing and the use of blockchain technology in this area.

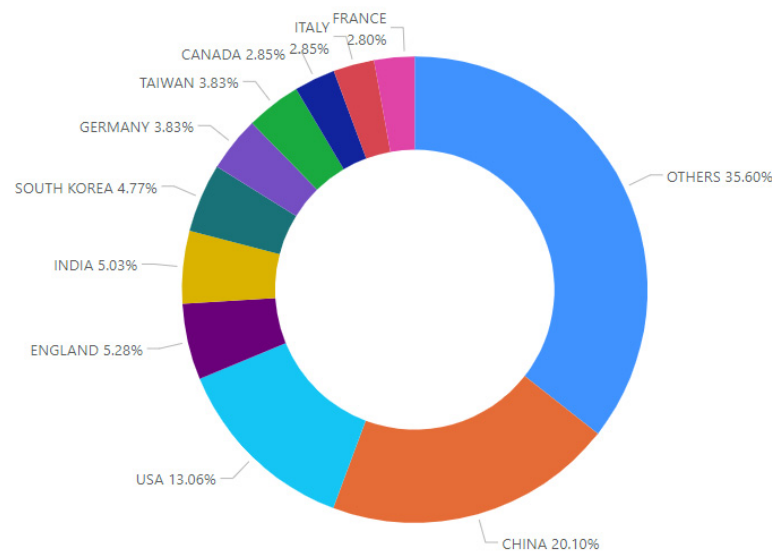
Figure 1 illustrates the significant research and development efforts that have been dedicated to process planning and scheduling in the context of distributed manufacturing systems. The use of blockchain-based methods has also been explored in this context. The figure displays the volume of work conducted in the last 15 years across different countries using the keywords: blockchain technology, process planning and scheduling, and manufacturing management. Additionally, Figure 2 provides a more detailed view of the contributions made by different countries in this field.

**Table 1.** Blockchain technology applied in manufacturing scheduling.

Source Title	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022	Total
Computers & Industrial Engineering	1	3	5	2	2	1	3	2	3	8	3	3	5	8	12	5	66
IEEE ACCESS												1	3	12	10	5	31
International Journal of Advanced Manufacturing Technology	3	1	2		2	6	5	7	5	3	4	3	4	1	1	7	54
International Journal of Computer Integrated Manufacturing	3	1	2	3	1	2	2	1	4	2		6	3	2	1		33
International Journal of Production Economics		3	1	1	3		2	2	3	2	2			1	4	3	27
International Journal of Production Research	1	7	11	1	9	14	7	6	3	5	6	5	7	11	5	10	108
Journal of Intelligent Manufacturing	1		2	1		1		2	2			4	2	1	2	2	20
Journal of Manufacturing Systems	1								3	3	2	3		4	8	8	32
Robotics And Computer-Integrated Manufacturing	1	1	2	4			1					2	1	3	4	4	24
Sustainability											2	2	1	5	7	10	27
<b>Total</b>	<b>11</b>	<b>16</b>	<b>25</b>	<b>12</b>	<b>17</b>	<b>24</b>	<b>20</b>	<b>20</b>	<b>23</b>	<b>23</b>	<b>19</b>	<b>29</b>	<b>26</b>	<b>48</b>	<b>54</b>	<b>54</b>	<b>422</b>



**Figure 1.** Geographical view of work contributions in last 15 years for the keywords: blockchain technology, process planning scheduling, and manufacturing management.



**Figure 2.** Country wise (Top 10 Countries) contribution of articles that cover the keywords blockchain technology in manufacturing and process planning scheduling in last 15 years.

### 2.2. Manufacturing Scheduling Environments and Approaches

Manufacturing scheduling involves planning tasks and activities using a group of production centers or machines, which can be located within a single company or across multiple manufacturing companies and organizations [25–27]. To address scheduling challenges, a variety of approaches and methods can be utilized, including heuristics and meta-heuristics, as well as other scheduling techniques [4,16–18,25–31].

The scheduling process can occur in different kinds of manufacturing environments, namely extended, distributed or VE [3,4,8,32], turning these problems into complex ones, under varying types of conditions that are dynamically changing [33,34].

### 2.3. Distributed Manufacturing Scheduling

In collaborative distributed production scheduling (DMS), the coordination of production tasks is managed through a task chain process, resulting in a complex production schedule. A program is characterized by distributing and sequencing tasks on appropriate production resources to satisfy a set of constraints imposed on production resources and tasks. These constraints are critical to achieving predetermined production goals.

Distributed manufacturing systems (DMS) play a vital role in the current era of globalization since they can be used to manage and control distributed systems in organizations or networks. A popular approach to CDMS is the use of multi-agent systems (MAS). A MAS consists of a set of autonomous agents that can work together toward a common goal. To facilitate this, the MAS approach to CDMS uses specific architectures and protocols. These frameworks and protocols provide efficient communication and coordination between agents, enabling efficient management and control of the distributed system [29].

CDMS is the most important part of the modern global manufacturing environment as it provides coordination and management of distributed production processes. In [9], an approach to dynamic planning based on a dynamic decision-making model with multiple criteria is proposed. This approach aims to integrate strategies that allow for finding a compromise between different performance indicators such as cost, quality, and delivery time. Furthermore, various approaches, algorithms, tools, systems, and platforms support production planning from centralized to decentralized architectures. These approaches aim to integrate production planning as well as other management functions such as process planning, nesting, system balancing, and layout determination. They can be found in many sources including [4,7,8,16–18].

The main purpose of this study is to analyze the potential benefits that can be derived from the application of an extended generic programming approach. The approach adapts



to different production environments, from classical workshops to the implementation of manufacturing enterprises or virtual enterprises. The general approach to advanced scaling is versatile, allowing for a variety of solution approaches, methods, and combinations of algorithms. This includes mathematical or optimization methods as well as other planning methods and systems, to provide a comprehensive solution for planning in production environments.

To that end, next, some relevant and closely related scheduling approaches and systems from the literature are briefly mentioned in order to better contextualize the work carried out and its novel contribution.

A simulation model including a dynamic scaling scheme is proposed to generate training examples that the authors consider good scaling. The training process uses a genetic algorithm combined with a tolerance-based learning algorithm to derive general planning rules from training examples [35]. This allows the model to adapt to new observed cases and modify its knowledge accordingly. We propose that the experimental results show that the dynamic scheduling scheme with a scheduling rule is superior to the static scheduling scheme for distributed scheduling purposes.

An agent-based approach is proposed for distributed production programs. This approach aims to achieve global combinatorial scheduling optimization by integrating workflow scheduling in a distributed production environment [36]. The proposed agent-based approach is adapted from the particle swarm optimization (PSO) algorithm, in which agents move toward a graph to find an optimal global time interval.

In the study conducted by Saeidlou et al., in 2019 [37], a novel approach for distributed manufacturing scheduling was presented. The system proposed is based on a cooperative framework that utilizes a set of rules that were identified as being most relevant for the scheduling task at hand. The decision-making process is implemented through an agent-based decision support system, which allows for the integration of the selected rules. By using this approach, the authors claim that the system is able to find near-optimal solutions for the scheduling problem within a computationally feasible amount of time. The cooperative system proposed in this study aims to improve the efficiency of CDMS by integrating the most relevant rules in a coordinated and efficient manner.

Zhang et al. [38] proposed a new task scheduling system in a production environment with multiple factories and workflows. A system consists of a set of rules that are considered necessary to meet the constraints of the production environment. To facilitate the planning process, the authors used an agent-based decision support system, a special type of computer program designed to make optimal decisions in a timely manner. The proposed system aims to improve the efficiency of work planning in a production environment using a set of important rules and a special decision support system.

Wang, Ghenniwa, and Shen, in 2008 [39], proposed a different task scheduling method that uses a special type of computer algorithm called the discrete fruit fly optimization algorithm, designed to reduce costs and power consumption. They proposed that their method outperformed some others. In 2008, a group of researchers proposed a real-time method for scheduling work in a factory. They used a type of computer program called agent-based service-oriented architecture, which divided the manufacturing floor into small sections called work cells. They use their own dynamic scheduling system, which offered a way for multiple computer programs and web services to work together to integrate these programs.

Manupati et al., in 2017 [40], used a telefacturing-based distributed manufacturing environment as a means of optimizing manufacturing services by enhancing the interoperability between various hub facilities. The objective of this approach is to establish improved communication and collaboration between the hubs in order to achieve a more efficient and effective manufacturing process. The implementation of this telefacturing approach is expected to lead to enhanced production outcomes and a more streamlined manufacturing service.

Özgülven, Özbakır, and Yavuz, in 2010 [41] did propose a mathematical model for job-shop scheduling problems with routing and process plan flexibility, which enables to integrate two fundamental management functions related to the tasks scheduling and process planning, with a certain degree of flexibility in this joint management process.

In a recent study, [42], a novel approach for task scheduling in a multi-factory manufacturing setting with workflow constraints was proposed. The authors utilized an integrated brainstorm optimization algorithm to balance multiple objectives simultaneously. The results of the study showed that this approach outperformed other tested methods and proved to be an effective solution for task scheduling in complex manufacturing environments.

Mao, Li, Guo, and Wu, in 2020 [43], submitted research on planning and symmetrical planning for parallel shipbuilding projects in an open distributed production environment. This proposal allowed for effective planning of tasks for more efficient use of distributed resources, support for multilateral cooperation, assistance to developers of decision-making methods, and further assistance to project managers in the shipbuilding process by repeating iterative combined auction (ICA) during integration planning and project planning. The authors gave a visual example of its effectiveness and justified using the proposed approach.

In the study published by Lou, Ong, and Nee, in 2010 [3], a distributed scheduling approach utilizing multi-agent systems was proposed for the assignment of tasks to machines. The approach is designed to work in fast-changing virtual job shop environments and employs market mechanisms and a negotiation-based distributed scheduling method. In 2020, Cheng, Bi, Tao, and Ji [44] presented a novel approach for distributed scheduling and collaborative manufacturing operations within the context of smart manufacturing. Their hypernet-based manufacturing service integrates cloud services and real-time information to facilitate the scheduling process and uses graph coloring and an ABC algorithm to solve the manufacturing scheduling problem. The approach was evaluated through experiments in different distributed and collaborative manufacturing operation scenarios and was found to have better performance compared to existing methods.

Han et al., in 2022 [45], proposed a solution to minimize energy consumption while ensuring resource utilization in a multi-factory production environment. They proposed a new algorithm called “VNIG”, based on the “greedy” technique and incorporating elements of learning, which was found to have better results compared to existing methods. Manupati et al., in 2014 [46], developed a solution for improving the efficiency of network-based manufacturing systems by optimizing multiple objectives such as completion time and workload distribution. The solution includes a mathematical model and two algorithms, the “Modified Block-based Genetic Algorithm” and the “Modified Non-dominated Sorting Genetic Algorithm”, which were found to be more effective compared to previous algorithms.

Several performance measures such as makespan with more than one machine case in a Flowshop scheduling problem with dependent processing times were solved by several authors, namely in the works described in [8,9,25,27,45].

#### *2.4. Blockchain in Distributed Manufacturing Scheduling*

Kapitonov et al., in 2017 [20], discussed the implementation of the BC model in autonomous object decision-making processes in an unconstrained environment. Sikorski et al., in 2017 [21], leveraged blockchain technology to improve efficiency and reduce costs in machine-to-machine communication. Skowronski et al., in 2019 [22], explored the potential for using blockchain in a multi-agent model for improved decision-making in intelligent production systems. Yang et al., in 2019 [23], proposed a blockchain-enabled digital twin algorithm that eliminates the delay between planning and execution through a smart contract that automatically plans and schedules machines based on workpiece requirements.

Assaqtly et al., in 2020 [24], described the practicalities that exist for collaborating between the manufacturing environment through the blockchain that are suitable for smart manufacturing. Meanwhile, Cambou et al., in 2020 [47], focused on including blockchain in the context of additive manufacturing to overcome or eliminate attacks by intermediaries.

Moreover, Westerkamp et al., in 2019 [48], mentioned the potential for blockchain-based methods to improve the security aspects of production systems.

Blockchain technology has been proposed by researchers such as Xu and Sauri [49] to improve the privacy and security of data transmission and communication on the Internet of Things (IoT). Scholars such as Vangala et al. [50] have also proposed a generalized architecture using blockchain technology in smart agriculture for security purposes. Furthermore, in [49], a smart manufacturing security model was proposed that uses blockchain technology to enhance security, privacy, and tamper protection. Kumar et al. [51] presented an example of using the blockchain structure to prevent fraud scenarios and secure a logistics business.

Several authors are studying the use of blockchain technology for data exchange and data security due to its characteristics of immutability, transparency, automation, and integrity. For instance, in [52], a partially decentralized IIoT architecture was proposed using blockchain technology to improve security and privacy. Shabazi and Bain [53] proposed integrating blockchain technology and machine learning to address security and data management issues in smart manufacturing.

Assakti et al. [24] believe that blockchain technology can help improve coordination and collaboration in the smart manufacturing industry. Some have suggested using blockchain technology to create private systems to keep track of products and materials while maintaining privacy. Bai et al. [54] suggested using a blockchain-based platform called BPIIoT to process all transactions on the blockchain network.

Researchers have also suggested using blockchain technology to improve manufacturing. One idea is to use it to control machines and make sure they are doing the job correctly. Another idea is using it to ensure data and transactions are secure. Some researchers want to ensure that all different parts of the company can use blockchain technology in the same way [55–57].

Sgarbossa et al., in 2020 [58], proposed a conceptual model for cloud material handling systems and explores the use of cloud computing for scheduling material handling activities. Peron et al., in 2020 [59], highlighted the importance of considering sustainability in facility layout planning and the potential role of emerging technologies in this context. The results of this study could be valuable for companies looking to implement sustainable facility layout strategies and could inspire further research in this area.

The field of blockchain has seen a lot of development in recent years, but there is still a lack of studies focusing specifically on blockchain scheduling. Most papers in this area have only proposed frameworks for scheduling; none have actually implemented blockchain-based smart contracts for scheduling in the context of distributed production. The proposed framework in this study aims to fill this gap by implementing blockchain-based smart contracts for scheduling manufacturing enterprises with due dates.

The framework, as shown in Figure 3, consists of different layers that work together to achieve the scheduling objectives. The physical layer acts as a bridge between the data-scanning layer and the blockchain layer. Data collected by the data-scanning layer through various IoT devices can be integrated with the blockchain using the physical layer. The smart contract and consensus mechanism help in the scheduling process in the distributed manufacturing system. In the planning layer, all the tasks are sorted according to their priority, and the allocation of tasks and time for each task is carried out using specific machines. The proposed framework for BCDMS helps to exchange data between machines using sensors distributed in different locations. The data-scanning layer helps the scheduling agent to know the actual state of the machines, which enables the agent to change schedules based on the situation.

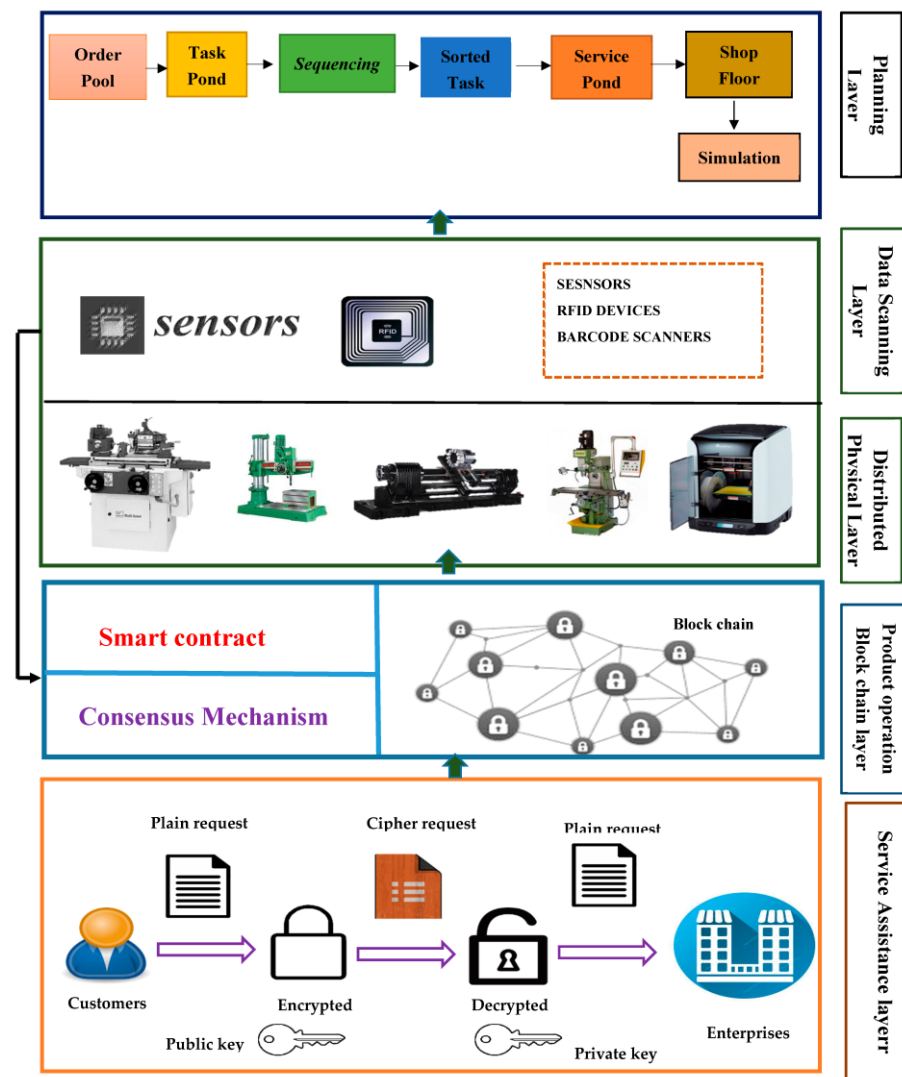


Figure 3. Framework for blockchain-based collaborative distributed manufacturing scheduling.

The product operational blockchain layer, as shown in Figure 3, uses smart contracts to make decisions without human intervention, based on the situation that arises during the manufacturing process. This helps to create a trusting and secure environment between actors in distributed production systems. The service level is a way to collect the information needed to fulfil orders and to help customers receive products. Overall, this proposed approach will help to address the gap in the literature and improve the efficiency of scheduling in distributed manufacturing systems.

### 3. Proposed Blockchain-Based Collaborative Distributed Manufacturing Scheduling Approach

The proposed BCDMS is based on the concept of joint and decentralized production units or collaborative networks, which can function independently and produce products. The key idea is to establish a peer-to-peer network between these enterprises using blockchain technology, enabling coordination and collaboration among them. Each enterprise has its own production facilities for manufacturing products.

The concept behind the proposed BCDMS system is to have a network of distributed production units or collaborative networks that can work and produce products independently. Each company has its own production facilities. By using blockchain technology, these companies form a peer-to-peer network, allowing them to coordinate their actions.



Orders from customers are received by each enterprise and their equipment schedules are organized accordingly. The machines in each company are connected to sensors, which act as digital twins to the machine. The data collected by these sensors are continuously analyzed and stored.

Each machine in each enterprise has an IoT device, which detects any available time that is not being used based on the existing schedule. The machine shares information such as processing time, free time, and processing capabilities through the network via the sensors. This information is then issued as a smart contract on the blockchain. If the enterprise's machine capacity is not sufficient to complete the operations required for product production, the IoT device searches the network for a smart contract that meets the necessary specifications. If the requirements are met, the smart contract allows the use of other available machines in other enterprises and adds the transaction to the blockchain.

### *3.1. Proposed Methodology Based on Blockchain Technology*

In the basic structure of a collaborative distributed manufacturing system, manufacturing units or enterprises are connected to gain a competitive advantage in the global market. Industry 4.0 brought us several emerging technologies that help to exchange information and resources and transfer technologies within manufacturing units or enterprises. However, a major drawback is that trust among them is required. It is high time for organizations to adopt a technology that helps to solve the trust-based problems in the collaborative distributed manufacturing system. The emerging blockchain technology (BCT) offers the entities or organizations in the distributed manufacturing environment an option besides trusting each other blindly, where blockchain technology will take care of this issue. Not only that, BCT also improves transparency and security with its immutable features.

The data exchanged and shared between peers are stored as a distributed ledger on the blockchain. The blockchain architecture allows data to be transparent, secure, and immutable. BCT allows you to eliminate intermediaries or trusted third parties. A consensus protocol is maintained between peers and transactions are encrypted using a cryptographic method. Objects in the blockchain have permission to access the blockchain and can verify the registry at any time. Thus, the blockchain provides instant, comprehensive transparency, and BCT can easily track resources and transactions since any added to the blockchain are difficult to duplicate.

### *3.2. Smart Contracts*

Szabo et al., in 1997 [60], discussed the basic idea of smart contracts, which combine computer protocols with user interfaces to enforce the terms of a contract. With the advent of blockchain, smart contracts are becoming more and more popular due to the possibility of using blockchain compared to the technologies available at the time of the invention 20 years ago.

These BCT-based smart contracts have the capability to replace physical assets and humans such as lawyers and banks by executing contracts automatically [61]. Code logic written in any computer language defines functions, operations, or rules between entities that are automatically executed when a smart contract is initiated. Smart contracts can be written in Java or Solidity on the Fabric, Corda, and Ethereum platforms. The proposed model for a smart contract is shown in Figure 4. Next, in Figure 5, the algorithm expressing the logic code for the proposed smart contract for BCDMS is presented.

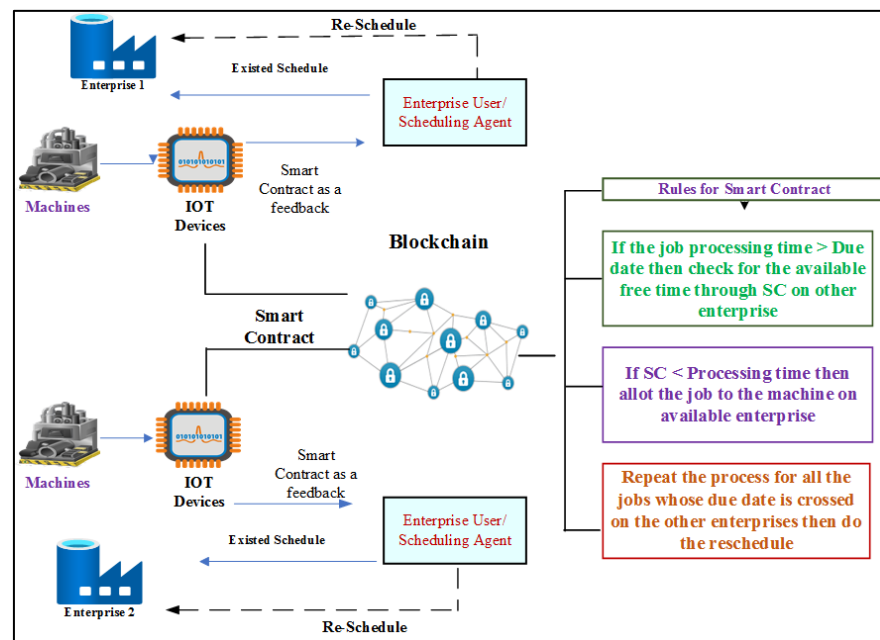


Figure 4. Proposed model for a smart contract in blockchain-based distributed manufacturing scheduling.

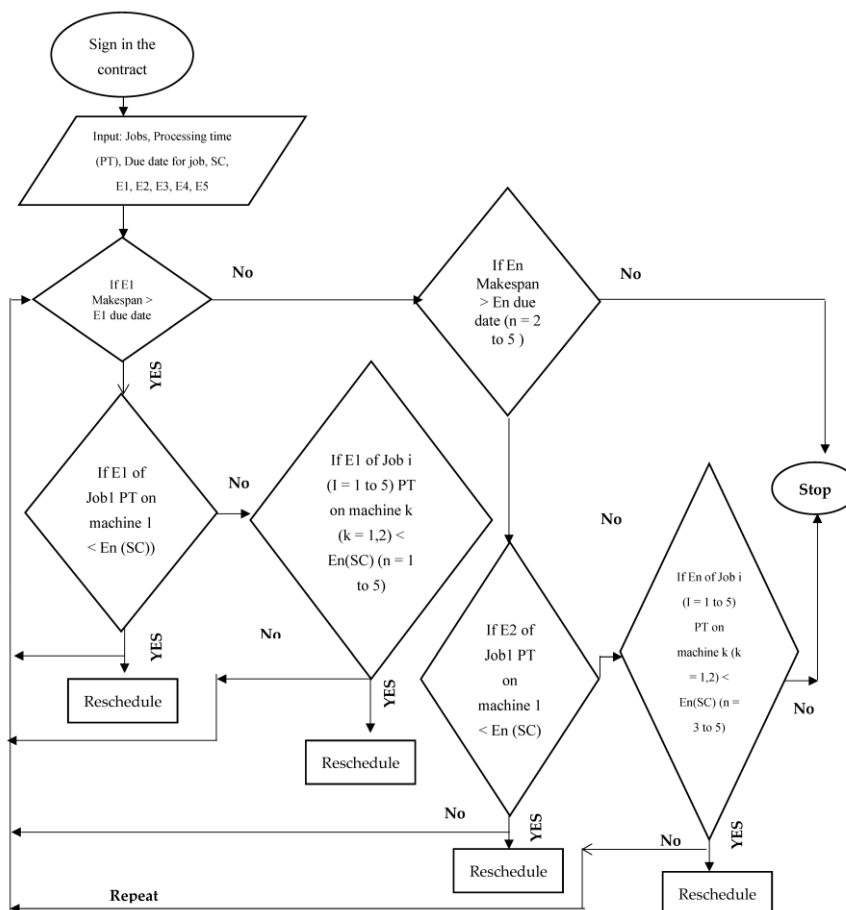


Figure 5. Algorithm expressing the logic code for the proposed smart contract for BCDMS.

#### 4. Illustrative Example

In this work, a distributed manufacturing system consisting of five enterprises placed at various locations is considered. An enterprise has the capability to manufacture products depending on the orders received from various customers.

Each enterprise has two machines. They have the ability to handle tasks or actions assigned to them since the system under consideration is by its nature a DMS. Enterprises (E1, E2, E3, E4, and E5) have the advantage of sharing tasks or activities when needed. Here in this scenario looking at five jobs that should be processed in each organization with two available machines. Initially, planning is carried out according to the Jackson model [10,11], or in case of manufacturing environments with three or more machines, an extended version of this model can be used [11]. Moreover, the proposed scheduling approach can be further used in combination with other techniques for solving management problems occurring in networked environments, often by using approaches based on social network analysis (SNA) [62]. Other approaches, for instance, include the application of diverse kinds of multi-criteria decision-making methods, along with varying kinds of data normalization techniques [63]. These kinds of approaches, among a range of others, play a crucial role currently in Industry 4.0 [64].

In the case study considered in this work, after the application of the Jackson model, preference was given for processing tasks according to the suggested scheduling approach, with some tasks having crossed their deadlines. Moreover, special attention was also given to obtaining an optimal sequence, with a minimized makespan. To that end, additional procedures were considered in the proposed scheduling process.

Initially, free time was determined according to the schedules reached. This free time is the time available on the machine between the scheduled job and the next sequential job. A smart contract was developed for checking the jobs of each enterprise, complete with due dates and processing times. As soon as any job passed its due date, the smart contract would try to find free time in other enterprises to assign the job to the same machine. In that way, the job would be completed before the due date. This procedure was carried out for all jobs in the enterprise. In this way, the due date can be met without affecting the other products' timely delivery.

## 5. Results and Discussion

Ethereum is a blockchain platform that enables the creation of smart contracts. Smart contracts are self-executing contracts with the terms of the agreement written directly in code. They can be used to automate various processes, including planning and scheduling in a DMS (document management system). Transactions associated with the smart contract are recorded on the Ethereum blockchain, providing immutability and transparency. The smart contract can be programmed to accept specific parameters, such as the machine number, task number, and task due date, for scheduling purposes.

Smart contracts can be used to manage tasks and resources. Information such as machine numbers, task numbers, and due dates is entered into the smart contract. The smart contract can then use this information to decide how to allocate resources based on availability and priority. The smart contract can also automatically schedule tasks based on this information.

Figures 6 and 7 are ways for people to see and interact with this smart contract, where they can view details such as task numbers, due dates, and resource allocation. The two show smart contracts created for different enterprises. Each contract may have specific parameters that are relevant to the enterprise, such as the number of machines, tasks, and their respective due dates. These parameters can then be used by the smart contract to automate processes such as scheduling and allocation of resources within the enterprise. Smart contracts can also be programmed to automate other processes within the enterprise depending on the specific needs and requirements of the enterprise.

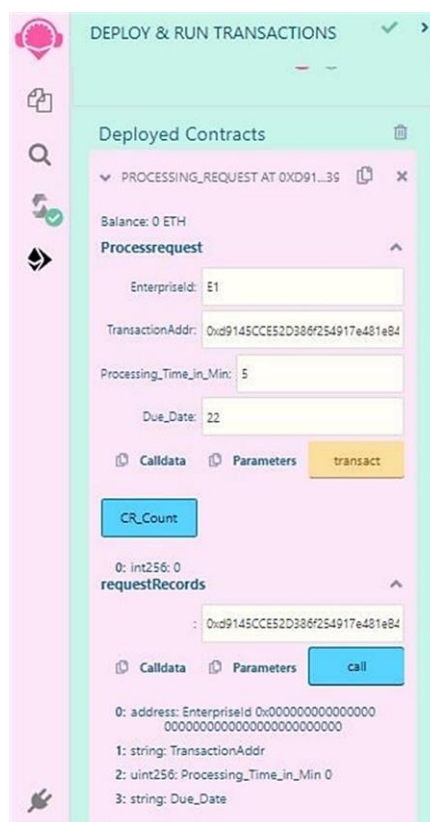


Figure 6. Screen shot of developed smart contract for process planning and scheduling in distributed manufacturing systems.

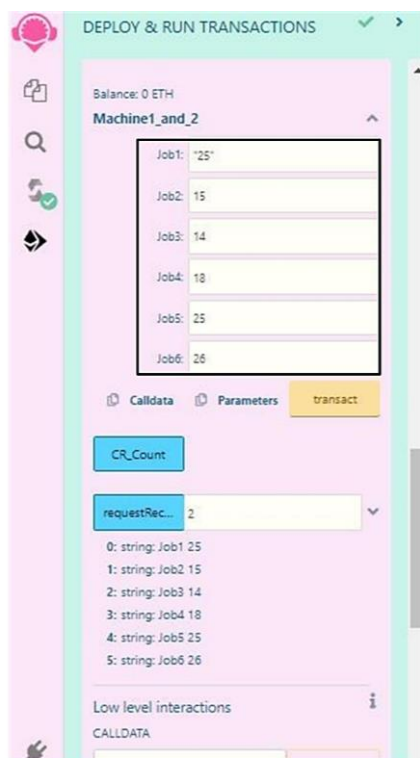


Figure 7. Screen shot of developed smart contract for process planning and scheduling in distributed manufacturing systems.



Smart contracts are coded in Ethereum. The proposed system consists of the following: AMD-8550U Processor @ 2.90GHz, 64GB RAM, running Ubuntu 20.04 LTS. To implement the proposed system, Remix Ethereum-Pragmatic Solidity 0.8.4 was used.

Figure 8 clearly shows that after the execution of a smart contract, job 2 for enterprise 1, which crossed the due date, was allocated to enterprise 4 when there was free time available at enterprise 4 on machine 1. Similarly, the blockchain transactions for all the five cases are shown in Figures 8–12. In Figure 9, job 3 that belongs to E2 has crossed its due date and been allocated to enterprise 5 when there is free time available on machine 1.

```
[vm]from: 0x5B3...eddC4to: distribution.check_enterprise(uint256) 0xd8b...33fa8value: 0
weidata: 0x163...0003logs: 0hash: 0x659...29aa3

Status          true Transaction mined and execution succeed
transaction hash 0x6598458fb65d759f528c53d15f05ab054a7bade043e7e72694c273dd2ea29aa3
from            0x5B38Da6a701c568545dCfcB03FcB875f56beddC4
to              DCE (uint256) 0xd8b934580fcE35a11B58C6D73aDeE468a2833fa8
gas             66638 gas
transaction cost 57728 gas
execution cost  57728 gas
input           1 0x163...0002
decoded input   {"uint256 id": { "E1" }}
decoded output  {"0": "string: stat ACCEPT "}
logs            [ { "from": "0x5B38Da6a701c568545dCfcB03FcB875f56beddC4
"topic": "0x6598458fb65d759f528c53d15f05ab054a7bade043e7e72694c273dd2ea29aa
ff68", "event": "status of enterprise",
"args
"0": "Allocate E1 of JOB 2 of M1 to E4 OF FREE SLOT SC ON M1"
"s": "Allocate E1 OF JOB 2 of M1 to E4 OF FREE SLOT SC ON M1"

Value          0 wei
```

Figure 8. Transactions stored on the blockchain for case 1—E1.

```
[vm]from: 0x5B3...eddC4to: distribution.check_enterprise(uint256) 0xd8b...33fa8value: 0
weidata: 0x163...0003logs: 0hash: 0x659...29aa3

Status          true Transaction mined and execution succeed
transaction hash 0xebc9db98c666da773be773b61da2bbd1f12a94616035bb60a0bce3346407da75
from            0x5B38Da6a701c568545dCfcB03FcB875f56beddC4
to              DCE (uint256) 0xd8b934580fcE35a11B58C6D73aDeE468a2833fa8
gas             66638 gas
transaction cost 57728 gas
execution cost  57728 gas
input           1 0x163...0002
decoded input   {"uint256 id": { "E2" }}
decoded output  {"0": "string: stat ACCEPT "}
logs            [ { "from": "0x5B38Da6a701c568545dCfcB03FcB875f56beddC4
"topic": "0x637df8ab02c4eb7cd2966fb45354b19512b5b216fcddb553643ed1cc6577
ff68", "event": "status of enterprise",
"args
"0": "Allocate E2 of JOB 3 of M1 to E5 OF FREE SLOT SC ON M1"
"s": "Allocate E2 OF JOB 3 of M1 to E5 OF FREE SLOT SC ON M1"

Value          0 wei
```

Figure 9. Transactions stored on the blockchain for case 2—E2.

```

[vm] from: ex583...eddc4 to: distribution.check_enterprise (uint256) 8x7EF...8C847
value: 0 wei data: 8x163...eeee1 logs: 1 hash: exc19...6cd26

Status          true Transaction mined and execution succeed
transaction hash 0x83925fa78ca72511dac56b4ef63487cdb9fa8c06b191c358d6b5877a29bbd0fa
from            0x0A098Eda01Ce92ff4A4CCb7A4FFFB5A43EBC70DC
to             DCE (uint256) xd9145CCE52D386f254917e481Eb44ee9943F39138
gas            66638 gas
transaction cost 55648 gas
execution cost  55648 gas
input          1 0x163...0003
decoded input   {"uint256 id": { "E3" } }
decoded output  {"0": "string: stat ACCEPT "}
logs           [ { "from": "0x0A098Eda01Ce92ff4A4CCb7A4FFFB5A43EBC70DC
"topic": "xab296d706472bbetdebd72681fa65f89a47fe995f95e84f43c38556162357,
, "event": "status of enterprise",
"args
"0": "Allocate E4 of JOB 6 of M1 to E5 OF FREE SLOT SC ON M1"
"s": "Allocate E4 OF J08 6 of M1 to E5 OF FREE SLOT SC ON M1"

Value          0 wei

```

Figure 10. Transactions stored on the blockchain for case 3—E3.

```

[vm] from: 0x583...eddc4 to: distribution.check_enterprise (uint256) 8x7EF...8C847
value: 0 wei data: 8x163...eeee1 logs: 1 hash: exccd7...01c58

Status          true Transaction mined and execution succeed
transaction hash 0xdd72bb49a4cb729538c899e8263d5bid1a145822e6b3d7c3f328cb4868158
from            0x5B38Da6a701c568545dCfcB03FcB875f56beddC4
to             DCE (uint256) xd869345881511858607324654233a8
gas            68638 gas
transaction cost 59948 gas
execution cost  56618 gas
input          1 0x163...0004
decoded input   {"uint256 id": { "E4" } }
decoded output  {"0": "string: stat ACCEPT "}
logs           [ { "from": "0xd8b934580fcE35A11b58C6D73aDeE468a2833faB
"topic": "xab296d706472bbetdebd72681+a65f89a47fe995+95e84f43c38556162357,
, "event": "status of enterprise",
"args
"0": "Allocate E2 or JOB 3 of M2 to E3 OF FREE SLOT SC ON M2"
"s": "Allocate E2 OF J08 3 of M2 to E3 OF FREE SLOT SC ON M2"

Value          0 wei

```

Figure 11. Transactions stored on the blockchain for case 4—E4.

```

[vm] from: ex583...eddc4 to: distribution.check_enterprise (uint256) 8x7EF...8C847
value: 0 wei data: 8x163...eeee1 logs: 1 hash: exc19...6cd26

Status          true Transaction mined and execution succeed
transaction hash Bxc191816de7961+20062422ee7c7a8cb54d88ea4468e56de99d84946cd26 D
from            0x5B38Da6a701c568545dCfcB03FcB875f56beddC4
to             DCE (uint256) 0x7EF2e0048f5bAeDe046f6Bf07979f4Eda8CD47
gas            66388 gas
transaction cost 57728 gas
execution cost  57728 gas
input          1 0x163...0001
decoded input   {"uint256 id": { "5" } }
decoded output  {"0": "string: stat ACCEPT "}
logs           [ { "from": "8x7FF20084845A=D646+6h797944ED9C847
"topic": "296d786472hbelocebd72681++65+d89a47fe095+95e84f43c3859616235,
"event": "status of enterprise",
"args
"0": "Allocate E4 or JOB 6 of M2 to E3 OF FREE SLOT SC ON M2"
"s": "Allocate E4 OF J08 6 of M2 to E3 OF FREE SLOT SC ON M2"

Value          0 wei

```

Figure 12. Transactions stored on the blockchain for case 5—E5.

In Figure 9, after the execution of a smart contract, job 6 for enterprise 4, which crossed the due date, has been allocated to enterprise 5 when there is a free time available on machine 1.

Figures 13–17 show the initial schedules of all jobs on the machines of each enterprise. Figure 13 shows the schedule of all the jobs on the machines for enterprise 1. The makespan of all the jobs in this case is 36. Red indicates job 1, yellow indicates job 2, grey indicates job 3, cyan indicates job 4, blue indicates job 5, and green indicates the smart contract, i.e., free time on the machines.

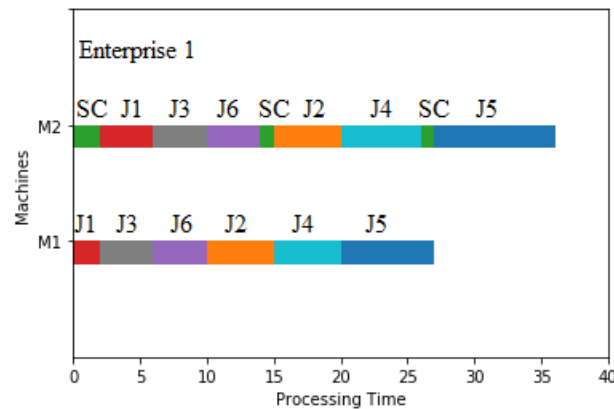


Figure 13. Initial Gantt chart for enterprise 1.

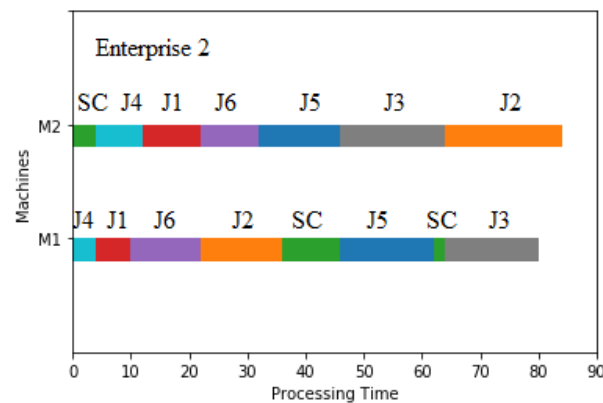


Figure 14. Initial Gantt chart for enterprise 2.

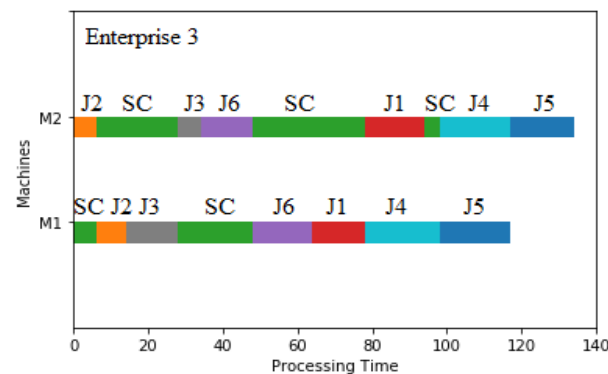


Figure 15. Initial Gantt chart for enterprise 3.

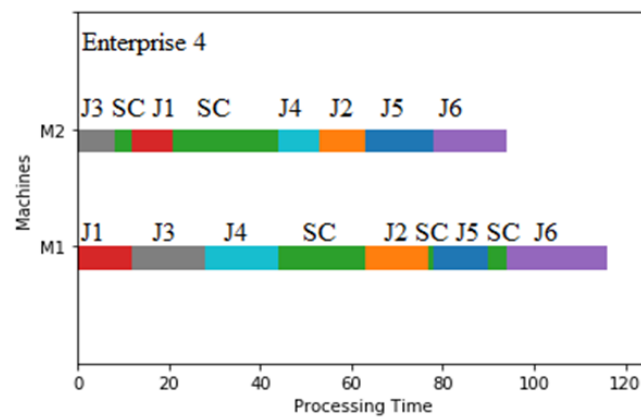


Figure 16. Initial Gantt chart for enterprise 4.

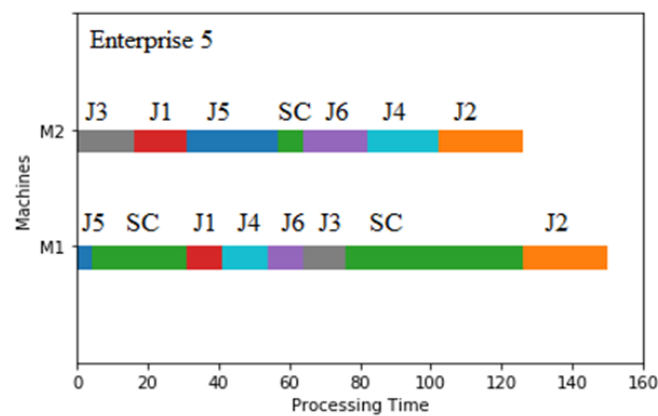


Figure 17. Initial Gantt chart for enterprise 5.

Figure 14 shows the schedule of all the jobs on to the machines for enterprise 2. The makespan of all the jobs in this case is 82. Figure 15 indicates the makespan of all the jobs at enterprise 3, with a value of 132. Figure 16 shows the schedule of all the jobs at enterprise 4, indicating a makespan of 116. Figure 17 shows the schedule of all the jobs at enterprise 5, in this case, with a makespan of 155.

To demonstrate the proposed block chain-based system, in this work, an example is considered that contains five different independent enterprises (E1, E2, E3, E4, E5) involved in networked distributed manufacturing. Each of these enterprises (E1–E5) contains two different machines in the job shop environment. For each task, the corresponding type of machine M and the processing time for each operation t are displayed in Figure 6. DD specifies the due date for each job.

Initially, each machine is scheduled according to Jackson's algorithm based on the processing time and operation priorities. Each free time (time > 1) available on the machine is tagged and published as a smart contract (SC) on the Gantt charts shown in Figures 13–17. A Gantt chart provides a pictorial representation of process planning and scheduling. It provides details such as when various operations are performed and how much time is required to perform tasks or operations on each machine. All the rescheduled Gantt charts are shown in Figures 18–22.



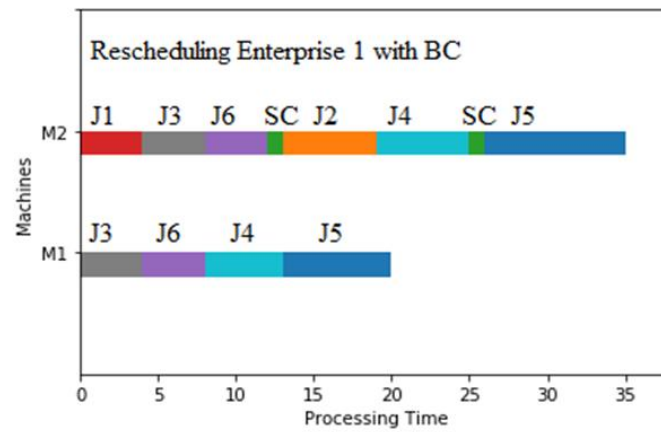


Figure 18. Rescheduled Gantt chart for enterprise 1.

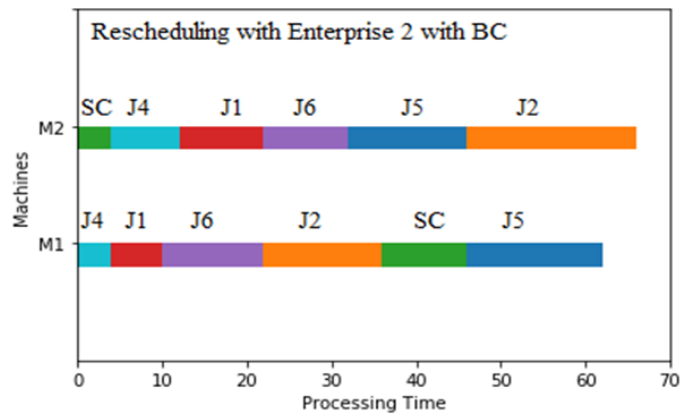


Figure 19. Rescheduled Gantt chart for enterprise 2.

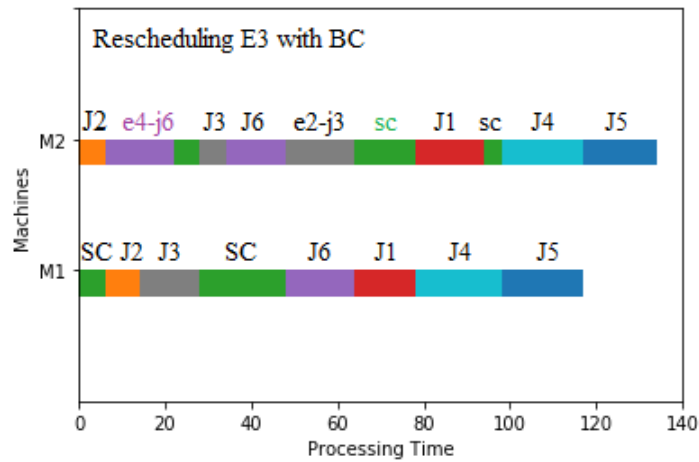


Figure 20. Rescheduled Gantt chart for enterprise 3.

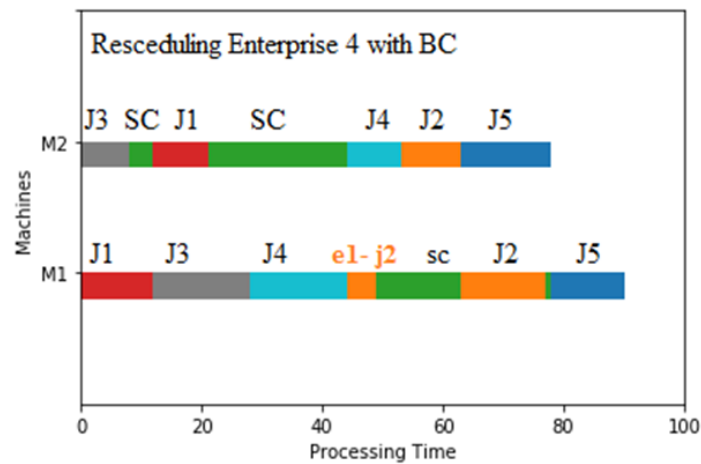


Figure 21. Rescheduled Gantt chart for enterprise 4.

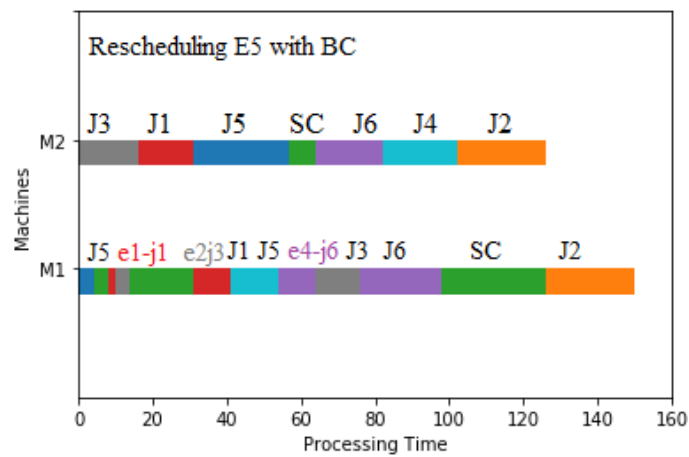


Figure 22. Rescheduled Gantt chart for enterprise 5.

Similarly, the IOT device installed on machine M1 of enterprise E2 starts searching for a suitable smart contract with free time at other enterprises (E1, E3, E4, and E5) for the same machine M1. In this way, E2 is able to complete the order before the due date that is shown in Figure 19, using a rescheduling process carried out by using the block chain process based on smart contracts.

After redistribution of jobs using blockchain-based smart contracts, most of the deadlines were met without changing the actual schedules in the DMS being considered. The important point is that this model can be extended to solve more complex planning and scheduling problems, under different kinds of application scenarios. There are many opportunities for integrating scheduling algorithms in the context of distributed manufacturing systems.

On carefully checking Table 2, job 3 at enterprise 2 has a DD of 75, but it has already passed the DD in the schedule. In such cases, the IOT device installed on machine 2 of enterprise 2 starts searching for a suitable smart contract that has free time in other enterprises (E1, E3, E4, and E5) for the same machine 2. The smart contract runs with a logic that is expressed in Figure 4. After the execution of the smart contract, the processing time of job 2 on the machine is less than the free time SC for enterprise 3. Immediately rescheduling is implemented to E3, a change marked as E2-J3 on E3 after the rescheduling. It does not affect the makespan of the pre-existing jobs at E3.

**Table 2.** Input data considered for distributed manufacturing enterprises with due dates.

E1					E2					E3					E4					E5				
Job	Op	Mac	PT	DD	Job	Op	Mac	PT	DD	Job	Op	Mac	PT	DD	J	o	M	PT	D	J	O	M	T	D
1	i	P	2		1	i	P	6		1	i	P	18		1	i	P	12		1	i	Q	15	
	ii	Q	4	10		ii	Q	10			ii	Q	16		ii	Q	9		ii	P	10			
2	i	P	5		2	i	P	14		2	i	Q	6	15	2	i	Q	10	22	2	i	Q	24	102
	ii	Q	6	14		ii	Q	20			ii	P	8		ii	P	14		ii	P	7			
3	i	P	4		3	i	Q	18		3	i	P	14		3	i	P	16		3	i	P	16	
	ii	Q	4	12		ii	P	16	75		ii	Q	6		ii	Q	8		ii	Q	12			
4	i	Q	5		4	i	P	4		4	i	P	20		4	i	P	16	52	4	i	P	13	93
	ii	P	6			ii	Q	8			ii	Q	20		ii	Q	9		ii	Q	20			
5	i	P	7		5	i	Q	14		5	i	P	19		5	i	Q	15		5	i	P	4	
	ii	Q	9	16		ii	P	16			ii	Q	17	20	ii	P	12		ii	Q	26			
6	i	Q	12		6	i	P	12		6	i	Q	14		6	i	Q	18		6	i	P	14	
	ii	P	4			ii	Q	10			ii	P	14		ii	P	22		ii	Q	18			

**6. Discussion and Results**

The results of the proposed BCDMS model were analyzed and evaluated against various test scenarios [65] using GAMS software and CPLEX solver. The performance of the model was evaluated in terms of makespan and machine utilization. A time limit of 3600 s was set for solving the test scenarios mentioned in Table 3. A comparison of the results obtained from the CPLEX GAMS solver and the proposed BCDMS is shown in Table 3. The results indicate that the proposed BCDMS model was able to optimize the objective functions of the scheduling problem and achieved good results in terms of makespan and machine utilization, as demonstrated by [65–67].

**Table 3.** Comparison of results obtained by CPLEX solver in GAMS software with proposed BC method.

Test Scenario (Small-Sized Problems)	Jobs × Machines	CPLEX Solver in GAMS		BCDMS Model	
		Makespan	Machine Utilization	Makespan	Machine Utilization
Test Scenario 1	2 × 2	33	0.6	31	0.63
Test Scenario 3	2 × 2	39	0.54	36	0.61
Test Scenario 3	3 × 2	36	0.51	33	0.56
Test Scenario 4	3 × 2	45	0.53	43	0.56
Test Scenario 5	3 × 2	49	0.59	41	0.67
Test Scenario 6	3 × 2	44	0.53	39	0.64
Test Scenario 7	3 × 3	49	0.54	43	0.63
Test Scenario 8	3 × 4	54	0.56	46	0.59
Test Scenario 9	3 × 5	51	0.58	39	0.63
Test Scenario 10	4 × 5	53	0.51	43	0.61

The results in Table 3 indicate that the proposed BCDMS approach outperforms the standard CPLEX solver with GAMS software in terms of a lower makespan and higher machine utilization. This demonstrates the superior performance of the proposed BCDMS model. However, it should be noted that when the number of jobs exceeds 4 or 5, the BCDMS model struggles to solve the problem as the computational complexity increases. This highlights the need for further improvements in the proposed BCDMS model to handle larger and more complex scheduling problems in the future.

## 7. Managerial and Academic Implications

The manufacturing industry is changing and becoming more customer-oriented thanks to new technologies. To stay competitive, companies or enterprises can form a group called a collaborative distributed manufacturing system (CDMS). However, there is a problem of mistrust between companies or enterprises. To solve this problem, the authors propose to use blockchain technology, which provides greater security and transparency. Thus, companies can exchange information without completely trusting each other. The authors believe that it is important for companies to implement this technology to improve their security and competitiveness. The authors' work makes a significant contribution to the field of distributed manufacturing systems (DMS) and blockchain technology. Previous studies have explored the utilization of blockchain in DMS through the proposal of various frameworks. However, few works have been carried out in the implementation of smart contracts in supply chain management for product tracking, and very little attention has been paid to the implementation of smart contracts in scheduling. The current study represents a meaningful advancement in the application of blockchain technology and smart contracts within the realm of collaborative distributed manufacturing systems (CDMS). The implementation of Ethereum-based smart contracts for scheduling in a manufacturing system highlights the uniqueness and potential of this approach. The findings of this work suggest there is significant scope for further exploration of the utilization of blockchain-technology-based smart contracts in increasingly complex manufacturing systems for scheduling purposes.

## 8. Conclusions

In this paper, a novel approach to support blockchain-based distributed manufacturing systems (BCDMS) was presented. The proposed model extends Jackson's model, allowing for the scheduling of jobs in classical job shops with multiple machines, as well as in the context of extended and virtual enterprises (EME/VE). The model introduces a blockchain-based smart contract that facilitates the secure and transparent exchange of information in a distributed production system, built on the Ethereum platform.

The proposed BCDMS approach was analyzed through its application in three different architecture models, namely centralized, P2P, and hybrid. Based on an illustrative scenario, the suitability of each architecture for BCDMS was evaluated, and it was concluded that the proposed hybrid architecture provides the best balance between effort reduction and efficiency.

In future work, the potential benefits and limitations of the proposed model will be further analyzed by applying it to real industrial case studies in various manufacturing environments.

**Author Contributions:** In this paper, the conceptualization and methodology definition were established by V.K.M., L.V. and G.P.; the main investigation, preparation, and writing—original draft were carried out by V.B.R., L.V. and V.K.M.; the writing—review and editing and the visualization were jointly carried out by V.B.R., L.V. and V.K.M.; the general supervision of this work was performed by Prof. V.K.M., L.V. and G.P.; and the project administration and funding acquisition were accomplished by L.V. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by national funds through the FCT—Fundação para a Ciência e Tecnologia “R&D Units Project”: UIDB/00319/2020 and EXPL/EME-SIS/1224/2021.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

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



## References

1. Putnik, G.D.; Putnik, Z.; Shah, V.; Varela, L.; Ferreira, L.; Castro, H.; Catia, A.; Pinheiro, P. Collaborative Engineering definition: Distinguishing it from Concurrent Engineering through the complexity and semiotics lenses. *IOP Conf. Ser. Mater. Sci. Eng.* **2021**, *1174*, 012027. [[CrossRef](#)]
2. Putnik, G.D.; Putnik, Z.; Shah, V.; Varela, L.; Ferreira, L.; Castro, H.; Catia, A.; Pinheiro, P. Collaborative Engineering: A Review of Organisational Forms for Implementation and Operation. *IOP Conf. Ser. Mater. Sci. Eng.* **2021**, *1174*, 012028. [[CrossRef](#)]
3. Lou, P.; Ong, S.K.; Nee, A. Agent-based distributed scheduling for virtual job shops. *Int. J. Prod. Res.* **2010**, *48*, 3889–3910. [[CrossRef](#)]
4. Vieira, G.; Varela, M.L.R.; Putnik, G.D. Technologies integration for distributed manufacturing scheduling in a virtual enterprise. In Proceedings of the International Conference on Virtual and Networked Organizations, Emergent Technologies, and Tools, Ofir, Portugal, 6–8 July 2011; Springer: Berlin/Heidelberg, Germany, 2012; pp. 337–347.
5. Eijnatten, F.M.; Putnik, G.D. *Chaos, Complexity, Learning, and the Learning Organization: Towards a Chaordic Enterprise*; The Learning Organization; Emerald Group Publishing: Bingley, UK, 2004.
6. Putnik, G.D.; Ferreira, L. Industry 4.0: Models, tools and cyber-physical systems for manufacturing. *FME Trans.* **2019**, *47*, 659–662. [[CrossRef](#)]
7. Guo, Z.; Ngai, E.; Yang, C.; Liang, X. An RFID-based intelligent decision support system architecture for production monitoring and scheduling in a distributed manufacturing environment. *Int. J. Prod. Econ.* **2015**, *159*, 16–28. [[CrossRef](#)]
8. Varela, M.L.R.; Putnik, G.D.; Cruz-Cunha, M.M. Web-based Technologies Integration for Distributed Manufacturing Scheduling in a Virtual Enterprise. *Int. J. Web Portals* **2012**, *4*, 19–34. [[CrossRef](#)]
9. Varela, M.L.R.; Ribeiro, R.A. Distributed Manufacturing Scheduling Based on a Dynamic Multi-criteria Decision Model. In *Recent Developments and New Directions in Soft Computing. Studies in Fuzziness and Soft Computing*; Zadeh, L., Abbasov, A., Yager, R., Shahbazova, S., Reformat, M., Eds.; Springer: Cham, Switzerland, 2014; Volume 317. [[CrossRef](#)]
10. Jackson, J.R. An extension of Johnson's results on job lot scheduling. *Nav. Res. Logist. Q.* **1956**, *3*, 201–203. [[CrossRef](#)]
11. Varela, L.R.; Alves, C.F.V.; Santos, A.S.; Vieira, G.G.; Lopes, N.; Putnik, G.D. Analysis of a Collaborative Scheduling Model Applied in a Job Shop Manufacturing Environment. *Machines* **2022**, *10*, 1138. [[CrossRef](#)]
12. Alves, F.; Varela, M.L.R.; Rocha, A.M.A.; Pereira, A.I.; Leitão, P. A human centered hybrid MAS and meta-heuristics based system for simultaneously supporting scheduling and plant layout adjustment. *FME Trans.* **2019**, *47*, 699–710. [[CrossRef](#)]
13. Wilczyński, A.; Kołodziej, J. Modelling and simulation of security-aware task scheduling in cloud computing based on Blockchain technology. *Simul. Model. Pract. Theory* **2020**, *99*, 102038. [[CrossRef](#)]
14. Thakur, A.; Chaudhary, N.; Tilokani, P.; Manupati, V.; Costa, E.; Varela, L.; Machado, J. A cyber-physical system based collaborative distributed manufacturing system architecture for intelligent manufacturing. In Proceedings of the Regional HELIX'17-International Conference on Innovation, Entrepreneurship and Technology Transfer, Covilha, Portugal, 21–23 June 2017.
15. Kays, E.; Karim, A.; Varela, L.; Putnik, G.; Ávila, P. Ranked sequence positional weight heuristic for simultaneous balancing and scheduling jobs in a distributed manufacturing environment. *Procedia CIRP* **2018**, *67*, 3–7. [[CrossRef](#)]
16. Ramakurthi, V.; Manupati, V.; Machado, J.; Varela, L. A Hybrid Multi-Objective Evolutionary Algorithm-Based Semantic Foundation for Sustainable Distributed Manufacturing Systems. *Appl. Sci.* **2021**, *11*, 6314. [[CrossRef](#)]
17. Ramakurthi, V.; Manupati, V.; Varela, M.L.R.; Putnik, G. A Novel Integrated Framework Approach for TEBC Technologies in Distributed Manufacturing Systems: A Systematic Review and Opportunities. In *Innovations in Mechanical Engineering (ICIE 2021), Lecture Notes in Mechanical Engineering*; Machado, J., Soares, F., Trojanowska, J., Ottaviano, E., Eds.; Springer: Berlin/Heidelberg, Germany, 2021; Volume 1. [[CrossRef](#)]
18. Ramakurthi, V.; Manupati, V.; Machado, J.; Varela, L.; Babu, S. An innovative approach for resource sharing and scheduling in a sustainable distributed manufacturing system. *Adv. Eng. Inform.* **2022**, *52*, 101620. [[CrossRef](#)]
19. Varela, L.; Putnik, G.; Romero, F. The concept of collaborative engineering: A systematic literature review. *Prod. Manuf. Res.* **2022**, *10*, 784–839. [[CrossRef](#)]
20. Kapitonov, A.; Lonshakov, S.; Krupenkin, A.; Berman, I. Blockchain-based protocol of autonomous business activity for multi-agent systems consisting of UAVs. In Proceedings of the Workshop on Research, Education and Development of Unmanned Aerial Systems (RED-UAS), Linköping, Sweden, 3–5 October 2017; pp. 84–89. [[CrossRef](#)]
21. Sikorski, J.J.; Houghton, J.; Kraft, M. Blockchain technology in the chemical industry: Machine-to-machine electricity market. *Appl. Energy* **2017**, *195*, 234–246. [[CrossRef](#)]
22. Skowroński, R. The open blockchain-aided multi-agent symbiotic cyber-physical systems. *Futur. Gener. Comput. Syst.* **2019**, *94*, 430–443. [[CrossRef](#)]
23. Yang, X.; Wang, G.; He, H.; Lu, J.; Zhang, Y. Automated Demand Response Framework in ELNs: Decentralized Scheduling and Smart Contract. *IEEE Trans. Syst. Man Cybern. Syst.* **2019**, *50*, 58–72. [[CrossRef](#)]
24. Assaqt, M.I.S.; Gao, Y.; Hu, X.; Ning, Z.; Leung, V.C.M.; Wen, Q.; Chen, Y. Private-Blockchain-Based Industrial IoT for Material and Product Tracking in Smart Manufacturing. *IEEE Netw.* **2020**, *34*, 91–97. [[CrossRef](#)]
25. Pinedo, M.; Chao, X. *Operations Scheduling with Applications in Manufacturing and Services*; McGraw Hill: New York, NY, USA, 1999.
26. Baker, K.R.; Trietsch, D. Safe scheduling: Setting due dates in single-machine problems. *Eur. J. Oper. Res.* **2009**, *196*, 69–77. [[CrossRef](#)]
27. Pinedo, M.L. *Scheduling Theory, Algorithms and Systems*, 4th ed.; Springer: Berlin/Heidelberg, Germany, 2012.

28. Zijm, W.; Kals, H. The Integration of Process Planning and Shop Floor Scheduling in Small Batch Part Manufacturing. *Ann. CIRP* **1995**, *44*, 429–432. [[CrossRef](#)]
29. Shen, W. Distributed manufacturing scheduling using intelligent agents. *IEEE Intell. Syst.* **2002**, *17*, 88–94. [[CrossRef](#)]
30. Aytug, H.; Lawley, M.A.; McKay, K.; Mohan, S.; Uzsoy, R. Executing production schedules in the face of uncertainties: A review and some future directions. *Eur. J. Oper. Res.* **2005**, *161*, 86–110. [[CrossRef](#)]
31. Shen, W.; Wang, L.; Hao, Q. Agent-based distributed manufacturing process planning and scheduling: A state-of-the-art survey. *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.* **2006**, *36*, 563–577. [[CrossRef](#)]
32. Arrais-Castro, A.; Varela, M.L.R.; Putnik, G.D.; Ribeiro, R.A.; Machado, J.; Ferreira, L. Collaborative framework for virtual organisation synthesis based on a dynamic multi-criteria decision model. *Int. J. Comput. Integr. Manuf.* **2018**, *31*, 857–868. [[CrossRef](#)]
33. Goren, S.; Sabuncuoglu, I. Robustness and stability measures for scheduling: Single-machine environment. *IIE Trans.* **2008**, *40*, 66–83. [[CrossRef](#)]
34. Ouelhadj, D.; Petrovic, S. A survey of dynamic scheduling in manufacturing systems. *J. Sched.* **2009**, *12*, 417–431. [[CrossRef](#)]
35. Chiu, C.; Yih, Y. A learning-based methodology for dynamic scheduling in distributed manufacturing systems. *Int. J. Prod. Res.* **1995**, *33*, 3217–3232. [[CrossRef](#)]
36. Zhou, R.; Chen, G.; Yang, Z.H.; Zhang, J.B. Distributed manufacturing scheduling using a novel cooperative system. In Proceedings of the 2008 IEEE International Conference on Service Operations and Logistics, and Informatics, Beijing, China, 12–15 October 2008; IEEE: New York, NY, USA, 2008; Volume 1, pp. 256–260.
37. Saeidlou, S.; Saadat, M.; Sharifi, E.A.; Jules, G.D. Agent-based distributed manufacturing scheduling: An ontological approach. *Cogent Eng.* **2019**, *6*, 1565630. [[CrossRef](#)]
38. Zhang, X.; Liu, X.; Tang, S.; Królczyk, G.; Li, Z. Solving Scheduling Problem in a Distributed Manufacturing System Using a Discrete Fruit Fly Optimization Algorithm. *Energies* **2019**, *12*, 3260. [[CrossRef](#)]
39. Wang, C.; Ghenniwa, H.; Shen, W. Real time distributed shop floor scheduling using an agent-based service-oriented architecture. *Int. J. Prod. Res.* **2008**, *46*, 2433–2452. [[CrossRef](#)]
40. Manupati, V.K.; Krishnan, M.G.; Varela, M.L.R.; Machado, J. Telefacturing based distributed manufacturing environment for optimal manufacturing service by enhancing the interoperability in the hubs. *J. Eng.* **2017**, *2017*, 9305989. [[CrossRef](#)]
41. Özgüven, C.; Özbakır, L.; Yavuz, Y. Mathematical models for job-shop scheduling problems with routing and process plan flexibility. *Appl. Math. Model.* **2010**, *34*, 1539–1548. [[CrossRef](#)]
42. Fu, Y.; Wang, H.; Huang, M. Integrated scheduling for a distributed manufacturing system: A stochastic multi-objective model. *Enterp. Inf. Syst.* **2019**, *13*, 557–573. [[CrossRef](#)]
43. Mao, X.; Li, J.; Guo, H.; Wu, X. Research on Collaborative Planning and Symmetric Scheduling for Parallel Shipbuilding Projects in the Open Distributed Manufacturing Environment. *Symmetry* **2020**, *12*, 161. [[CrossRef](#)]
44. Cheng, Y.; Bi, L.; Tao, F.; Ji, P. Hypernetwork-based manufacturing service scheduling for distributed and collaborative manufacturing operations towards smart manufacturing. *J. Intell. Manuf.* **2020**, *31*, 1707–1720. [[CrossRef](#)]
45. Han, X.; Han, Y.Y.; Zhang, B.; Qin, H.X.; Li, J.Q.; Liu, Y.P.; Gong, D.W. An effective iterative greedy algorithm for distributed blocking flowshop scheduling problem with balanced energy costs criterion. *Appl. Soft Comput.* **2022**, *129*, 109502. [[CrossRef](#)]
46. Manupati, V.K.; Chang, P.C.; Tiwari, M.K. Intelligent search techniques for network-based manufacturing systems: Multi-objective formulation and solutions. *Int. J. Comput. Integr.* **2016**, *29*, 850–869. [[CrossRef](#)]
47. Cambou, B.F.; Jain, S. Key Recovery for Content Protection Using Ternary PUFs Designed with Pre-Formed ReRAM. *Appl. Sci.* **2022**, *12*, 1785. [[CrossRef](#)]
48. Westerkamp, M.; Victor, F.; Küpper, A. Tracing manufacturing processes using blockchain-based token compositions. *Digit. Commun. Netw.* **2020**, *6*, 167–176. [[CrossRef](#)]
49. Xu, J.; Tian, Y.; Ma, T.; Al-Nabhan, N. Intelligent manufacturing security model based on improved blockchain. *Math. Biosci. Eng.* **2020**, *17*, 5633–5650. [[CrossRef](#)]
50. Vangala, A.; Das, A.K.; Kumar, N.; Alazab, M. Smart secure sensing for IoT-based agriculture: Blockchain perspective. *IEEE Sens. J.* **2020**, *21*, 17591–17607. [[CrossRef](#)]
51. Kumar, A.; Abhishek, K.; Nerurkar, P.; Ghalib, M.R.; Shankar, A.; Cheng, X. Secure smart contracts for cloud-based manufacturing using Ethereum blockchain. *Trans. Emerg. Telecommun. Technol.* **2022**, *33*, e4129. [[CrossRef](#)]
52. 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](#)]
53. Shahbazi, Z.; Byun, Y.-C. Integration of Blockchain, IoT and Machine Learning for Multistage Quality Control and Enhancing Security in Smart Manufacturing. *Sensors* **2021**, *21*, 1467. [[CrossRef](#)] [[PubMed](#)]
54. Bai, L.; Hu, M.; Liu, M.; Wang, J. BPiIoT: A Light-Weighted Blockchain-Based Platform for Industrial IoT. *IEEE Access* **2019**, *7*, 58381–58393. [[CrossRef](#)]
55. Yu, C.; Jiang, X.; Yu, S.; Yang, C. Blockchain-based shared manufacturing in support of cyber physical systems: Concept, framework, and operation. *Robot. Comput. Manuf.* **2020**, *64*, 101931. [[CrossRef](#)]
56. Shahbazi, Z.; Byun, Y.-C. Improving Transactional Data System Based on an Edge Computing–Blockchain–Machine Learning Integrated Framework. *Processes* **2021**, *9*, 92. [[CrossRef](#)]

57. Leng, J.; Yan, D.; Liu, Q.; Xu, K.; Zhao, J.L.; Shi, R.; Wei, L.; Zhang, D.; Chen, X. ManuChain: Combining Permissioned Blockchain with a Holistic Optimization Model as Bi-Level Intelligence for Smart Manufacturing. *IEEE Trans. Syst. Man Cybern. Syst.* **2019**, *50*, 182–192. [[CrossRef](#)]
58. Sgarbossa, F.; Peron, M.; Fragapane, G. Cloud Material Handling Systems: Conceptual Model and Cloud-Based Scheduling of Handling Activities. In *Scheduling in Industry 4.0 and Cloud Manufacturing*; Springer: Cham, Switzerland, 2020; pp. 87–101. [[CrossRef](#)]
59. Peron, M.; Fragapane, G.; Sgarbossa, F.; Kay, M. Digital Facility Layout Planning. *Sustainability* **2020**, *12*, 3349. [[CrossRef](#)]
60. Szabo, N. Formalizing and Securing Relationships on Public Networks. *First Monday* **1997**, *2*. [[CrossRef](#)]
61. Fairfield, J.A. Smart contracts, Bitcoin bots, and consumer protection. *Wash. Lee L. Rev. Online* **2014**, *71*, 35.
62. Reddy, M.S.; Ratnam, C.; Agrawal, R.; Varela, M.; Sharma, I.; Manupati, V. Investigation of reconfiguration effect on makespan with social network method for flexible job shop scheduling problem. *Comput. Ind. Eng.* **2017**, *110*, 231–241. [[CrossRef](#)]
63. Vafaei, N.; Ribeiro, R.A.; Camarinha-Matos, L.M.; Valera, L.R. Normalization techniques for collaborative networks. *Kybernetes* **2019**, *49*, 1285–1304. [[CrossRef](#)]
64. Sony, M.; Naik, S.S. Ten Lessons for Managers While Implementing Industry 4.0. *IEEE Eng. Manag. Rev.* **2019**, *47*, 45–52. [[CrossRef](#)]
65. Qin, H.-X.; Han, Y.-Y.; Liu, Y.-P.; Li, J.-Q.; Pan, Q.-K.; Han, X. A collaborative iterative greedy algorithm for the scheduling of distributed heterogeneous hybrid flow shop with blocking constraints. *Expert Syst. Appl.* **2022**, *201*, 117256. [[CrossRef](#)]
66. Torabi Yeganeh, F.; Zegordi, S.H. A Multi-Objective Optimization Approach to Project Scheduling with Resiliency Criteria under Uncertain Activity Duration. *Ann. Oper. Res.* **2020**, *285*, 161–196. [[CrossRef](#)]
67. Yang, D.-L.; Kuo, W.-H. Minimizing Makespan in A Two-Machine Flowshop Problem with Processing Time Linearly Dependent on Job Waiting Time. *Sustainability* **2019**, *11*, 6885. [[CrossRef](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.