Advanced Engineering Informatics 52 (2022) 101620

Contents lists available at ScienceDirect

Advanced Engineering Informatics

journal homepage: www.elsevier.com/locate/aei

Full length article

# An innovative approach for resource sharing and scheduling in a sustainable distributed manufacturing system

Veerababu Ramakurthi [a], V.K. Manupati [a], José Machado [b,*], Leonilde Varela [c], Suresh Babu [d]

[a] Department of Mechanical Engineering, NIT Warangal, Warangal, India
[b] Department of Mechanical Engineering, School of Engineering, University of Minho, 4804-533 Guimarães, Portugal
[c] Department of Production and Systems, School of Engineering, University of Minho, 4804-533 Guimarães, Portugal
[d] Department of Computer Science and Engineering, NIT Warangal, Warangal, India

ARTICLE INFO

*Keywords:*
Blockchain technology
Network-based distributed manufacturing systems
Moth flame evolutionary optimization algorithm
Smart contract
Ethereum

ABSTRACT

Secure, transparent, and sustainable distributed manufacturing system (DMS) is a pressing need for current Industry 4.0. In this paper, exchange of highly sensitive information in a more transparent and secure way and to avoid the misunderstandings and trust issues between the enterprises a smart contract based on blockchain technology has been proposed in case of a distributed manufacturing environment. Here, we used a public-permission less Ethereum platform to execute the smart contracts in the Blockchain to process the customer orders and to identify the right enterprise. Later, a multi-objective mixed-integer linear programming (MILP) model is formulated for optimal resource sharing and scheduling in a considered sustainable DMS. The objectives of the proposed model consist of simultaneously improvement of the performance measures such as makespan, machine utilization, energy consumption, and reliability. To solve this MILP model, a new Multi-objective-based Hybridized Moth Flame Evolutionary Optimization Algorithm (HMFEO) is developed and then the effectiveness of the proposed algorithm is validated with the Non-dominated Sorting Genetic Algorithm (NSGA-III). The results obtained from implementing the model using experimental data along with different cases show the efficiency and the validity of the proposed model and solution approach. Moreover, several performance indicators like hyper volume are increased by nearly 15–20 % that shows the superiority of the proposed algorithm with the NSGA-III.

## 1. Introduction

Industry 4.0 radically changed global manufacturing practices to a greater extent by transforming the production process into an optimized cell. This transformation has introduced several key characteristics i.e., connected, optimized, transparent, proactive, and agile [15], to enable the production process to be more effective and efficient. Moreover, this paradigm shift brought several changes in the market such as global competition, security, safety, etc. Out of several manufacturing systems, the distributed manufacturing system (DMS) gains several advantages due to its very nature of integrating several enterprises to fulfil the requirements. In DMS, the enterprises are located at various geographical locations and connected closely thereby fulfilling the need for modern organizational models for small, scalable, flexible, and units to fulfil the customer requirements and enable sustainable manufacturing [43,60].

With the advent of emerging technologies, as well as the importance of individual personalization of customers towards the market [16], and expectation of high response times stipulated the manufacturing systems to turn towards customer-oriented manufacturing. Furthermore, a large amount of information is exchanged between the entities of the DMS. The large volume of data generated in the maintenance of DMS leads to increases in the chances of the vulnerability of data theft [10]. Moreover, the use of external platforms like cloud space [52], increases the challenges in the security aspects of maintaining the manufacturing systems. To attain sustainability in DMS interconnection of manufacturing resources and mutual transfer of product-related information [2], is essential between trust-less manufacturing entities that eliminates the third parties who do not add any value. In addition to several parameters mentioned above for DMS cybersecurity, connectivity, transparency, and trust are the most important performance

* Corresponding author.
*E-mail addresses:* rvbabu.nitw@gmail.com (V. Ramakurthi), manupati.vijay@nitw.ac.in (V.K. Manupati), jmachado@dem.uminho.pt (J. Machado), leonilde@dps.uminho.pt (L. Varela), esbabu@nitw.ac.in (S. Babu).

https://doi.org/10.1016/j.aei.2022.101620
Received 5 January 2022; Received in revised form 29 March 2022; Accepted 18 April 2022
Available online 5 May 2022

measures in the era of Industry 4.0. To attain this, Blockchain Technology (BCT), one of the most strategically important technologies of the 21st century, with its key features like extended visibility, traceability, and disintermediation changed the scenario to trust-based resource sharing and economic activities [17]. More importantly, this technology helps to fulfil the needs of the Digital era [30], particularly for the DMS by making decisions on distributed platforms for peer-to-peer communication.

Additionally, in this article, a DMS environment has been considered to optimize the manufacturing functions i.e., process planning and scheduling requirements. Process planning and scheduling discuss the need for manufacturing resources, operations required to produce a job, and schedule the operations of all the jobs on various machines, while the precedence relationships in the process plans are satisfied [35,45].

This paper seeks to address the following questions.

- How can a blockchain-based methodology be employed to find an efficient way for service decomposition and to exchange the information transparently, securely, and immutably in a distributed manufacturing environment?
- What type of mathematical model can be developed to optimize the conflicting objectives such as completion time, energy consumption, machine utilization rate, reliability subjected to various constraints?
- In which way the proposed approach optimizes the considered objectives by improving the process planning and scheduling functions?

In this paper, smart contracts are used in the blockchain to execute the contractual agreements between peers without any interventions of third parties. It helps in identifying the potential enterprises in the DMS and further helps while allocating the jobs to the machines thereby reducing the complexity in process planning and scheduling. Here, we used the Ethereum platform to execute the smart contracts in the Blockchain to process the customer orders to the right enterprise in the proposed model. The problem is further extended by incorporating the sustainable parameters along with traditional parameters such as energy consumption, reliability, machine utilization, and makespan, to minimize the negative impact of the industrial and ecological system. Hence, this work aims to propose a multi-objective mixed-integer linear programming (MILP) model by consideration of all the objectives along with constraints for effective process planning and scheduling in DMS. To solve this MILP model, an efficient multi-objective hybridized moth flame evolutionary optimization algorithm (HMFEO) is developed and, then the performance of the presented model is compared using NSGA-III algorithm to test the robustness of the designed framework.

The remainder of this paper is organized as follows. Section 2 discusses the literature review. In Section 3, the problem description and the mathematical model are developed. Section 4 explains the proposed framework, algorithm for the blockchain approach. In Section 5 a case study of the gear manufacturing industry in the context of DMS is presented, and the corresponding outcomes are explained in Section 6. Section 7 discussed managerial and academic implications. The paper is concluded in Section 8 by providing scope for future work.

## 2. Literature review

In this section, an extensive literature review is presented to explore the applications of Blockchain Technology (BCT) in the case of DMS and also to give a clear idea of how our proposed methodology is useful in fulfilling the gaps that are identified. Although remarkable work has been carried out in the DMS field, most of these works concentrated on operational and interoperability issues [58,36]. An order matching model [19], has been developed by considering the possibilities of manufacturers and requirements of retailers to fulfil the demand in DMS. This model mainly depends on internet which suffers from several security issues. However, some studies have mentioned a theoretical approach to overcome the challenges associated with the security issues in big data [1] for distributed environments. In order to find by [34], the advancements and overview of state-of-the-art blockchain is presented and discussed its important components such as blockchain-assisted IoT, blockchain-assisted security, blockchain-assisted management of data, and their applications, and its demerits potential trends and challenges. Where, few studies developed by Sharma et al., [50] concentrated on frameworks, architectures, and methods by considering the key functionalities in blockchain i.e., security, and transparency in DMS. In line with this context, a blockchain-based distributed framework has been presented in the automotive industry to trace and track the information across their supply chain. Li et al., [27] proposed a blockchain-based architecture that concentrates on secure data sharing and typical characteristics are discussed and stated the requirements of key technologies for the proposed architecture. Risius et al., [46] introduced a BCT framework along with future prospective research topics to enable meaningful scholarly engagement has been explained. Furthermore, Yang et al., [68] developed a data carrier framework, which emphasizes reducing contract deployment prices and regulates contract events without applying any filter at the Ethereum node. This framework involves mainly three components: ‘Mission Manager’, ‘Task Publisher’, and, a ‘Worker’ who interacts with a smart contract. Viriyasitavat et al., [61] presented a business process management framework that explains the need for incorporation of BCT in the workflow composition and management to verify the trustworthiness of the business partner. Later, Udokwu et al., [59], carried out several studies on blockchain-based smart contracts in the scenario of the business model for enterprises and also explored the opportunities and challenges while implementing the smart contracts. Johng et al., [24] have proposed a BCT approach that explains the trustworthiness of businesses of a firm is processed as text with the help of smart contracts. Chung et al., [13] described a mining procedure to analyze a variety of data associated with a traceability system in cognitive manufacturing. Supranee et al., [53] explained the feasibility of the BCT as a case of the Thai Automotive Industry and a questioner-based survey and regression analysis model was constructed and analyzed. Going one step further Angrish et al., [5], demonstrated a Feb Rec Prototype that can explain the concept of linking computing nodes and, CNC machines to demonstrate the possibility of connecting them on a distributed interoperable network. Furthermore, Westerkamp et al., [64] proposed a Blockchain-based technology for tracking and tracing the products including their evolution in the manufacturing process with help of smart contracts.

Zafar et al., [71] developed a tool that helped to translate 65–70% of solidity smart contracts that are available in Ethereum to the hyper ledger by using JavaScript. The presented tool also helped in reducing the size of code to a greater extent and leads to reduce the memory usage of the blockchain network. Bogner et al., [8] implemented a web application that allows users to register and share the entities based on a smart contract that can deploy on the Ethereum test network. The smart contract allows users to get the rented services with the help of a simple mobile scan without giving sensitive personal information and also no need for a trusted third party. Hasan et al., [20] introduced a blockchain-based smart contract to overcome the threats and limitations with present approaches followed in inventory management and it helps to trace and track the spare part details from manufacturer to the supplier and end-customer. Lohmer et al., [32] presented a concept that is noteworthy that a concept for the usage of blockchain-based smart contracts for sharing of resources in distributed manufacturing networks, moreover, the benefits and risks of the proposed concept are explained. A summary of literature on blockchain based smart contracts and their applications are mentioned in the Table 1.

As mentioned above clearly our considered problem deals with the process planning and scheduling and requires optimization algorithms namely a Branch and price algorithm [37], Simulated annealing [49], and Chaotic particle swarm optimization algorithm [42] in DMS. HS/4/IJMS a hybrid Harmony Search and Genetic Algorithm [7], Ant Colony

**Table 1**
Summary of Literature on various areas of blockchain based smart contracts.

| S. No | Author details | Application area | BC platform | Extent of Application |
|---|---|---|---|---|
| 1. | Bogner et al., [8] | Secure rented services using a mobile apps | Ethereum | Proposed a framework. |
| 2. | Shermin et al., [51] | Governance | Ethereum | Conceptual Explanation |
| 3. | Hou et al., [22] | Power vehicles | NA | Introduced a method |
| 4. | Risius et al., [46], Li et al., [27], | Secure data sharing using Blockchain | NA | Proposed a framework. |
| 5. | Sharma et al., [50] | Automotive Industry in a smart city | BCT based DLT | Proposed a distributed framework |
| 6. | Mondragon et al., [39] | Supply chain of composite materials | BCT based DLT | Proposed a framework |
| 7. | Yeh et al., [69] | Payments using mobile | Ethereum | Introduced a method |
| 8. | Zafar et al., [71] | A tool that translates translate solidity contracts from one platform to other | Ethereum/ hyper ledger | Conceptual Explanation with help of framework. |
| 9. | Schinckus et al., [48] | Secure and transparent data storage | NA | Proposed a framework. |
| 10. | Xu et al., [66] | Manufacturing Supply chain management | Ethereum | A design scheme for sharing of information is proposed |
| 11. | Lohmer et al., [32] | Smart contract for sharing of resources in a DMS | NA | Conceptual Explanation |
| 12. | Wang et al., [62] | Finance | Hyperledger Fabric | Developed a framework |
| 13. | Mohanta et al., [9] | Internet of Things | Ethereum | Developed a framework |
| 14. | Kumar et al., [21] | Cloud Manufacturing | Ethereum based (DLT) | Developed a framework |
| 15. | Lu et al., [34] | IOT, Security, Data sharing in BCT | NA | Presented the State of the art literature |
| 16. | Alkaabi et al., [4] | Additive Manufacturing | Ethereum based smart contract | Implemented or tracing of products in the supply chain. |
| 17. | Panja et al., [41] | Online voting | Ethereum | Implementation |
| 18. | Hasan et al., [20] | Tracing of parts in the manufacturing Supply chain | Ethereum based Smart Contract | Implementation |
| 19. | Leng et al., [26] | Sustainability in blockchain | NA | Presented the State of the art survey. |
| 20. | Zheng et al., [72] | Comparison of smart contract | Ethereum, hyperledger fabric, corda. | Presented the State of the art survey |
| 21. | Wu et al., [65] | Supply chain management in manufacturing | BCT based DLT | Framework of BCT for supply chain integration. |
| 22. | Wang eta al., [63] | Security related smart contract | Ethereum | A Systematic Literature Survey |
| 23. | Kamble et al., [25] | Indian Automobile Industry | NA | Framework of BCT for supply chain integration. |

Optimization (ACO) [31] has been used to optimize the process parameters in the process planning and scheduling in DMS. According to this perspective, existing reviews are analyses in the scheduling of sustainable manufacturing systems to characterize the challenges in achieving it [18]. In more detail, a multi-objective Greedy-based non-dominated sorting genetic algorithm III (GNSGA III) is presented to solve the scheduling problems with Interfering Jobs. Furthermore, several instances were tested for comparison of proposed GNSGA III [12] and Benchmark algorithms under various performance indicators. Zuo et al., [73] proposed an Evolutionary Algorithm (EA) recommendations system architecture is developed for effective selection of EA for solving the scheduling problems in DMS. Liu et al., [30] implemented a multi-stage heuristic method that helps to solve energy-efficient scheduling problem in DMS and the effectiveness of the proposed method is compared with the genetic algorithm. A summary of literature for various methodologies in case of IPPS in context of distributed manufacturing shown in Table 2.

As mentioned above several performance measures were calculated by various solution algorithms in a DMS, still, there is a need of investing the sustainable parameters. In this regard, Leng et al., [26] conducted a survey concerning the manufacturing system where sustainability across product life cycles has been discussed and traced with the blockchain-based approach. In their work, evaluation metrics were developed for the consideration of BCT in the manufacturing sector and then a summary of challenges for achieving sustainable manufacturing has been discussed. Rauch et al., [44] discussed in detail the importance of sustainability parameters in DMS and their reasons and insights for suitability in DMS have been discussed. However, Schinckus et al., [48] have investigated security and transparency are the most important issues in any of the sectors in the current market. An emerging technology literature classification level (ETLCL) framework has been proposed that depends on grounded theory for conducting a Systematic Literature Review (SLR) in various target areas of upcoming technology. It is noteworthy to mention the three BCT approaches Good BCT, Bad BCT, and Ugly BCT suggested. Saber et al., [47] emphasized the requirement of BCT and the increase in the transparency and security of transactions, the BCT also improves the enterprise internal system that leads to profit as well as the sustainability of the system.

The research contributions of the study can be outlined as given below:

- Proposing an integrated Blockchain-assisted process planning framework for distributed manufacturing systems.
- Developing a Blockchain-based smart contract using Ethereum to identify the potential enterprises and sharing of resources

**Table 2**
Summary of Literature on applications of various methodology for IPPS in DMS.

| S. No | Authors | Area of application | Parameters measured | Methodology |
|---|---|---|---|---|
| 1. | Shao et al., [49] | Integrated Process planning and scheduling (IPPS) in DMS | Mean flow time, Makespan | Simulated annealing |
| 2. | Petrović et al., [42] | IPPS in DMS | Makespan, Machine Utilization, Mean flow time | Chaotic particle swarm optimization algorithm |
| 3. | Liu et al., [30] | IPPS in DMS | Energy Consumption, Total tardiness | Heuristic-based two-stage approach |
| 4. | Liu et al., [31] | IPPS in DMS | Makespan | ACO |
| 5. | Manupati et al., [36] [zs.1] | IPPS in DMS | Makespan, Machine Utilization | Mobile agent based method. |
| 6. | Menezes et al., [37] | IPPS in DMS | Operational cost | Branch-and-price algorithm |
| 7. | Cheng et al., [12] | Scheduling problems with Interfering Jobs | Makespan | GNSGA III |
| 8. | Zuo et al., [73] | Scheduling | Makespan | Evolutionary algorithm. |
| 9. | Lohmer et al., [33] | Process planning and scheduling | NA | Systematic Literature Survey |

transparently and securely across the network and tested its feasibility with various real-life cases.

- A multi-objective evolutionary algorithm-based Hybridized Moth Flame Evolutionary Optimization Algorithm (HMFEO) is used to solve the considered problem in the scenario of distributed gear manufacturing industries.
- The results of the proposed HMFEO method is validated with the Non-Dominated Sorting Genetic Algorithm (NSGA-III) to evaluate its usefulness.
- Several performance indicators were tested for both the proposed HMFEO and NSGA- III for the evaluation of effectiveness of the algorithms purpose.

Security, transparency aspects of distributed manufacturing system has been addressed to some extent i.e. Few authors conducted literature survey, some proposed smart contract blockchain-based frameworks and architectures for manufacturing and supply chains. Very little work was carried out on smart contracts for tracking and tracing their products in their supply chains. A clear gap is identified in resource sharing and scheduling with the help of Ethereum based smart contract implementation in the case of distributed manufacturing. In addition, parameters that are considered while Process planning and scheduling lead to improving the sustainability of the distributed manufacturing system which confirms the novelty of this research area. To the best of the authors' knowledge, the first work that has been proposed and implemented in DMS addresses the issues of Security, Transparency, and Sustainability parameters combined.

## 3. Problem description

This paper presents a multi-objective MILP model for a sustainable distributed manufacturing system (DMS) shown in Fig. 1 to optimize process planning and scheduling. This process generally begins when customers request a product from a distributed manufacturing environment (DME) and it proceeds with identifying the potential enterprises that are capable of fulfilling the services required to manufacture the product's out of all the available enterprises. In particular, a customer's order consists of various products known as jobs (n) and each individual job can be manufactured using a series of process plans for better utilization of available resources that leads to efficient scheduling. DMS has the flexibility to utilize alternative process plans to produce the products. Moreover, every process plan demands several sequential operations that need to be performed on various machines. It is noteworthy that the same operation can be performed on different machines of the same type. Hence for a particular job, more than one process plan with a series of operations is available, and thus selecting the process plan with a sequence of operations that gives the best schedule poses a challenge of a computationally complex optimization problem.

In this problem, we have considered a real-life case with thirty-six manufacturing industries established at various places to carry out the operations on the related products. For performing the experimentation, the main parts of the gearbox are considered and it is shown in Fig. 2, which consists of gear, shaft, coupling flanges, a key shaft, pinion, ball bearing, crown wheel which are considered for further analysis. In the proposed model, apart from traditional objective functions such as minimizing the makespan, and maximizing machine utilization, additional objective like minimization of energy consumption and enhancement [3;67,55,56], are considered which adds sustainability aspects to the DMS while carrying out the experimentation.

This work primarily emphasis on the exchange of highly sensitive information for the production process more securely and promotes transparency between the enterprises to select of potential enterprise to avoid misunderstanding and eliminate biasness in a DME by employing the Blockchain technology. The above-mentioned scenario states that the process planning and scheduling in the DME are computationally complex and cannot be solved with classical optimization techniques. Notably, the problem nature is NP-hard, and to solve this with an effective method is required. To achieve this, we propose an evolutionary multi-objective Hybridized Moth Flame Evolutionary Optimization (HMFEO) Algorithm to optimize the process planning and scheduling. Here transparency, security, and tamper-proofing have been achieved through BCT, and optimized performance measures are achieved through HMFEO. To fulfill the above-stated problem and its objective functions, a mathematical model has been developed and

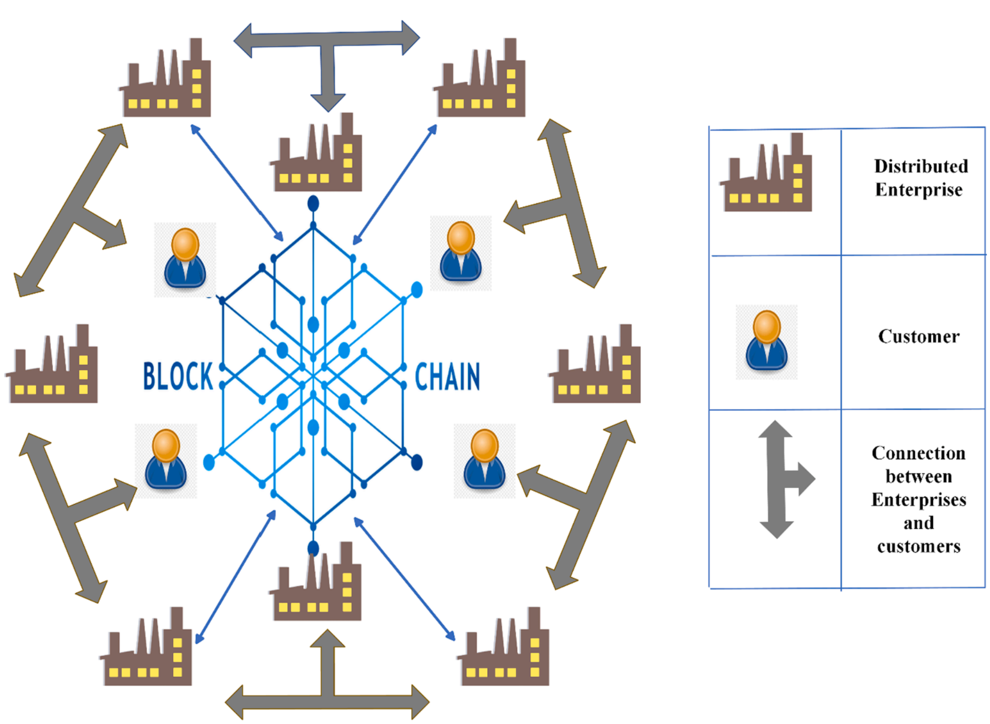


**Fig. 1.** Proposed blockchain based Distributed Manufacturing Systems (DMS).

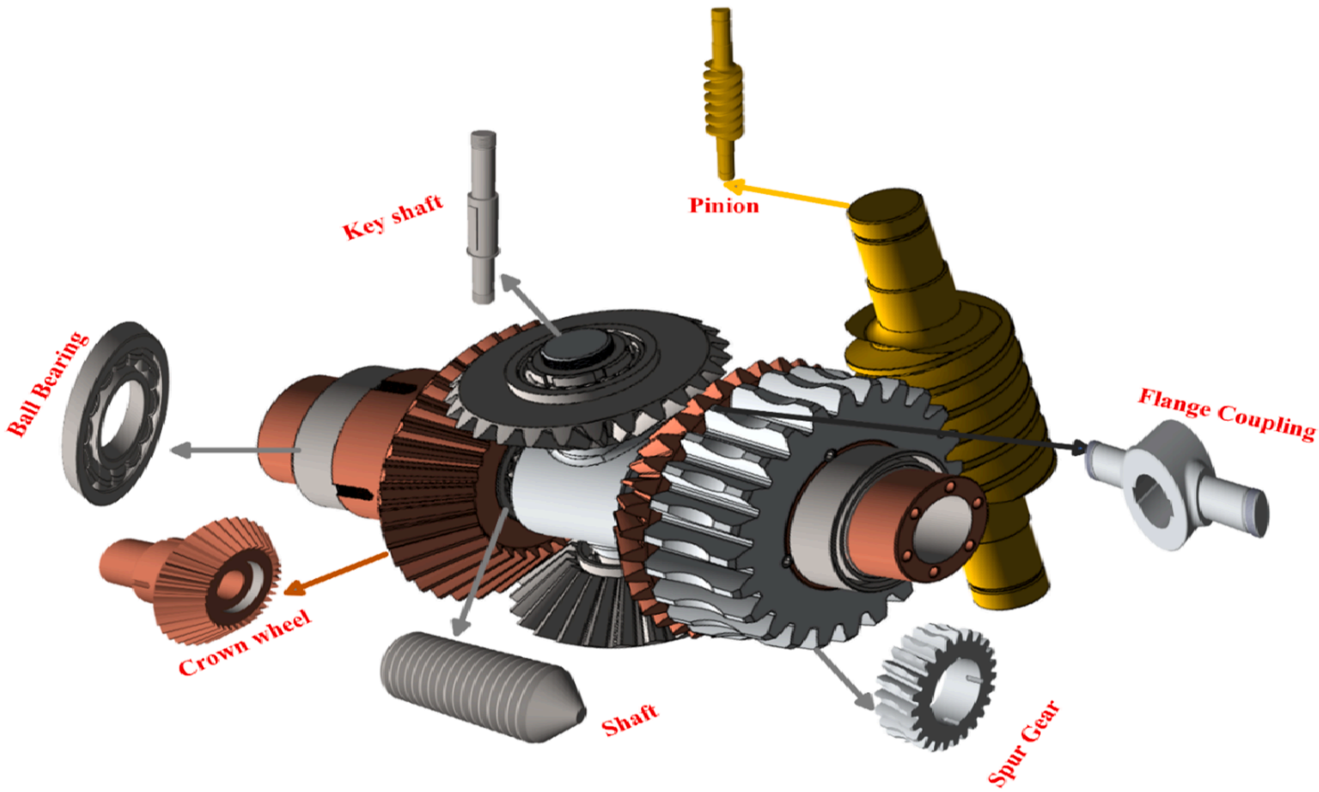


**Fig. 2.** Various parts used in differential gearbox assembly.

notations of the mathematical model are shown in Table 3. While developing the model few assumptions and constraints are considered that are stated below:

Job Pre-emption is not allowed.

Before a new job is processed the preceding job must be completed.
At an instant processing of more than one job is not possible on the same machine.
The reliability aspects of the enterprise are considered as a characteristic of the enterprise and it will not change for all the operations and process plans.
All the machines are assumed to be ready always.
The operations of all the jobs and their sequence must contain future tasks that need to be defined earlier.

**Table 3**
Presents the notations used in the mathematical model.

| Notation | Description |
|---|---|
| *Indices* | |
| $c$ | Index of jobs |
| $n$ | Index of alternative process plans |
| $a$ | Index of operations |
| $q$ | Index of machines |
| *Parameters* | |
| $F$ | Total number of jobs available |
| $R$ | Total number of machines available |
| $T_c$ | The number of alternative process plans of job $c$. |
| $Q_{cna}$ | $n^{th}$ alternative process plan for $a^{th}$ operation of job c |
| $E_{cn}$ | The whole number of operations in the $n^{th}$ alternative process plan of the job $c$ |
| $W$ | Maximum completion time of $c^{th}$ job from the all the total process plans |
| $H_{cnaq}$ | For operation $Q_{cna}$ corresponding processing time of the on machine $q$ |
| $O$ | An arbitrary large positive Integer. |
| $H_c$ | The completion time for processing of job $c$ |
| $D_{cnaq}$ | The earliest completion time till the operation $Q_{cna}$ on machine $q$ |
| $G_{cna}$ | Required energy consumption for $a^{th}$ operation of job $c$ on machine $q$ |
| *Decision Variables* | |
| $X_{cn}$ | 1 The *nth* alternative process plan of job $c$ is considered<br>0 Except above condition |
| $Y_{candboq}$ | 1 The operation $Q_{can}$ precedes the operation $Q_{dbo}$ on given machine $q$<br>0 Except above condition |
| $Z_{canq}$ | 1 If given machine $q$ is selected for $Q_{can}$<br>0 Except above condition |

*Objectives:*

$$\text{Minimization of makespan } (W_{min}) = Max\ D_{canq} \tag{1}$$

$$\text{Maximization of Machine Utilization } (U_c) = \frac{\sum_{q=1}^{R} H_{cq}}{\sum_{q=1}^{R} (mct_q - mst_q)} \tag{2}$$

$$\text{Minimization of energy consumption } (G_{min}) = \sum_{c=1}^{F} \sum_{a=1}^{E_{cn}} \sum_{q=1}^{R} G_{caq} \tag{3}$$

where $H_{cq}$ represents processing time of job c on the $q^{th}$ machine, and $mct_q$ indicates finishing time of $q^{th}$ machine i.e. the time taken to finish the final operation on $q^{th}$ machine. $Mst_q$ is the start time of $q^{th}$ machine.

*Subject to Constraints:*

The initial operation (a = 1) in the possible process plan $n$ of job $c$ is mentioned as.

$$D_{cn1q} + O(1 - X_{cn}) \geqslant H_{cn1q}$$
$$c \in [1, F],\ n \in [1, T_C], q \in [1, R] \tag{4}$$

The final operation for the possible process plan $n$ of job $c$ is mentioned below.

$$D_{cnE_{cn}q} - O(1 - X_{cn}) \leqslant H_{cnaq}$$
$$c \in [1, F],\ n \in [1, T_c], q \in [1, R] \tag{5}$$

Different operations for a same job having precedence constraints are unable to be processed simultaneously.

$$D_{cnaq} - D_{cn(a-1)q_1} + O(1 - X_{cn}) \geqslant H_{cnaq}$$
$$c \in [1,F],\ n \in [1,T_c], a \in [1,E_{cn}], q, q_1 \in [1,R] \tag{6}$$

Every machine can able to process only one operation at a time and expressed as.

$$D_{cnaq} - D_{dobq} + OY_{cnadobq} \geqslant H_{cnaq}$$
$$c,d \in [1,F],\ n,o \in [1,T_c], a,b \in [1,E_{cn}], q \in [1,R] \tag{7}$$

The aforementioned objectives, that is, minimization of makespan, minimization of total training cost of workers, minimization of energy consumption, maximization of service utilization is given by Equation (1) - (3) respectively. Equation (4) -(7) detailed the constraint related to process plans and precedence relationship of operations.

## 4. Blockchain framework for planning and scheduling in a DMS

In order to respond to the aforementioned challenges and issues, there is a need for an efficient approach that can fulfill the requirements of highly secure, trust, and sustainable parameters simultaneously. Hence, we proposed an integrated blockchain-assisted evolutionary algorithmic approach and a framework is developed for optimal allocation of resources and scheduling in a DMS. The proposed framework is shown in Fig. 3, which mainly contains three parts namely, a service

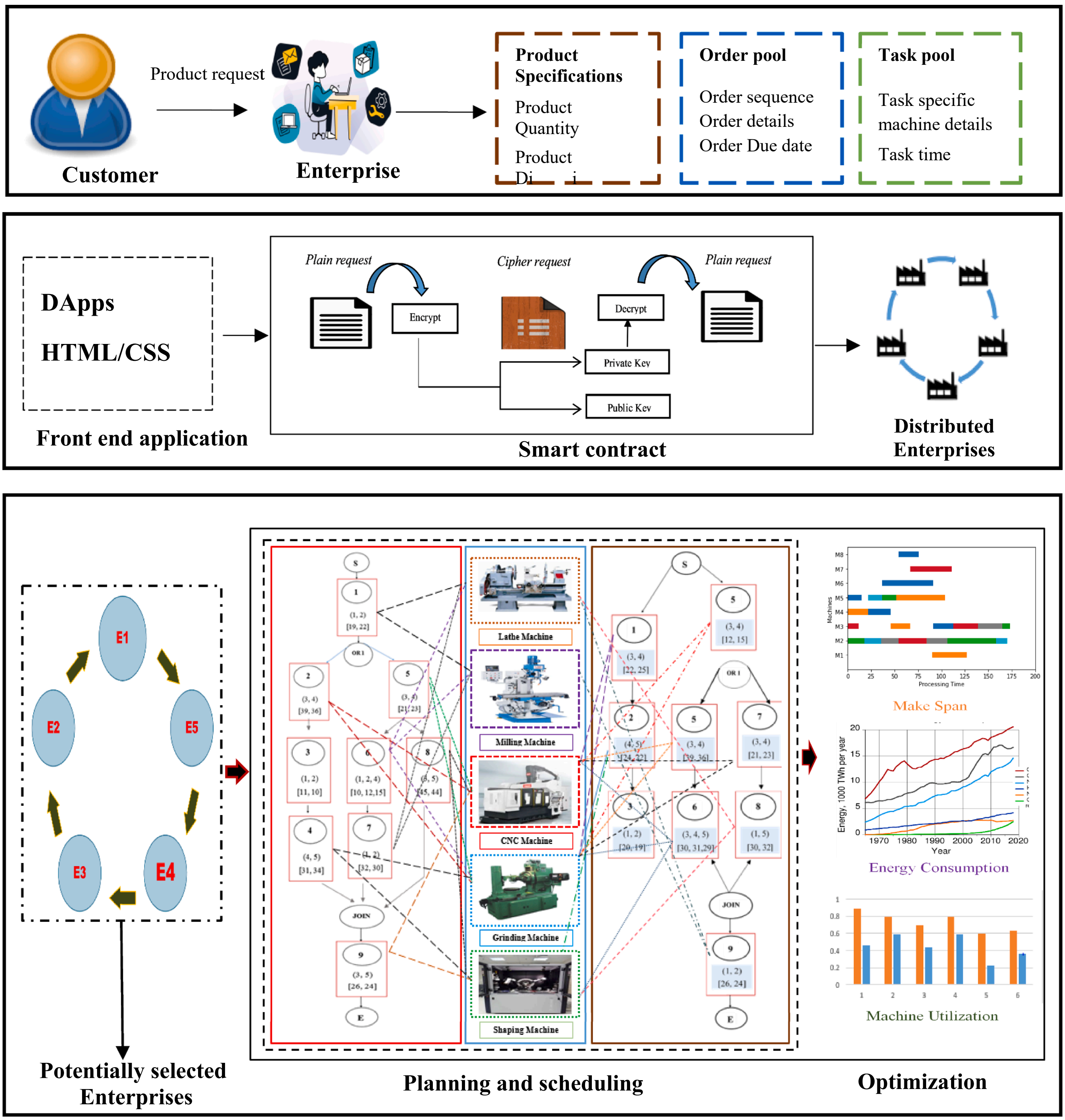


Fig. 3. Framework of blockchain-assisted evolutionary algorithm approach.

layer, a product operational blockchain layer, and a planning layer. In the following sub-sections, we clearly describe the significance of each layer.

### 4.1. Service assistance layer

In this layer, the customer request for a product based on his requirements by mentioning the basic details of the product through a front end application such as a web application generally written in JavaScript, HTML, and CSS language that allows the customers to specify their requirements in the Ethereum platform, which in turn interacts with the blockchain. Immediately the customer product request is received by the Enterprise User (EU). EU have the capability to decompose the orders by their technical specifications such as product specifications (product quantity, product details), order specifications (Order sequence, order details, and order due dates), and Task specifications (Machine available details, task details). Moreover, upon further analysis, the EU list out the various manufacturing services that are required to manufacture that product. For example, if the product request is to manufacture a gearbox then enterprise users list out the services that are required namely grinding, milling, drilling, boring etc.

After identification of services, the EU requests the available enterprises in DMS to respond based on the enterprise ability or interest to fulfil that particular service. The considered DMS environment consists of enterprises that are small and medium scale industries may or may not have the capability to offer all the services one at a time. Hence upon request of the service from the EU, the enterprises verify its offering services and if any service is matched they will respond.

### 4.2. Operational blockchain layer

In this layer, the data related to product requests take place between customers and enterprises in a blockchain structure. This layer is one of the important and interesting parts of the mentioned framework that differs from other manufacturing frameworks. The philosophy behind this layer is to eliminate the concept of a trusted third party thereby developing a trust-less environment between the enterprises. To fulfil this the shared operational resources are stored in the immutable blockchain and it will be further used for tracing the consignment and to maintain privacies of entities associated with it. At the same time, a logic code consists of a sequence of instructions executed by a smart contract thereby ensuring control over the data that is transmitted into the blockchain. The predominant achievement of having security and transparency among all the available entities to identify the right resources out of many is possible and a successful code is implemented with the help of block chain based smart contract.

### 4.3. Planning layer

After completion of all the instructions, the transmitted data in the blockchain send to the Planning layer where all the orders are requested by various customers are stored in the order pool. Each order requires several operations called tasks, where each task can be completed with different machine services offered by various enterprises. Based on the efficiency, reliability of the network, service time, service cost, logistics cost, processing cost, etc. the right enterprise is selected. Tasks are performed with various optimization techniques to be scheduled and sequenced amongst the enterprises according to the preferences and requirements of the product. The procedure continues until all the orders are complete.

The proposed framework typically explains all the aspects that start with a customized order request to the enterprise where customers can request for customized product. Simultaneously the information sharing between the various entities securely and transparently using blockchain technology in DMS is represented in the framework. Finally, a path to solve the problem with desired objectives is represented in the framework.

## 5. Blockchain technology

Blockchain Technology is a linked list of chains that allows data to store and exchange in a transparent manner by confidentially maintaining the transactions on a peer-to-peer basis. Structurally, blockchain data can be consulted, shared, and secured where transactions are authenticated through cryptographic techniques. Here, the transactions are determined with a ledger that allows participants to verify without third-party interference. Therefore, BCT ensures tamper-proof, transparent, and trustable transactions among peers.

### 5.1. Blockchain based smart contracts applied to distributed manufacturing systems (DMS)

A smart contract is logic or code that was written in a computer language that can mix the user interface features with the computer net protocols to perform the contractual terms. It executes the code and performs the tasks in the distributed ledger even though its legal justification was not defined properly. In this paper, the considered distributed environment is more suitable for a public permission-less network where the enterprises that interact in the blockchain are not required any special permissions to participate in the BC network. Ethereum is an open-access software through which anybody can participate in the network under a public permissionless blockchain network and whoever participating groups are often called nodes, being based on smart contract's nodes work in the blockchain. The information stored in the Ethereum public blockchain is based on smart contracts and all the nodes can see the data in it. Fig. 1 shows the blockchain model for DMS. All the participants must possess an account that has a specific address that is nothing but the user public key.

### 5.2. Proposed blockchain model

In this case, we consider a distributed manufacturing system consists of ten enterprises that are distributed at various locations. An enterprise is having the capability to manufacture products depending on the order received from various customers. Enterprise user takes the responsibility of fulfilling orders that were requested by various customers. It acts as a bridge or a king of brokers in the context of virtual enterprises, linking between customers and enterprises. Initially, various customers request products, and enterprise users receive all the requests. Each product request must be fulfilled with the help of various manufacturing services that are offered by various enterprises. The idea behind our model is to implement the process of placing an order and matching of order with the suitable enterprise that is capable of fulfilling the order as smart contracts and place the smart contracts on a blockchain-enabled basis in the distributed platform, for both the execution of the contracts and storage of results. Primary entities in our model are the customer, enterprise user, and the manufacturing companies that offer services. The customer places the order, this order is taken by the enterprise user and is transferred to the companies to check if they can fulfill the order.

#### 5.2.1. Implementation of the proposed blockchain model in Ethereum

The proposed BC model includes two smart contracts. The first smart contract in Table 4 is between the customer and enterprise user. This smart contract automates the process of placing an order, this order is placed and stored for further processing. The second contract is in Table 5, between the enterprise user and the companies, through which the placed orders are now matched with the companies that provide necessary services to fulfil the order. Fig. 4 denotes the proposed model for working with smart contracts in the BC.

Table 4 shows the first smart contract is executed between the customer and enterprise user that automates the functionalities like the placing of the order along with specifications like Product Name, Col-

**Table 4**
Pseudo code for the smart contract between customer and enterprise user.

**Algorithm 1: Order Creation**

Input: *name, quantity, color, expected date*
1. Initialize Integer Order Sequence to 0 and a Mapping from integer to order structure called orders.
2. O ← (name, quantity, color, expected date)
3. Create the order structure O and store it
4. orders[Order Sequence++] ← O (storing the order)

**Algorithm 2: Query Order**
Input: *Order_id (Integer)*
Output: Order Object
1. O ← orders [Order_id]
2. Return O

**Table 5**
Pseudo code for the smart contract between customer and enterprise user.

**Algorithm 1: Enterprise creation**

Input: *name,list of services offered by the enterprise* (Bool Type)

1. E ← (name, list of services offered, address) // The enterprise are assigned Ethereum addresses
2. Create the enterprise structure E and store it

**Algorithm 2: Product creation**
Input: *name, list of services required to build the product (Bool Type)*
3. P ← (name, list of services required)
4. Create the product structure P and store it
mapped to the particular product so that the enterprises understand the processes required to build the product and fulfil the order.

**Algorithm 3: Checking if an enterprise can ACCEPT, PARTIALLY ACCEPT, or REJECT the order**
Input: Order id
Output: ACCEPT, PARTIALLY ACCEPT or REJECT
1. Initialise integer oc and ec to 0.
2. Processes required to complete order and processes/services offered by enterprise is checked against each other (since both are bool values)
3. If an order requires a process × and the enterprise offers that service ×, increment ec and oc, else increment oc only
4. if(oc==ec)
5. return "ACCEPT";
6. else if (ec > 0 && ec < oc)
7. return "PARTIALLY ACCEPT";
8. if(ec==0)
9. return "REJECT";

our, Quantity. Each order is mapped to a unique ID for easy access. Later querying of a placed order is possible by using the unique ID, which enables to access information like the total number of placed orders.

A second smart contract is executed between the enterprise user and the Enterprises that help to fulfill the orders which are stored using the first smart contract and automate the functionalities like the addition of new enterprises along with the services they offer. Here, with ten enterprises namely Enterprise 1 (E1), Enterprise 2 (E2), Enterprise 3 (E3), Enterprise 4 (E4), Enterprise 5 (E5), Enterprise 6 (E6), Enterprise 7 (E7), Enterprise 8 (E8), Enterprise 9 (E9), Enterprise 10 (E10). Let us consider below-mentioned Services offered by all the enterprise's Viz., Designing (DE), Drilling (Dr), Machining (M), Boring (Bo), Ream-ing (Re), Welding (W), Broaching (Br), Milling (Mi), Hobbing (Ho), Grinding (Gr), Shaping (Sh), Turning (Tu), Finishing (Fi), Electric discharge machining (EDM). To manufacture the gearbox it requires services like Designing (DE), Machining (M), Welding (W), Milling (Mi), Hobbing (Ho), Turning (Tu), Shaping (Sh). According to the proposed BC model enterprise, the user sends a request to all the enterprises in the BC, and based on the availability of services they need to respond. For testing, each enterprise is assigned with particular Ethereum addresses and Fig. 5 indicates the various addresses that are created for Enterprise 1 and Enterprise 2.

To respond each enterprise can access the Ethereum network and automatically check if they can ACCEPT, REJECT or PARTIALLY ACCEPT the order based on the services they offer and the services required for building the product(order). In the implementation for further understanding and simplicity, fixation of the services offered by each enterprise has been done. We have checked the enterprise user request for all the ten enterprises and each case has been explained in detail in the below section (i.e. case I to case X) and the corresponding transaction result is also shown in the below-mentioned figures.

Case (I): Required services specified by Enterprise User are (DE, M, W, MI, Ho, Tu and, Sh). The services offered by E1 are (DE, Dr and, B). Immediately the smart contract check for the matching service and the transaction will result in a Partially Accept order because it has only some of the services (i.e. only DE) are available to perform the task. And the corresponding result is mentioned in Fig. 6. In the figure, it was mentioned clearly to explain the transaction status of smart contracts between the enterprise user and enterprise 1, and corresponding hash values are assigned for their addresses. In this way, the transaction is cryptographically encrypted and a logic written in the smart contract is executed based on the available services and required services. Finally, the result is displayed as Accept the order highlighted in Fig. 6.

Case (II): Required services specified by Enterprise User are (DE, M, W, MI, Ho, Tu and, Sh) and Services offered by E2 are (DE, M, W, MI, Ho, Tu and, Sh). Immediately the smart contract checks for the matching service and the transaction will result in Accept the order because it has all of the services (i.e. DE, M, W, MI, Ho, Tu and, Sh) are available to perform the task. And the corresponding transactions are mentioned in Fig. 7.

Case (III): Required services specified by Enterprise User are (DE, M, W, MI, Ho, Tu and, Sh). The Services offered by E3 are (Gr, Bo, Fi and, EDM). Immediately the smart contract checks for the matching service and the transaction will result in Reject the order because has none of the services available to perform the task. And the corresponding transactions are mentioned in Fig. 8.

Case (IV): Required services specified by Enterprise User are (DE, M, W, MI, Ho, Tu, and Sh). The services offered by E4 are (EDM). Immediately the smart contract checks for the matching service and the transaction will result in Reject the order because it has none of the services available to perform the task. And the corresponding result is mentioned in Fig. 9.

Case (V): Required services specified by Enterprise User are (DE, M, W, MI, Ho, Tu and, Sh) and Services offered by E5 are (Ho, EDM and, Fi). Immediately the smart contract checks for the matching service and the transaction will result in Partially Accept the order because it has some of the services (i.e Ho) available to perform the task. And the corresponding result is mentioned in Fig. 9.

Case (VI): Required services specified by Enterprise User are (DE, M, W, MI, Ho, Tu and, Sh) and Services offered by E6 are (DE, Sh, Fi and, EDM). Immediately the smart contract checks for the matching service and the transaction will result in Partially Accept of the order because it has some of the services (i.e. DE and Sh) available to perform the task. And the corresponding result is mentioned in Fig. 9.

Case (VII): Required services specified by Enterprise User are (DE, M, W, MI, Ho, Tu and, Sh) and Services offered by E7 are (DE, M, W, MI, Ho, Tu and, Sh). Immediately the smart contract checks for the matching service and the transaction will result in Accept the order because it has all the services available to perform the task. And the corresponding result is mentioned in Fig. 9.

Case (VIII): Required services specified by Enterprise User is (DE, M, W, MI, Ho, Tu and, Sh), and Services offered by E8 are (EDM, W and, MI). Immediately the smart contract checks for the matching service and the transaction will result in Partially Accept the order because it has some of the services (W and MI) available to perform the task. And the corresponding result is mentioned in Fig. 9.

Case (IX): Required services specified by Enterprise User are (DE, M, W, MI, Ho, Tu and, Sh) and Services offered by E9 are (Fi and Bo). Immediately the smart contract checks for the matching service and the transaction will result in Reject the order because it has none of the

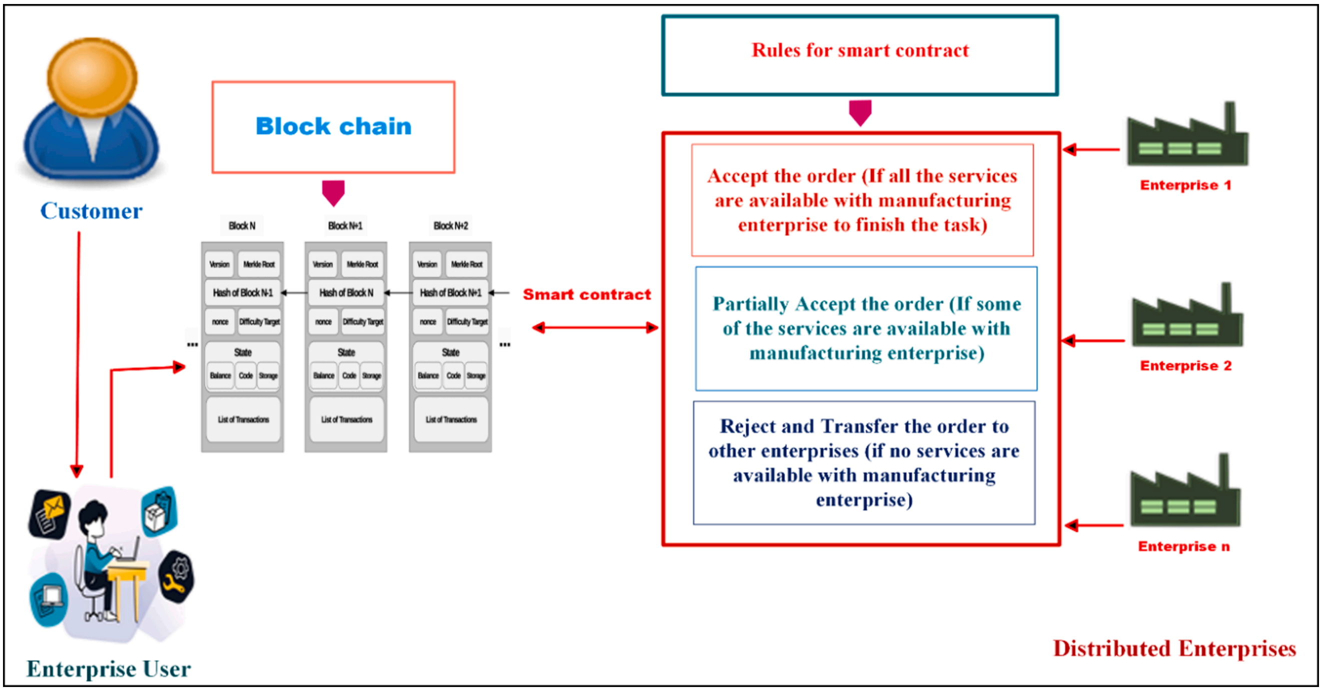


**Fig. 4.** Proposed model diagram for working of smart contract in the blockchain.

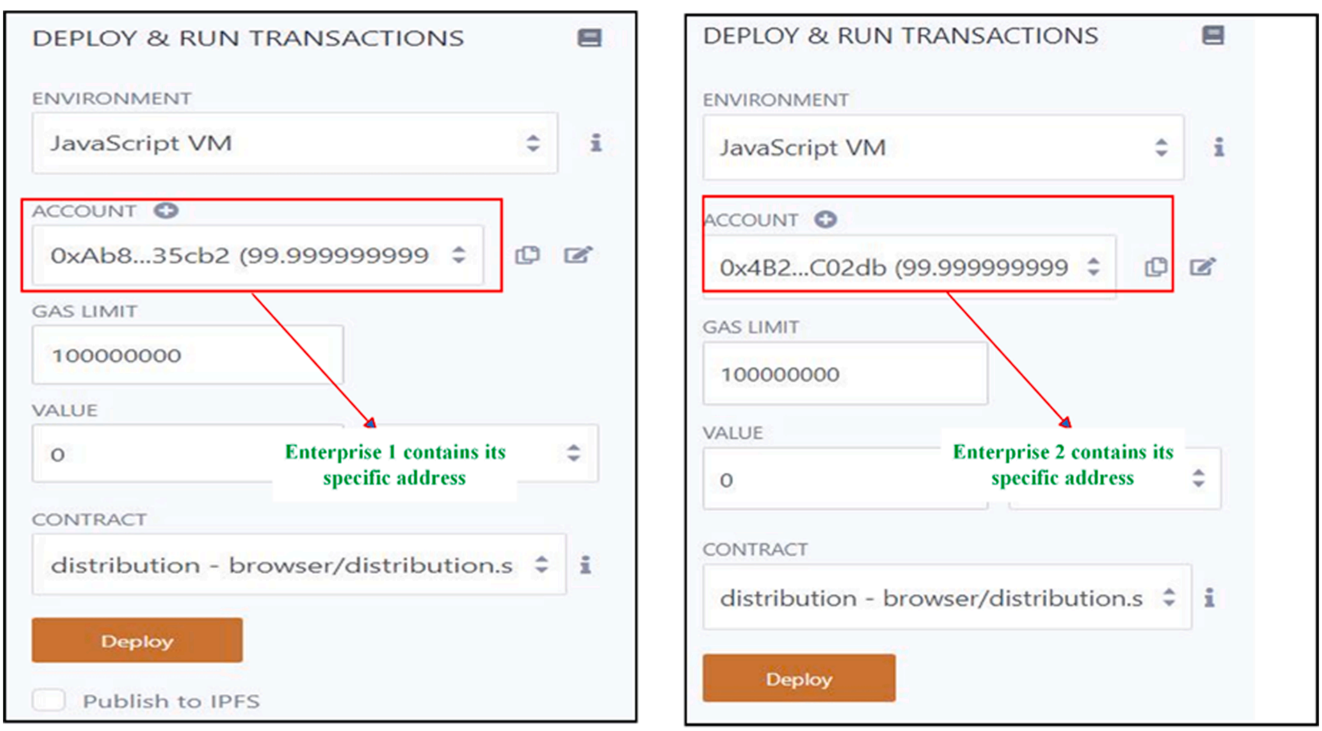


**Fig. 5.** Various addresses are created for Enterprises 1 and 2.

services available to perform the task. And the corresponding result is mentioned in Fig. 9.

Case (X): Required services specified by Enterprise User are (DE, M, W, MI, Ho, Tu and, Sh) and Services offered by E8 are (W, M, EDM and, Fi). Immediately the smart contract checks for the matching service and the transaction will result in Partially Accept the order because it has some of the services (W and M) available to perform the task. And the corresponding result is mentioned in Fig. 9.

The smart contracts were written in solidity which runs on Ethereum. The proposed system comprises the following components that have been implemented in Intel(R) Core (TM) i7-8550U CPU @ 1.80 GHz, 32 GB RAM Ubuntu 18.04 LTS. Remix Ethereum- pragma solidity ˆ0.7.4 is used to run the proposed system. The above results with the help of blockchain-based smart contracts that were run on Ethereum clearly help to identify the suitable enterprises among all the available enterprises in the distributed manufacturing environment. Apart from the smart contract helps to identify the right enterprise that has the capability to offer the services required to manufacturing the product.

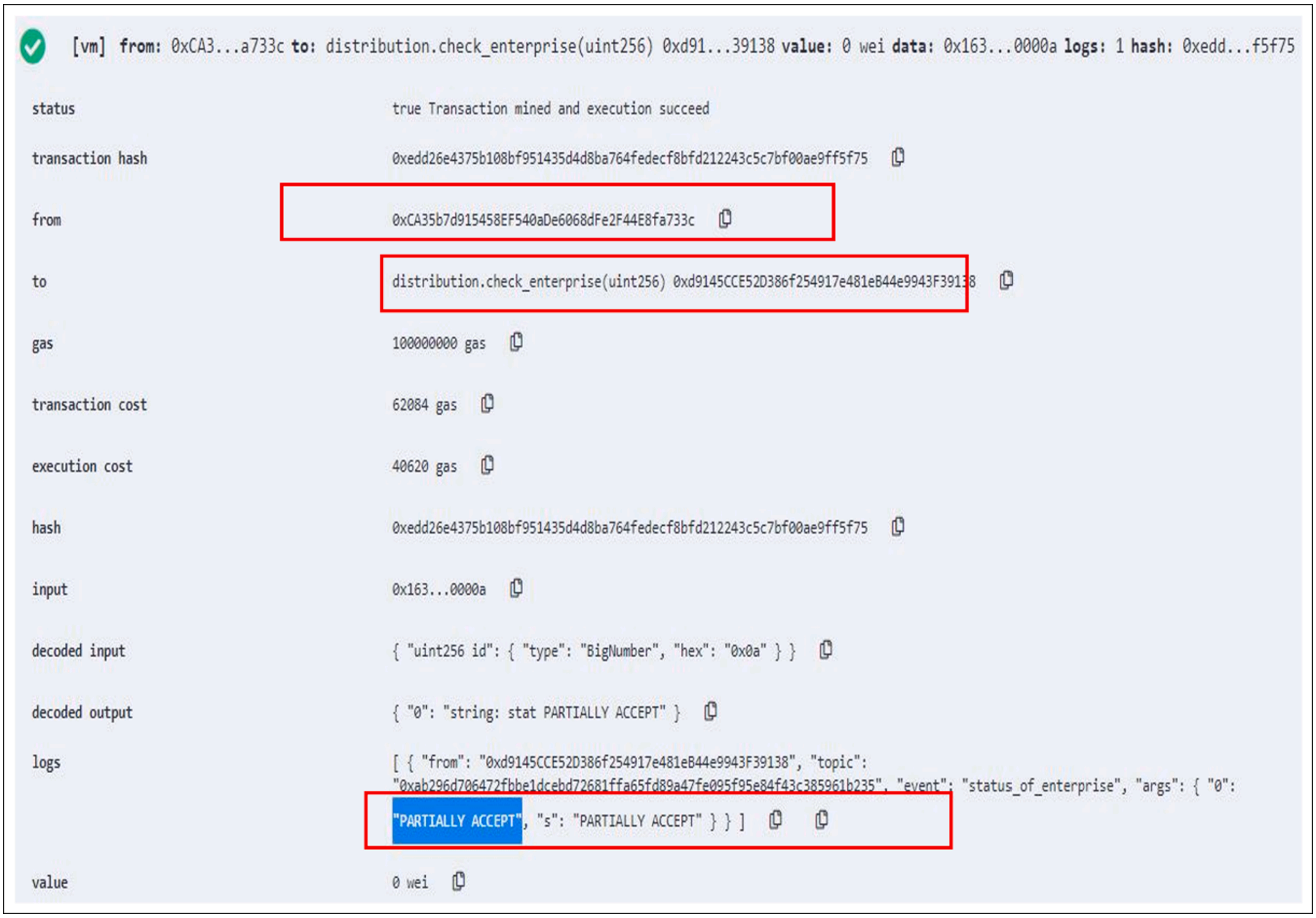


**Fig. 6.** Transactions recorded on the Ethereum based blockchain for case 1- E1.

Moreover, this entire process takes place in a secured and transparent environment with the help of blockchain based smart contract is an added advantage. Once this has been done the potential enterprise's data is transferred to the planning layer shown in the proposed framework shown in Fig. 3. In the planning layer to do the effective process planning and scheduling to achieve the desired sustainable parameters namely makespan and energy consumption, service utilization and reliability for the considered problem. To achieve this a suitable methodology has been employed that is discussed in the next coming sections.

### *5.3. Multi-objective hybridized moth flame evolutionary optimization (HMFEO) algorithms*

To solve the considered problem a newly established Bio-inspired Moth Flame Evolutionary Optimization algorithm (MFEO) has been adopted and further it has been mapped according to problem nature [38]. The superiority of MFOA over other algorithms (GA, PSO, ACO) is clearly shown in his work by conducting tests on several benchmark functions [52]. In this present work, a hybridized form of moth flame evolutionary algorithm (HMFEO) is presented. The operations are assigned to the machines in such a way that the considered objective functions are satisfied and an optimal sequence is obtained. The above-discussed approach is implemented for all formulated instances to find the robustness of the algorithm.

In order to validate the proposed model, several small-sized instances were solved by the CPLEX solver of GAMS software. Later the proposed HMFEO algorithm results were compared with a reference point based multi-objective algorithm NSGA-III is proposed by [29] considered. Proven to be more efficient for solving multi and many objective problems that works with a clustering operator instead of crowding distance operator in NSGA-II.

The parameters for the HMFEO technique are specified for the implementation of algorithms shown in Table 6 with the number of moths as 100 and the maximum number of iterations as 2500. Upper boundary and lower boundary values are specified based on the test data input.

*Step 1*: In HMFEO potential solutions are represented as moths and variables are represented as position in the moth space. A matrix consists of all the moths (n) and their dimension is d.

$$\begin{bmatrix} K_{11} & K_{12} & \cdots & K_{1d} \\ K_{21} & K_{22} & \cdots & K_{2d} \\ K_{31} & K_{32} & \cdots & K_{3d} \\ K_{41} & K_{42} & \cdots & K_{4d} \end{bmatrix}$$

Initialization of moth population and their spaces are defined with the time matrices and their corresponding inputs. In this proposed HMFEO a new type of encoding schema was presented to suit the problem nature. The encoding scheme for makespan is presented in Fig. 10.

To understand the encoding schema in Fig. 10 mainly contains three jobs that are to be processed on three machines and, each machine demands three operations. Furthermore, specifically observe the Fig. 10 a) encoding explains the number of operations and their sequence to be followed for each job i.e. job1, job2 and, job3 whereas Fig. 10 b) detailed the particular operations and their corresponding available machine for processing. Similarly, in Fig. 10 c) explains the encoding schema for processing time for corresponding machine for a given operation. The sequence of processing can be represented as Where $W_{31}^{2}$ is the 2nd operation of the third job will be processed on the first machine. The makespan(:,:,x, y) matrix shows the processing time of machines for the

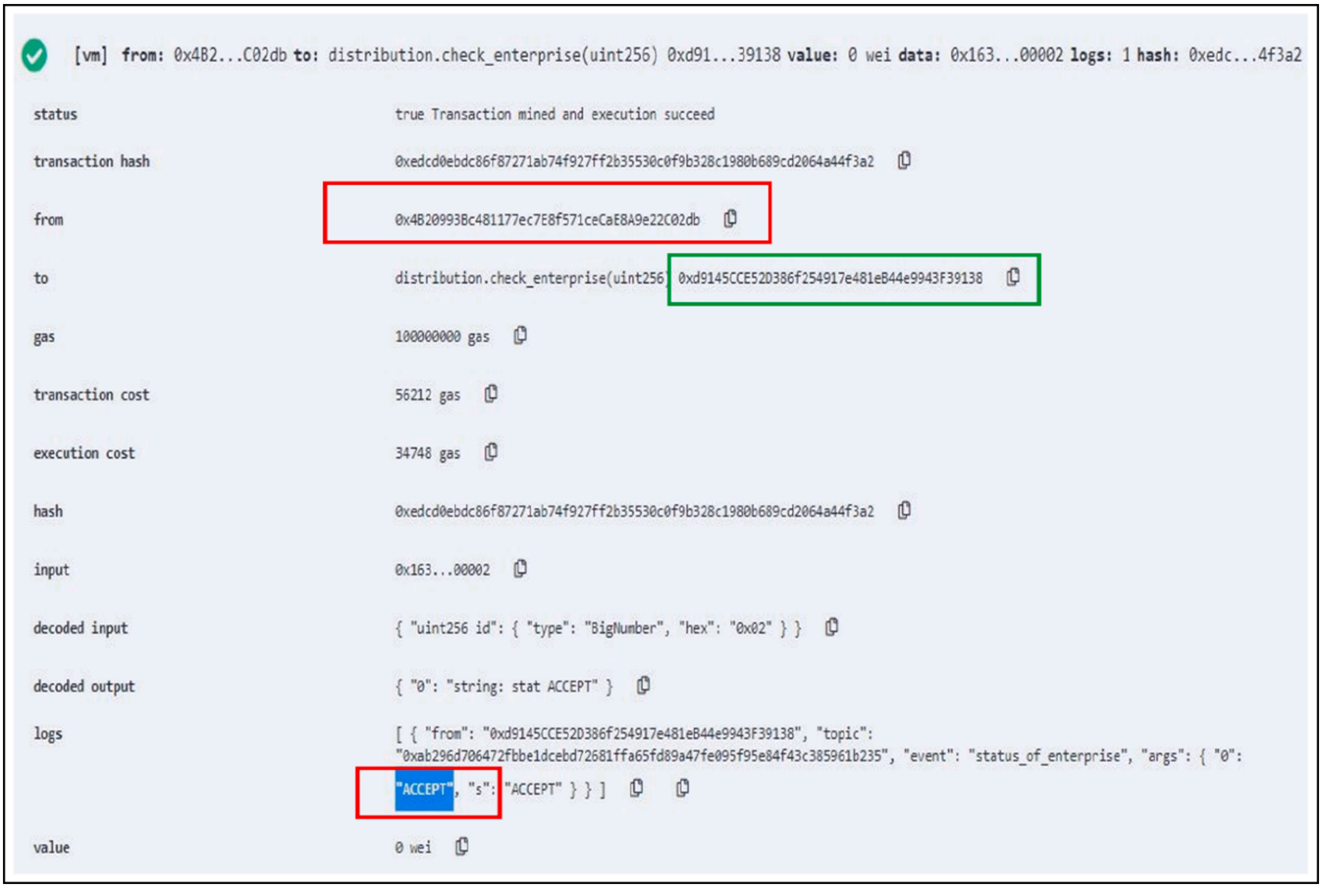


**Fig. 7.** Transactions recorded on the Ethereum based blockchain for case 2- E2.


**Fig. 8.** Transactions recorded on the Ethereum based blockchain for case 3- E3.

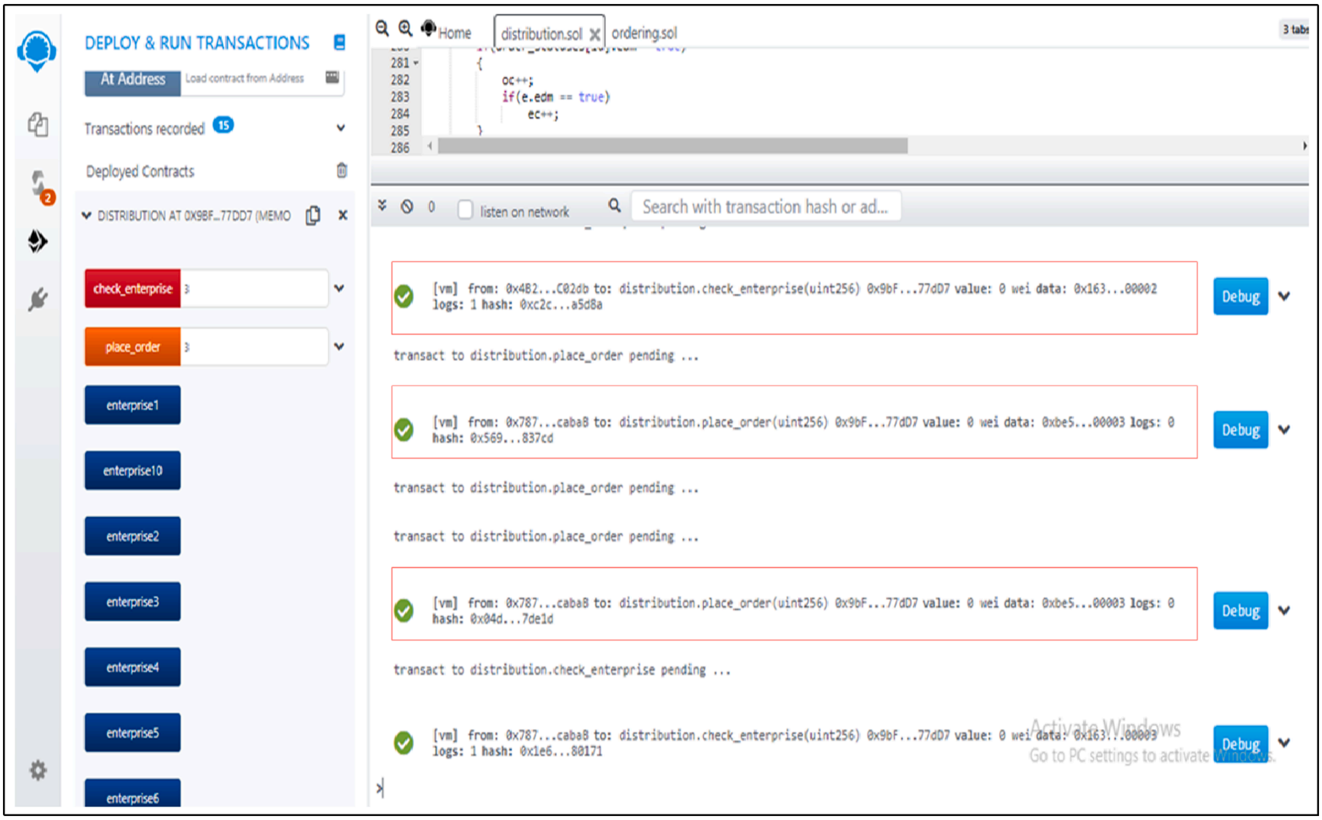


**Fig. 9.** Screen shot of transactions recorded on the Ethereum blockchain for various cases.

**Table 6**
Initialization of parameters for proposed solution algorithm.

| Process Parameters | HMFEO | NSGA III |
|---|---|---|
| Population Size/No of Moths | 100 | 100 |
| Number of generations | 2500 | 2500 |
| Mutation Probability | – | 0.062 |
| Cross Over Probability | – | 0.74 |
| No of Reference points | – | [90,200] |
| Cross Distribution Index | – | 20 |
| Cross over Operator | – | Simulated Binary crossover |
| Lower bound, upper bound | [0, ∞] | – |
| Number of objectives | 3 | 3 |

particular operation for the $x^{th}$ job and $y^{th}$ process plan. The remaining values in the matrix are kept as zeros.

**M1 M2 M3 M4 M5 M6 M7 M8 M9 M10 M11 M12**
**O1** [ 6 16 7 0 0 0 0 0 0 0 0 0;
**O2** 14 8 25 0 0 0 0 0 0 0 0 0;
**O3** 11 13 16 0 0 0 0 0 0 0 0 0;
**O4** 0 0 0 0 0 0 0 0 0 0 0 0;
**O5** 0 0 0 0 0 0 0 0 0 0 0 0];

*Step 2***:** Similarly, upon multiplying with the time matrix with the related corresponding energy rating shown in Table 7 leads to the energy consumption matrix (Equation (11)). The Fig. 11 indicates the corresponding encoding schema.

Energy consumption matrix (o, m1, p1, j) = Makespan matrix (o, m, p, j) *Energy (Rated energy matrix); (11).

To understand the encoding schema in Fig. 11, upon careful observation Fig. 11 a) encoding explains the number of operations and their sequence to be followed for each job i.e. job1, job2 and, job3 whereas Fig. 11 b) detailed the particular operations and their corresponding available machine for processing. Similarly, in Fig. 11 c) explains the encoding schema for energy consumption of corresponding machine for a given operation.

**M1 M2 M3 M4 M5 M6 M7 M8 M9 M10 M11 M12**
**O1** [ 90 464 224 0 0 0 0 0 0 0 0 0;
**O2** 210 232 800 0 0 0 0 0 0 0 0 0;
**O3** 165 377 192 0 0 0 0 0 0 0 0 0;
**O4** 0 0 0 0 0 0 0 0 0 0 0 0;
**O5** 0 0 0 0 0 0 0 0 0 0 0 0];

The above matrix indicating the energy consumption values of machines for the corresponding operation for the 3rd job and $2^{nd}$process plan. The remaining values in the matrix are kept as zeros.

*Step 3* The selection of process plan out of available process plans is carried by score function shown in Equation (12), a lower score value leads to a selection of better process plan.

$$\text{Score} = \frac{\text{Makespan X Energy consumption}}{\text{Reliability}} \tag{12}$$

*Step 4* To solve each objective function a matrix k is formed by considering all the moths that are stored in FK represented below.

$\text{FK} = [FK_1 \quad FK_2 \quad FK_3 \quad FK_4\ ]^T$.

Later a flame matrix (L) to store the fitness value is taken in to consideration that is of same size with that of moth matrix (K).

*Step 5* Once after the process of selecting a suitable process plan; finding of minimum values has been carried out by considering rows as individual light sources and their exploration in their respective rows for minimum entry once the required inputs are received and search space is clearly initialized.

*Step 6* Moths updates its position through a process of moving around the flag dropped by them while searching follows a spiral motion represented in Equation (13).

$$Z(\text{Kx}, \text{Ly}) = \text{Sx} \times \text{eat} \cos(2\pi t) + \text{Ly} \tag{13}$$

$K_x$ indicates the $x^{th}$ moth, $L_y$ indicates the $y^{th}$ flame and Z indicates spiral function. $S_x$ is the distance of $x^{th}$ moth for $y^{th}$ flame, $S_x = \| L_y - K_x \|$,

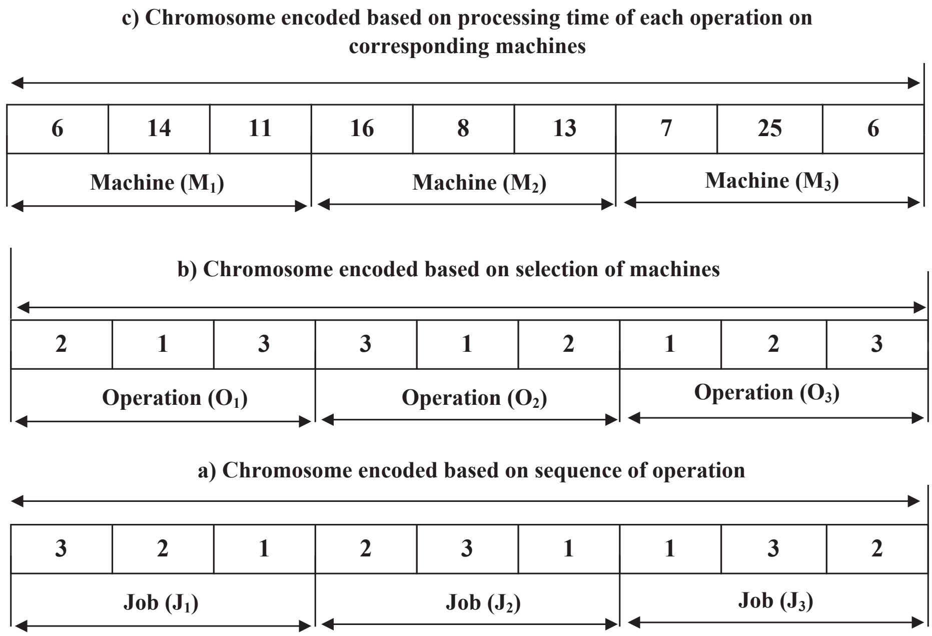


**Fig. 10.** Representation of chromosome initialization for make span.

**Table 7**
Energy and reliability data.

| Machine | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | M9 | M10 | M11 | M12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Energy consumption | 22 | 26 | 36 | 15 | 12 | 15 | 16 | 31 | 12 | 23 | 27 | 18 |

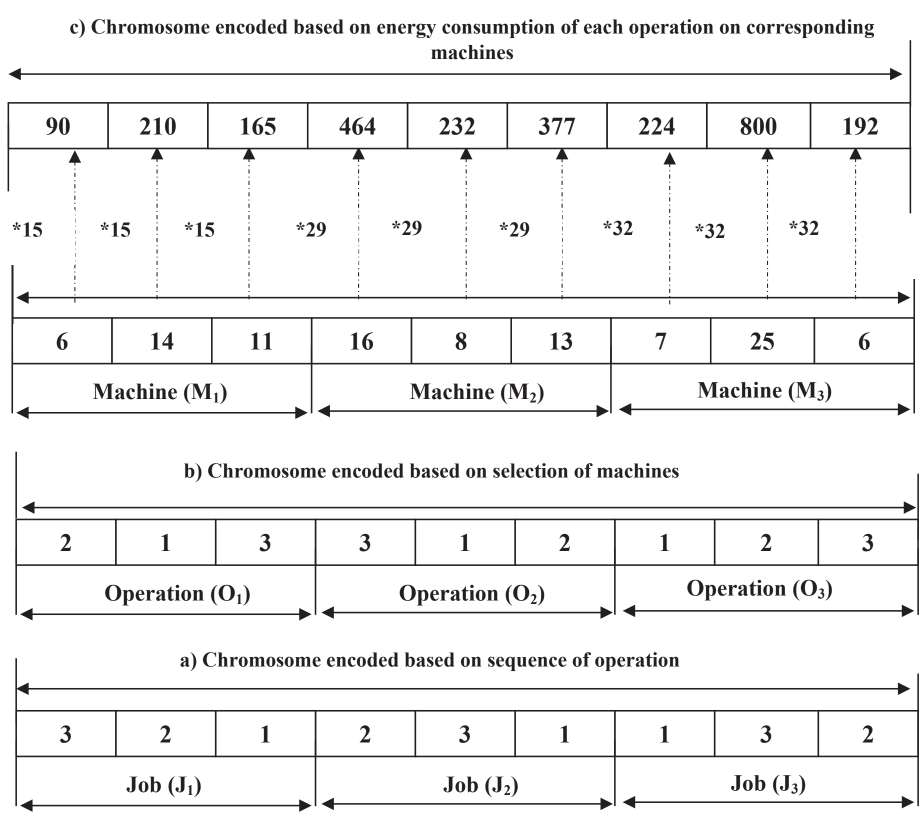


**Fig. 11.** Representation of chromosome initialization for energy consumption.

a is a constant defining shape of spiral motion. Where t [-1,1],

*Step 7* After finding the minimum entry in matrix and converting all ∞'s to 0 s, the sum of all the values is found in their respective objective function matrices.

*Step 9* Lastly to make sure whether each objective function has been optimized or not.

The flowchart for the proposed HMFEO is presented in Fig. 12. The real data collected contains the information regarding the makespan of the jobs, energy consumption, reliability of machines and service utilization rate. All the algorithms that were proposed were coded and executed with the help of MATLAB software and tests are conducted by using Lenovo Idea Pad 5 Laptop with Intel core I 7-11th generation windows 10 professional OS.

## 6. Discussion and results

The efficiency of the HMFEO algorithm is examined in various scenarios of the problem with the purpose of optimizing the objective functions of our problem, namely makespan, energy consumption, machine utilization. Table 7 shows the rated energy consumption for each machine. Out of all the available process plans Different scenarios are considered in this problem and their corresponding jobs and machines.

In order to validate the proposed model, several small-sized instances were solved by the CPLEX solver of GAMS software. A time limitation of 3600 s was taken into account for solving the test scenarios [40] mentioned in Table 8. A comparison of cplex results with the proposed HMFO was shown in Table 8. All the three objectives namely makespan, energy consumption, machine utilization values obtained by an augmented e- constraint method followed in CPLEX solver of GAMS [57] and the results are compared with the objective values of proposed HMFO. The lower makespan and energy consumption and higher energy consumption values of proposed HMFO indicates the better performance of the algorithm over all the test scenarios. Moreover, large size problem scenarios are not able to solve by using exact solution methodologies like CPLEX, taking huge amount of CPU times. Hence in this work to compare the proposed algorithm for large data scenarios a Non Dominated Sorting Genetic algorithm (NSGA -III) is considered.

### 6.1. *Comparison of the considered HMFEO with the experimental scenarios*

To check the feasibility of the proposed HMFEO method a comparison study has been carried out in this work by considering the experimental scenarios 1 to 36 mentioned in Table 9. From Table 9 scenarios 1 to 32 mentioned by Tang et al., [54] used the Genetic Algorithm based Simulated Annealing (GA-SA) were considered. Whereas in scenarios 33 to 35 mentioned by Jin et al., [23] applied the Genetic algorithm based memetic algorithm (GA-MA) for getting optimal values of makespan and Energy consumption (EC) while process planning and scheduling. In this regard, we tested the proposed HMFEO for all the practical scenarios 1 to 36 and their values are shown in Table 9. Furthermore, the superiority of the proposed algorithm for all the experimental instances were identified.

### 6.2. *Comparison of the considered HMFEO with the practical scenarios*

Hereafter thorough analysis of experimental scenarios (Table 9) and, the effectiveness of the proposed HMFEO algorithm. We further considered the Practical scenarios 1 to 10 in Table 10. Each scenario consists of multiple jobs and machines i.e scenario 1 consists of six jobs and six machines scenario 3 consists of six jobs and eight machines etc. shown in Table 10. Moreover, Table 10 shows the optimal process plans selected for every job in various problem scenarios. A formula mentioned in Equation (12) is used to calculate the score value used to

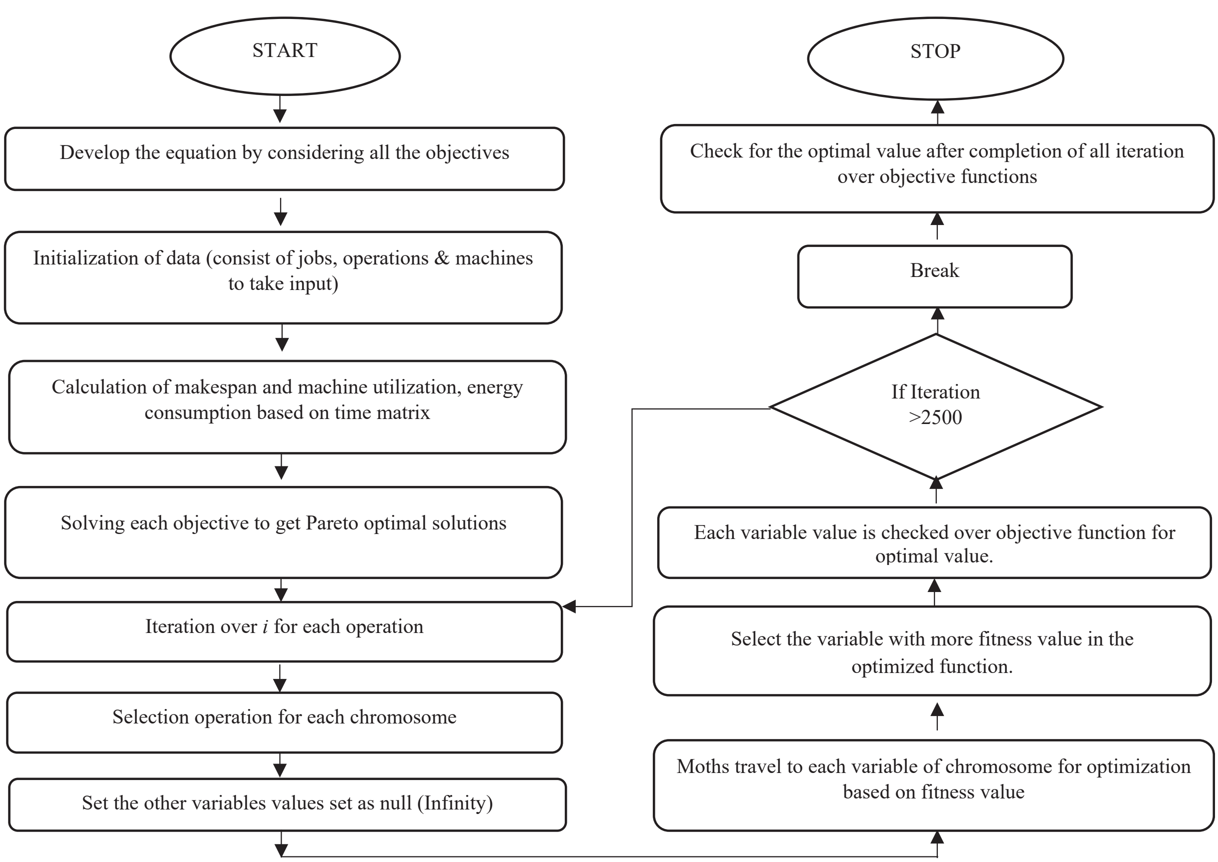


**Fig. 12.** Flowchart of the proposed Hybridized MFEO.

**Table 8**
The comparison of results obtained by cplex solver in GAMS software with Proposed HMFO.

| Test Scenario (Small sized problems) | Jobs*machines | MILP by cplex solver | | | | HMFO | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | Makepsan (Minutes) | EC (KW) | MU | CPU Time (Seconds) | Makepsan | EC | MU | CPU |
| Test Scenario 1 | 2*2 | 32 | 526 | 0.6 | 210 | 31 | 502 | 0.62 | 20 |
| Test Scenario 2 | 2*2 | 29 | 429 | 0.54 | 220 | 26 | 412 | 0.61 | 32 |
| Test Scenario 3 | 3*2 | 36 | 621 | 0.51 | 292 | 32 | 596 | 0.56 | 28 |
| Test Scenario 4 | 3*2 | 45 | 789 | 0.53 | 284 | 42 | 741 | 0.56 | 37 |
| Test Scenario 5 | 3*2 | 49 | 873 | 0.59 | 292 | 41 | 762 | 0.67 | 36 |
| Test Scenario 6 | 3*2 | 44 | 764 | 0.53 | 361 | 39 | 723 | 0.64 | 36 |
| Test Scenario 7 | 3*3 | 49 | 856 | 0.54 | 792 | 42 | 799 | 0.62 | 37 |
| Test Scenario 8 | 3*4 | 54 | 963 | 0.56 | 1260 | 46 | 856 | 0.59 | 34 |
| Test Scenario 9 | 3*5 | 51 | 789 | 0.58 | 1505 | 39 | 693 | 0.63 | 30 |
| Test Scenario 10 | 4*5 | 53 | 823 | 0.51 | 2163 | 42 | 752 | 0.61 | 39 |

**Table 9**
Comparison of make span and energy consumption results for all experimental scenarios.

| | **Jobs** | **Machines** | **GA-SA (Scenario 1 to 32) GA- MA(Scenario 33 to 35)** | | **Proposed HMFEO** | |
|---|---|---|---|---|---|---|
| | | | **Makespan** | **EC** | **Makespan** | **EC** |
| Scenario 1 | Three | Five | 41 | 138.1 | **29** | **119** |
| Scenario 2 | Three | Seven | 54 | 205.4 | **44** | **170** |
| Scenario 3 | Three | Ten | 6 | 229.1 | **50** | **179** |
| Scenario 4 | Three | Five | 190 | 708.7 | **165** | **692** |
| Scenario 5 | Three | Seven | 253 | 960.6 | **224** | **743** |
| Scenario6 | Three | Ten | 334 | 1273.3 | **250** | **1100** |
| Scenario7 | Three | Five | 375 | 1307.1 | **320** | **1100** |
| Scenario8 | Three | Seven | 532 | 1895.4 | **499** | **1698** |
| Scenario9 | Three | Ten | 729 | 2830.5 | **665** | **2563** |
| Scenario10 | Five | Five | 35 | 140.4 | **20** | **101** |
| Scenario11 | Five | Seven | 46 | 187 | **31** | **172** |
| Scenario12 | Five | Ten | 51 | 199.9 | **41** | **166** |
| Scenario13 | Five | Five | 165 | 671.5 | **149.8** | **576** |
| Scenario14 | Five | Seven | 225 | 951.2 | **201.7** | **810** |
| Scenario15 | Five | Ten | 317 | 1303.6 | **302** | **1035** |
| Scenario16 | Five | Five | 325.5 | 1253.2 | **307** | **1101** |
| Scenario17 | Five | Seven | 437 | 1909 | **404** | **1597** |
| Scenario18 | Five | Ten | 610 | 2589 | **568** | **2142** |
| Scenario19 | Seven | Five | 29 | 111 | **16** | **92** |
| Scenario20 | Seven | Seven | 39 | 162.2 | **21** | **122** |
| Scenario21 | Seven | Ten | 56 | 241.6 | **39** | **185** |
| Scenario22 | Seven | Five | 160 | 607 | **123** | **523** |
| Scenario23 | Seven | Seven | 221 | 919.1 | **189** | **819** |
| Scenario24 | Seven | Ten | 305 | 1310.5 | **269** | **1106** |
| Scenario25 | Seven | Five | 351 | 1422.9 | **306** | **1265** |
| Scenario26 | Seven | Seven | 426 | 1978.3 | **356** | **1696** |
| Scenario27 | Seven | Ten | 626 | 2664.1 | **546** | **2214** |
| Scenario28 | Ten | Ten | 940 | 9873.2 | **826** | **8142** |
| Scenario29 | Fifteen | Fifteen | 1554 | 22505.2 | **1397** | **1993** |
| Scenario30 | Twenty | Twenty | 4778 | 80577.2 | **4263** | **72,693** |
| Scenario31 | Twenty | Twenty | 7753 | 100073.4 | **6356** | **85,741** |
| Scenario32 | Twenty | Twenty | 15,062 | 197787.5 | **13,897** | **173,652** |
| Scenario 33 | Eighteen | Fifteen | 531 | 13340.3 | **523** | **12,869** |
| Scenario 34 | Eighteen | Fifteen | 810 | 2036.32 | **719** | **1895** |
| Scenario 35 | Eighteen | Fifteen | 680 | 2267.88 | **582** | **1742** |

select the optimal process plan out of all available process plans for each job. Both the proposed HMFEO and NSGA III algorithms were used to obtain the objective functions simultaneously for all various problem scenarios i.e. (Scenarios 1 to 10) that are tabulated in Table 11 with their Pareto-optimal values of makespan and energy consumption.

From Table 11, it is possible to find out that for scenario 1, six jobs and six machines (6X6) were considered, and the makespan (i.e. maximum completion time of all the jobs) is 51-time units, in the case of the HMFEO algorithm, which is lesser than the makespan of 55-time units, in the case of the NSGA III. Similarly, the energy consumption values for scenario 1 are 5723 and 6358 for HMFEO and NSGA III, respectively. From this, we may conclude that the proposed HMFEO enables lesser energy consumption than NSGA III. For scenario 3, i.e. six jobs and eight machines (6*8), the process parameters makespan and energy consumption values are more compared to six jobs, six machines (6*6), and eight jobs, eight machines (8*8) cases. The main reason for the increase in values in (6*8) the case may be due to the fewer number of jobs that need to be completed by more machines where there is a chance of less utilization of machines. A similar trend also found in the literature [35]. Upon observing and comparison of both proposed HMFEO and NSGA III for all the scenarios i.e (1 to 10) the HMFEO algorithm obtained far better Pareto-optimal results when compared to the standard NSGA III algorithm.

To understand the results obtained by the HMFEO algorithm in a

**Table 10**
Optimal process plans selected for each job for all scenarios 1 to 10.

| Scenario | Different Cases | | Chosen Process plans | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Jobs number | Machines | Job 1 | Job2 | Job3 | Job4 | Job5 | Job6 | Job7 | Job 8 |
| Scenario 1 | Six | six | 2 | 3 | 1 | 3 | 2 | 1 | – | – |
| Scenario 2 | Six | six | 2 | 2 | 2 | 1 | 2 | 3 | – | – |
| Scenario 3 | Six | Eight | 1 | 1 | 3 | 3 | 2 | 1 | – | – |
| Scenario 4 | Eight | Eight | 3 | 2 | 1 | 3 | 1 | 2 | 2 | 1 |
| Scenario 5 | Eight | Eight | 3 | 2 | 3 | 1 | 2 | 1 | 3 | 1 |
| Scenario 6 | Six | Twelve | 2 | 2 | 3 | 3 | 2 | 1 | – | – |
| Scenario 7 | Six | Twelve | 3 | 1 | 1 | 2 | 2 | 1 | – | – |
| Scenario 8 | Six | Twelve | 3 | 1 | 1 | 3 | 3 | 3 | – | – |
| Scenario 9 | Six | Twelve | 2 | 3 | 3 | 1 | 1 | 3 | – | – |
| Scenario10 | Six | Twelve | 3 | 2 | 2 | 1 | 3 | 2 | – | – |

**Table 11**
Results of the practical scenarios with makespan and energy consumption values.

| Scenario | Number of Jobs | Machines | Proposed HMFEO | | NSGA III | |
|---|---|---|---|---|---|---|
| | | | Makespan | Energy consumption | Makespan | Energy consumption |
| Scenario 1 | six | six | 51 | 5722.29 | 55 | 6358 |
| Scenario 2 | six | six | 48 | 5481 | 62 | 6090 |
| Scenario 3 | six | Eight | 160 | 15,654 | 187 | 25,636 |
| Scenario 4 | Eight | Eight | 59 | 8846 | 69 | 9828 |
| Scenario 5 | Eight | Eight | 68 | 7050 | 82 | 7833 |
| Scenario 6 | six | Twelve | 693 | 11,148 | 793 | 12,386 |
| Scenario 7 | six | Twelve | 1023 | 10,575 | 1153 | 11,750 |
| Scenario 8 | six | Twelve | 1398 | 8950 | 1479 | 9944 |
| Scenario 9 | six | Twelve | 2050 | 9387 | 2195 | 10,429 |
| Scenario10 | six | Twelve | 1310 | 9183 | 1560 | 10,203 |

more detailed way Gantt charts for all the ten scenarios were plotted. Gantt charts explain the process of planning and scheduling in a pictorial way. The X-coordinate indicates the processing time for each job and Y coordinates indicating the corresponding machine. For a better visibility, Gantt charts for scenarios 1 to 5 are shown in Fig. 13. All remaining scenarios, i.e. from scenarios 6 to 10, are represented as Gantt charts in Fig. 14. As specified in Fig. 13, the makespan for various problem cases, from scenario1 to 5, is 51, 48, 160, 59, and 68 respectively. From Fig. 14 makespan for scenarios 6 to 10, it is 693, 1023, 1398, 2050, and 1310 respectively. In Fig. 13 and Fig. 14 the makespan values indicate that the solution depends not only on the number of machines considered but also on the various process plans that are selected for manufacturing a product.

Apart from the comparison of both proposed HMFEO and NSGA III algorithms for makespan and energy consumption Pareto optimal solutions for all scenarios, furthermore, a comparison of machine utilization for all the scenarios has been done in this work. The machine utilization values for Scenarios 1 to 10 are plotted in Fig. 15. For various

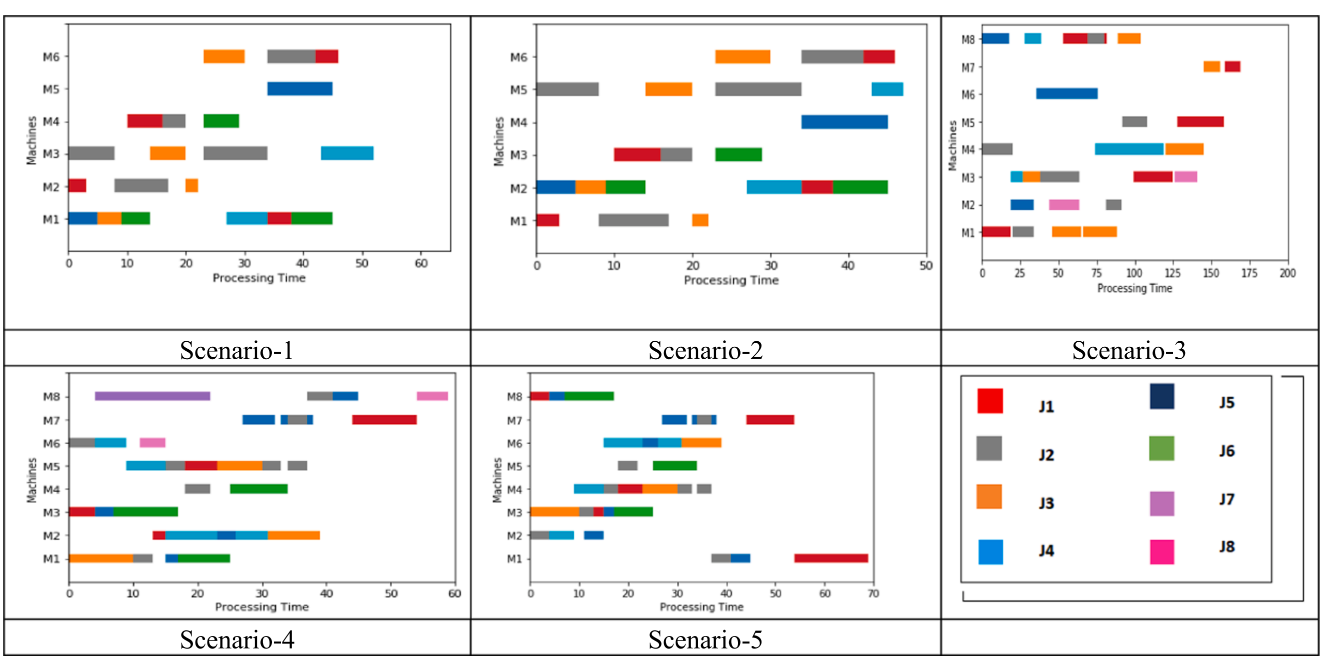


**Fig. 13.** Gantt charts for the scenarios 1 to 5 for the proposed HMFEO.

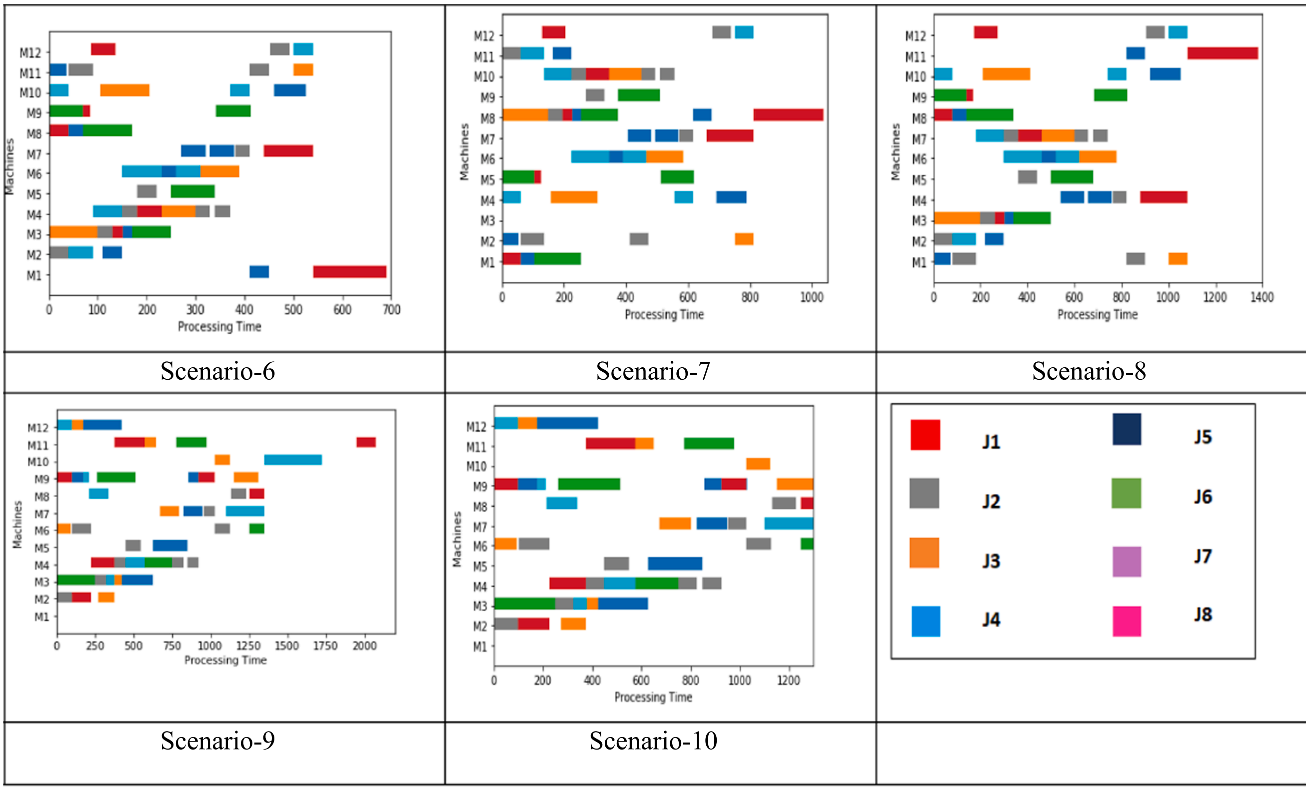


**Fig. 14.** Gantt charts for the scenarios 6 to 10 for the proposed HMFEO.

scenarios, the machine utilization for machines is different i.e., from Fig. 15 for scenario 1 the Machines M1, M2 are better utilized whereas machine M5 is the least utilized. Moreover, for scenario 2, Machines M2 is better utilized whereas other machines are utilized equally. For scenarios 3 and 4, the machine 7 utilization is very low, in fact, it is possible to say that almost not utilized. For scenario 5, the machine M2 has less utilization capacity as shown in the figure. Through Fig. 15., and comparing all the scenarios 1 to 10, it is possible to realize that the proposed HMFEO gives comparatively gives better Pareto- optimal results when compared to the NSGA III algorithm.

The Energy Consumption data is shown with a bar chart shown in Fig. 16 for all scenarios 1 to 10. From this data, it is possible to infer that the energy consumption for scenario 3 (i.e. six jobs and eight machines case) is more when compared to the other scenarios. This higher value indicating that the process plan that is selected for scenario 3 is maybe containing more energy-consuming operations with the available machines. For both proposed HMFEO and NSGA III algorithms the pattern is similar. But the proposed HMFEO gives lower values of energy consumption indicating better performance over the NSGA III algorithm.

### *6.3. Various performance indicators for validity of proposed HMFEO with NSGA-III*

Several performance indicators (PI) were suggested by [11,70,28,6] and these indicators compares the performance of the multi/ many objective algorithms. Mostly used Performance Measure out of all is the hyper volume [14]. Hyper volume (HV) is the volume surrounded by the dominated Pareto front approximation ‘K’ from a reference point × £ $X^P$, such that b £ K, K < x. The HV is given by Equation (14). Here, $\eta_P$ represents P dimensional lebesgue measure.

$$HV(K,\ x) = \eta_P(\bigcup_{B\in K}[B,x] \tag{14}$$

The HV values for all ten scenarios are plotted with the help of a box plate for the proposed HMFEO and NSGA III algorithms. For scenario 1 the highest median and worst values are 0.6873, 0.6573, 0.5823 respectively or NASGA III. In the same way, box plots were presented for all the ten scenarios shown in Fig. 17. The higher the HV indicates, the better is the performance. Fig. 17 depicts that the proposed HMFEO is better than the NSGA III algorithm. From the Fig. 17 concluded that there is an average increase of nearly 21 percentage of hyper volume is observed for all the scenarios. Hence, indicates the superiority in all sects of parameters in the objective space of the proposed algorithm HMFO over the NSGA III algorithm.

Along with the Hyper volume calculation, in this work several other performance measures were calculated according to problem context for all the ten instances. Performance measures (PM) for all the scenarios 1 to 5 are indicated in Table 12 and for scenarios 6 to 10 the PM values are represented in Table 13. The ratio of Non-Dominated (ND) solutions identified by the both the algorithms considered as (χ /ψ) one of the performance measure to compare the effectiveness of the algorithms. Where χ is the number of ND solutions identified by the proposed HMFO algorithm and ψ denotes the number of ND solutions produced by the bench mark algorithm. Another useful performance measure mentioned here is the dominance ratio ω in Equation (15). The more the value of ω indicates better performance of the algorithm.

$$\omega = \frac{\left|C\left(\underset{j}{\cup}N_j\right)\backslash C\left(\underset{j\neq l}{\cup}N_j\right)\right|}{C\left(\underset{j}{\cup}N_j\right)} \tag{15}$$

where $\left|C\left(\underset{j}{\cup}N_j\right)\backslash C\left(\underset{j\neq l}{\cup}N_j\right)\right|$ is the ND solutions identified by the algorithm N that are not found by the other algorithms.

κ (p,q) in Equation (16) comparison of pareto fronts which helpful to identify the week solution produced by one algorithm q over the other algorithm p. (q > p), helps to identify the correctness of the algorithm. The lower value indicates the smaller number of weak solutions identified by that algorithm.


**Fig. 15.** Comparison of Machine Utilization for the scenarios 1 to 10 for the HMFEO and NSGA III.

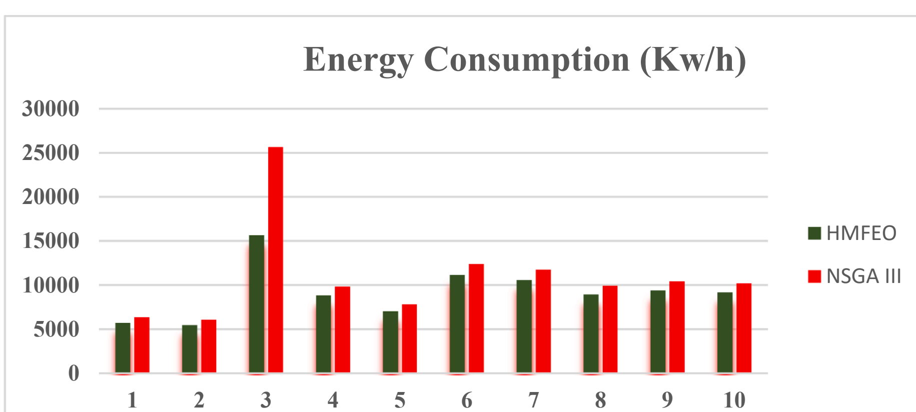


**Fig. 16.** Energy consumption for all the scenario for the proposed HMFEO and NSGA III.

$$\kappa(p,q) = \frac{|q \in Q, \exists p \in P : p > q|}{|Q|} \tag{16}$$

Lesser λ is necessary, and the values which are very nearer to zero indicate the highly distributed uniformly over the Pareto front. Equation (17) values of π which is the Euclidean length between end points of the identified ND Pareto set by an algorithm is compared to the net ND Pareto front.

The uniform distribution of solutions over the Pareto front is given by the diversity ($\partial$) measure in Equation (18) where the extreme solutions in the ND Pareto set are represented by solution Gf and $G_f$;the number of solutions identified is denoted by J; G is the Euclidian length

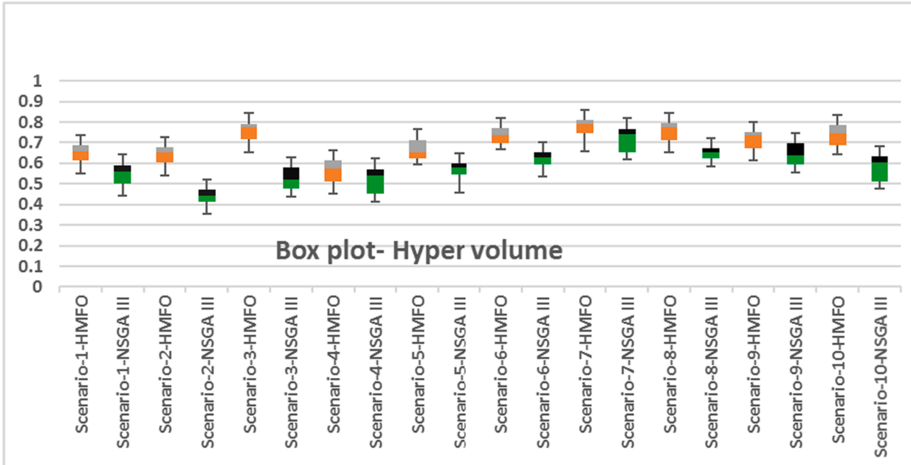


**Fig. 17.** Box plot indicating Hyper Volume values for all the ten scenarios of HMFEO and NSGA-III.

**Table 12**
Performance indicators for all Scenario 1 to 5 for both HMFEO and NSGA -III.

| Indicator | Algorithm | Scenario | | | | |
|---|---|---|---|---|---|---|
| | | I | II | III | IV | V |
| χ | HMFEO | **9.0** | **11.7** | **9.7** | **10.0** | **9.1** |
| | NSGA III | 8.1 | 10.5 | 8.9 | 10.0 | 8.4 |
| ψ | HMFEO | **8.7** | **10.6** | **9.5** | **10.0** | **8.5** |
| | NSGA III | 7.5 | 9.5 | 8.1 | 9.0 | 8.1 |
| χ /ψ | HMFEO | **0.9666** | **0.9900** | **0.9793** | **1.0000** | **0.9340** |
| | NSGA III | 0.8750 | 0.9047 | 0.9101 | 0.9 | 0.9642 |
| ω | HMFEO | **0.5485** | **0.5124** | **0.6245** | **0.5245** | **0.5652** |
| | NSGA III | 0.4375 | 0.4987 | 0.3985 | 0.5196 | 0.4841 |
| κ | HMFEO | **0.0700** | **0.0156** | **0.0320** | **0.0012** | **0.0300** |
| | NSGA III | 0.1300 | 0.0894 | 0.0116 | 0.1333 | 0.0412 |
| ∂ | HMFEO | **0.4256** | **0.0042** | **0.7378** | **0.0023** | **0.0068** |
| | NSGA III | 11.321 | 8.666 | 7.345 | 9.5321 | 9.6631 |
| £ | HMFEO | 0.3176 | **0.4963** | **0.4814** | **0.4785** | **0.4258** |
| | NSGA III | **0.4569** | 0.5666 | 0.6325 | 0.6841 | 0.6124 |
| CPUTIME (s) | HMFEO | **137.8** | **140.3** | **142** | **139.1** | **141.7** |
| | NSGA III | 246 | 216 | 263.4 | 243.6 | 256.3 |

**Table 13**
Performance indicators for all Scenario 6 to 10 for both HMFEO and NSGA -III.

| Indicator | Algorithm | Scenario | | | | |
|---|---|---|---|---|---|---|
| | | VI | VII | VIII | IX | X |
| χ | HMFEO | **8.9** | **11.6** | **8.6** | **10.0** | **8.1** |
| | NSGA III | 8.1 | 10.5 | 8.8 | 10.0 | 8.5 |
| ψ | HMFEO | **8.6** | **10.6** | **8.5** | **10.0** | **8.5** |
| | NSGA III | 6.5 | 8.5 | 8.1 | 8.0 | 8.1 |
| χ /ψ | HMFEO | **0.8666** | **0.8800** | **0.8684** | **1.0000** | **0.8450** |
| | NSGA III | 0.8650 | 0.8056 | 0.8101 | 0.8 | 0.8651 |
| ω | HMFEO | **0.5585** | **0.5115** | **0.6155** | **0.5155** | **0.5651** |
| | NSGA III | 0.5465 | 0.5886 | 0.4885 | 0.5186 | 0.5851 |
| κ | HMFEO | **0.0600** | **0.0156** | **0.0410** | **0.0011** | **0.0400** |
| | NSGA III | 0.1400 | 0.0885 | 0.0116 | 0.1444 | 0.0511 |
| ∂ | HMFEO | **0.5156** | **0.0051** | **0.6468** | **0.0014** | **0.0068** |
| | NSGA III | 11.411 | 8.666 | 6.455 | 8.5411 | 8.6641 |
| £ | HMFEO | 0.4166 | **0.5864** | **0.5815** | **0.5685** | **0.5158** |
| | NSGA III | **0.5568** | 0.5666 | 0.6415 | 0.6851 | 0.6115 |
| CPUTIME (s) | HMFEO | **146** | **144.3** | **146** | **148.1** | **149.7** |
| | NSGA III | **236** | **245** | **274** | **263** | **241** |

between two consecutive points average Euclidian length $\overline{G}$ over all the available non-Dominated solutions.

$$\partial = \frac{G_f + G_l + \sum_{i=1}^{J-1}|G_i - \overline{G}|}{G_f + G_l + (J-1)\overline{G}} \quad (17)$$

£ indicates convergence power if smaller £ values are useful for identified ND solutions by the algorithm and fall very close range in the vicinity of net ND solutions for Euclidean lengths. The CPU time is one of the other performance indicator is mentioned in the Table 12 and Table 13. The CPU time for the proposed algorithm is less than the NSGA III indicating the superiority of the algorithm.

## 7. Managerial and academic implications

With the advent of key enabling technologies, the present manufacturing scenario changes from an enterprise-driven system to a customer-driven system. The manufacturing firms must rethink the existing strategies and its high time to adopt emerging technologies to withstand huge competition in the market. In this scenario, several manufacturing firms located at various places come together forming DMS that helps to gain competitive advantages. The main problem in the DMS is that the manufacturing firms must blindly trust each other to carry out their operations. This kind of scenario limits the further exploration of the DMS system in the highly competitive customer-driven market. Hence enterprises looking for high technology that helps to overcome the trust issue. In this regard, BCT contains several advantages of high security and transparency that help the DMS to share their resource information without blindly trusting each other. In the past work, people have suggested several frameworks on Blockchain-based resources in the DMS. Very little literature focuses on the implementation of smart contracts in supply chain management to track and trace their products.

In this work first, the public permission-less Ethereum blockchain is implemented to share the resources and also to identify capable enterprises. Later, the block chain information is used as an input for the considered Distributed Gear Manufacturing case study for the further process planning and scheduling problem and the nature of the problem is NP-hard. The main focus is to optimize the problem that improves the sustainability of the DMS and that is solved by a proposed HMFEO solution algorithm.

The proposed work is blockchain-based sustainable DMS to encourage the adoption of BCT into their firms. Even though several companies started using BCT in their supply chains. Authors feel that it is high time to adopt this BCT into their shop floors to solve the security issues and simultaneously improves the sustainability of the system.

## 8. Conclusion

In this work, a distributed manufacturing system has been considered where sharing of resources in a secure and transparent manner is of the highest priority. Recent transformation in industries across the world demands advanced technologies to achieve the mentioned issue. In this work, a blockchain-based smart contract has been developed for sharing of resources within the distributed manufacturing system. In addition, apart from sharing the information securely and transparently, the developed Ethereum based smart contract is helpful to identify the capable enterprises in considered DMS that can fulfil the customer requirement. The critical functions in DMS in fact any kind of manufacturing lies in effective and efficient process plans and schedules. Moreover, the considered DMS environment was having challenges like multiple process plans and multiple performance measures that need to be investigated and evaluate in real-time.

Hence, this research paper also investigated alternative process plans for the objective functions makespan, energy consumption, service utilization, and reliability of services. A MILP model was developed, and by acknowledging the NP-hard nature of the above scenario, a multi-objective evolutionary algorithm was decided to be utilized. As a result, was used a Bio-inspired HMFEO and tuned the algorithm to fit the intended problem objectives. The results demonstrate that the use of HMFEO falls superior when compared to NSGA-III, proving the effectiveness of the methodology used in this research. It also provides similar results concerning the survivability of jobs as compared to NSGA-III. Out of all the considered objective functions, energy consumption is of utmost importance because of its effect on the current manufacturing environment. An experimental comparison also reveals the effectiveness of the proposed HMFEO. Thus, the results obtained showcase the effectiveness of the approach mentioned in this research. Finally, future work requires adopting a hybrid blockchain-based smart contract by combining both permissioned and permission-less blockchain smart contracts and application of the methodology on a wider dataset using various other evolutionary algorithms. There may be a requirement of investigation of some more interdependent objectives like service utilization, an optimal sequence of jobs, and the number of generations is significant enough for comparing the performance with different algorithms.

## Funding

This work has been funded by Department of Science and Technology, Science & En-gineering Research Board (DST-SERB), Statutory Body Established through an Act of Parliament: SERB Act 2008, Government of India with Sanction Order No ECR/2016/001808; and by Fundação para a Ciência e Tecnologia (FCT) within the RD Units Project Scope: UIDP/04077/2020 and UIDB/04077/2020 and UIDB/00319/2020.

## Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgement

None.

## References

[1] N. Abdullah, A. Hakansson, E. Moradian, Blockchain based approach to enhance big data authentication in distributed environment. In 2017 Ninth International Conference on Ubiquitous and Future Networks (ICUFN) (pp. 887-892), 2017, IEEE. DOI: 10.1109/ICUFN.2017.7993927.

[2] S.A. Abeyratne, R.P. Monfared, Blockchain ready manufacturing supply chain using distributed ledger, Int. J. Res. Eng. Technol. 5 (9) (2016) 1–10.

[3] S. Afshin Mansouri, E. Aktas, Minimizing energy consumption and makespan in a two-machine flowshop scheduling problem, J. Operational Res. Soc. 67 (11) (2016) 1382–1394, https://doi.org/10.1057/jors.2016.4.

[4] N. Alkaabi, K. Salah, R. Jayaraman, J. Arshad, M. Omar, Blockchain-based traceability and management for additive manufacturing. IEEE access, 8 (2020) 188363-188377. Digital Object Identifier 10.1109/ACCESS.2020.3031536.

[5] A. Angrish, B. Craver, M. Hasan, B. Starly, A Case Study for Blockchain in Manufacturing: "FabRec": A Prototype for Peer-to-Peer Network of Manufacturing Nodes, Procedia Manuf. 26 (2018) 1180–1192.

[6] C. Audet, J. Bigeon, D. Cartier, S. Le Digabel, L. Salomon, Performance indicators in multiobjective optimization, Eur. J. Oper. Res. 292 (2) (2021) 397–422.

[7] M. Bertolini, D. Mezzogori, F. Zammori, Comparison of new metaheuristics, for the solution of an integrated jobs-maintenance scheduling problem, Expert Syst. Appl. 122 (2019) 118–136, https://doi.org/10.1016/j.eswa.2018.12.034.

[8] A. Bogner, M. Chanson, A. Meeuw, A decentralised sharing app running a smart contract on the ethereum blockchain. In Proceedings of the 6th International Conference on the Internet of Things, 2016, November, (pp. 177-178). DOI: https://doi.org/10.1145/2991561.2998465.

[9] B.K. Mohanta, D. Jena, U. Satapathy, Trust management in IoT enable healthcare system using Ethereum based smart contract, Int. J. Sci. Technol. Res. 8 (9) (2019) 758–763.

[10] J. Campos, P. Sharma, E. Jantunen, D. Baglee, L. Fumagalli, The challenges of cybersecurity frameworks to protect data required for the development of advanced maintenance, Procedia Cirp 47 (2016) 222–227, https://doi.org/10.1016/j.procir.2016.03.059.

[11] C.Y. Cheng, S.F. Li, K.C. Ying, Y.H. Liu, Scheduling jobs of two competing agents on a single machine, IEEE Access 7 (2019) 98702–98714, https://doi.org/10.1109/ACCESS.2019.2929582.

[12] C.-Y. Cheng, S.-W. Lin, P. Pourhejazy, K.-C. Ying, S.-F. Li, Y.-C. Liu, Greedy-Based Non-Dominated Sorting Genetic Algorithm III for Optimizing Single-Machine Scheduling Problem With Interfering Jobs, IEEE Access 8 (2020) 142543–142556.

[13] K. Chung, H. Yoo, D. Choe, H. Jung, Blockchain network based topic mining process for cognitive manufacturing, Wireless Pers. Commun. 105 (2) (2019) 583–597, https://doi.org/10.1007/s11277-018-5979-8.

[14] J. Deng, Q. Zhang, Approximating hypervolume and hypervolume contributions using polar coordinate, IEEE Trans. Evol. Comput. 23 (5) (2019) 913–918, https://doi.org/10.1109/TEVC.2019.2895108.

[15] R. Burke, A. Mussomeli, S. Laaper, M. Hartigan, B. Sniderman, The smart factory: Responsive, adaptive, connected manufacturing, Deloitte Insights 31 (1) (2017) 1–10. https://www2.deloitte.com/content/dam/insights/us/articles/4051_The-smart-factory/DUP_The-smart-factory.pdf.

[16] R. Fornasiero, E. Carpanzano, Advances in customer-oriented manufacturing and value chain management, Advances in customer-oriented manufacturing and value chain management 30 (7) (2017) 677–679.

[17] J.A. Garcia-Garcia, N. Sanchez-Gomez, D. Lizcano, M.J. Escalona, T. Wojdynski, Using Blockchain to Improve Collaborative Business Process Management: Systematic Literature Review, IEEE Access 8 (2020) 142312–142336.

[18] A. Giret, D. Trentesaux, V. Prabhu, Sustainability in manufacturing operations scheduling: A state of the art review, J. Manuf. Syst. 37 (2015) 126–140.

[19] S. Guo, M. Dong, Order matching mechanism of the production intermediation internet platform between retailers and manufacturers, The Int. J. Adv. Manuf. Technol. 115 (3) (2021) 949–962.

[20] H.R. Hasan, K. Salah, R. Jayaraman, R.W. Ahmad, I. Yaqoob, M. Omar, Blockchain-based solution for the traceability of spare parts in manufacturing, IEEE Access 8 (2020) 100308–100322, https://doi.org/10.1109/access.2020.2998159.

[21] A. Kumar, K. Abhisek, B. Bhusan, C. Chakraborty, Secure access control for manufacturing sector with application of ethereum blockchain. Peer-to-Peer Networking and Applications, Peer-to-Peer Networking and Applications 14 (5) (2021) 3058–3074, https://doi.org/10.1007/s12083-021-01108-3.

[22] Y. Hou, Y. Chen, Y. Jiao, J. Zhao, H. Ouyang, P. Zhu, ..., Y. Liu, A resolution of sharing private charging piles based on smart contract. In 2017 13Th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (Icnc-Fskd) (pp. 3004-3008). IEEE, 2017, July, DOI: 10.1109/FSKD.2017.8393262.

[23] L. Jin, C. Zhang, X. Fei, Realizing energy savings in integrated process planning and scheduling, Processes 7 (3) (2019) 120, https://doi.org/10.3390/pr7030120.

[24] H. Johng, D. Kim, T. Hill, L. Chung, Using blockchain to enhance the trustworthiness of business processes: a goal-oriented approach. In 2018 IEEE international conference on services computing (SCC) (pp. 249-252). IEEE, 2018, July, DOI: 10.1145/3284764.3284768.

[25] S.S. Kamble, A. Gunasekaran, N. Subramanian, A. Ghadge, A. Belhadi, M. Venkatesh, Blockchain technology's impact on supply chain integration and sustainable supply chain performance: Evidence from the automotive industry, Ann. Oper. Res. 1–26 (2021), https://doi.org/10.1007/s10479-021-04129-6.

[26] J. Leng, G. Ruan, P. Jiang, K. Xu, Q. Liu, X. Zhou, C. Liu, Blockchain-empowered sustainable manufacturing and product lifecycle management in industry 4.0: A survey, Renew. Sustain. Energy Rev. 132 (2020), 110112, https://doi.org/10.1016/j.rser.2020.110112.

[27] Z. Li, A.V. Barenji, G.Q. Huang, Toward a blockchain cloud manufacturing system as a peer to peer distributed network platform, Rob. Comput. Integr. Manuf. 54 (2018) 133–144, https://doi.org/10.1016/j.rcim.2018.05.011.

[28] S.W. Lin, K.C. Ying, A multi-point simulated annealing heuristic for solving multiple objective unrelated parallel machine scheduling problems, Int. J. Prod. Res. 53 (4) (2015) 1065–1076, https://doi.org/10.1080/00207543.2014.942011.

[29] K. Deb, H. JAIN, An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints. IEEE transactions on evolutionary computation 18 (4) (2013) 577–601, https://doi.org/10.1109/TEVC.2013.2281535. In press.

[30] X.L. Liu, W.M. Wang, H. Guo, A.V. Barenji, Z. Li, G.Q. Huang, Industrial blockchain based framework for product lifecycle management in industry 4.0. Robotics and computer-integrated manufacturing 63 (2020), 101897, https://doi.org/10.1016/j.rcim.2019.101897.
[31] X. Liu, Z. Ni, X. Qiu, Application of ant colony optimization algorithm in integrated process planning and scheduling, Int. J. Adv. Manuf. Technol. 84 (1–4) (2016) 393–404, https://doi.org/10.1007/s00170-015-8145-4.
[32] J. Lohmer, Applicability of blockchain technology in scheduling resources within distributed manufacturing. In Logistics Management (pp. 89-103). Springer, Cham, 2019, https://doi.org/10.1007/978-3-030-29821-0_7.
[33] J. Lohmer, R. Lasch, Production planning and scheduling in multi-factory production networks: a systematic literature review, Int. J. Prod. Res. 59 (7) (2021) 2028–2054, https://doi.org/10.1080/00207543.2020.1797207.
[34] Y. Lu, The blockchain: State-of-the-art and research challenges, J. Ind. Inform. Integration 15 (2019) 80–90, https://doi.org/10.1016/j.jii.2019.04.002.
[35] V.K. Manupati, S. Deo, N. Cheikhrouhou, M.K. Tiwari, Optimal process plan selection in networked based manufacturing using game-theoretic approach, Int. J. Prod. Res. 50 (18) (2012) 5239–5258.
[36] V.K. Manupati, G.D. Putnik, M.K. Tiwari, P. Ávila, M.M. Cruz-Cunha, Integration of process planning and scheduling using mobile-agent based approach in a networked manufacturing environment, Comput. Ind. Eng. 94 (2016) 63–73, https://doi.org/10.1016/j.cie.2016.01.017.
[37] G.C. Menezes, G.R. Mateus, M.G. Ravetti, A branch and price algorithm to solve the integrated production planning and scheduling in bulk ports, Eur. J. Oper. Res. 258 (3) (2017) 926–937, https://doi.org/10.1016/j.ejor.2016.08.073.
[38] S. Mirjalili, Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm, Knowl.-Based Syst. 89 (2015) 228–249, https://doi.org/10.1016/j.knosys.2015.07.006.
[39] A.E.C. Mondragon, C.E.C. Mondragon, E.S. Coronado, Exploring the applicability of blockchain technology to enhance manufacturing supply chains in the composite materials industry. In 2018 IEEE International conference on applied system invention (ICASI) (pp. 1300-1303). IEEE, 2018, April, DOI: 10.1109/ICASI.2018.8394531.
[40] C. Özgüven, L. Özbakır, Y. Yavuz, Mathematical models for job-shop scheduling problems with routing and process plan flexibility, Appl. Math. Model. 34 (6) (2010) 1539–1548, https://doi.org/10.1016/j.apm.2009.09.002.
[41] S. Panja, S. Bag, F. Hao, B. Roy, A smart contract system for decentralized borda count voting, IEEE Trans. Eng. Manage. 67 (4) (2020) 1323–1339, https://doi.org/10.1109/TEM.2020.2986371.
[42] M. Petrović, N. Vuković, M. Mitić, Z. Miljković, Integration of process planning and scheduling using chaotic particle swarm optimization algorithm, Expert Syst. Appl. 64 (2016) 569–588, https://doi.org/10.1016/j.eswa.2016.08.019.
[43] E. Rauch, P. Dallasega, D.T. Matt, Sustainable production in emerging markets through Distributed Manufacturing Systems (DMS), J. Cleaner Prod. 135 (2016) 127–138, https://doi.org/10.1016/j.jclepro.2016.06.106.
[44] E. Rauch, M. Dallinger, P. Dallasega, D.T. Matt, Sustainability in manufacturing through distributed manufacturing systems (DMS), Procedia CIRP 29 (2015) 544–549, https://doi.org/10.1016/j.procir.2015.01.069.
[45] M.S. Reddy, C.H. Ratnam, R. Agrawal, M.L.R. Varela, I. Sharma, V.K. Manupati, Investigation of reconfiguration effect on makespan with social network method for flexible job shop scheduling problem, Comput. Ind. Eng. 110 (2017) 231–241, https://doi.org/10.1016/j.cie.2017.06.014.
[46] M. Risius, K. Spohrer, A blockchain research framework, Business & Inform. Syst. Eng. 59 (6) (2017) 385–409, https://doi.org/10.1007/s12599-017-0506-0.
[47] S. Saberi, M. Kouhizadeh, J. Sarkis, L. Shen, Blockchain technology and its relationships to sustainable supply chain management, Int. J. Prod. Res. 57 (7) (2019) 2117–2135, https://doi.org/10.1080/00207543.2018.1533261.
[48] C. Schinckus, The good, the bad and the ugly: An overview of the sustainability of blockchain technology, Energy Res. Social Sci. 69 (2020), 101614, https://doi.org/10.1016/j.erss.2020.101614.
[49] X. Shao, X. Li, L. Gao, C. Zhang, Integration of process planning and scheduling—a modified genetic algorithm-based approach, Comput. Oper. Res. 36 (6) (2009) 2082–2096, https://doi.org/10.1016/j.cor.2008.07.006.
[50] P.K. Sharma, N. Kumar, J.H. Park, Blockchain-based distributed framework for automotive industry in a smart city, IEEE Trans. Ind. Inf. 15 (7) (2018) 4197–4205, https://doi.org/10.1109/TII.2018.2887101.
[51] V. Shermin, Disrupting governance with blockchains and smart contracts, Strategic Change 26 (5) (2017) 499–509, https://doi.org/10.1002/jsc.2150.
[52] L. Silvestri, A. Forcina, V. Introna, A. Santolamazza, V. Cesarotti, Maintenance transformation through Industry 4.0 technologies: a systematic literature review, Comput. Ind. 123 (2020), 103335, https://doi.org/10.1016/j.compind.2020.103335.
[53] S. Supranee, S. Rotchanakitumnuai, The acceptance of the application of blockchain technology in the supply chain process of the Thai automotive industry. In Proceedings of the International Conference on Electronic Business (ICEB) (Vol. 2017, pp. 252-257), 2017. https://aisel.aisnet.org/iceb2017/30.
[54] D. Tang, M. Dai, Energy-efficient approach to minimizing the energy consumption in an extended job-shop scheduling problem, Chin. J. Mech. Eng. 28 (5) (2015) 1048–1055, https://doi.org/10.3901/CJME.2015.0617.082.
[55] R. Tavakkoli-Moghaddam, M. Ranjbar-Bourani, G.R. Amin, A. Siadat, A cell formation problem considering machine utilization and alternative process routes by scatter search, J. Intell. Manuf. 23 (4) (2012) 1127–1139, https://doi.org/10.1007/s10845-010-0395-2.
[56] S. Shirzadi, R. Tavakkoli-Moghaddam, R. Kia, M. Mohammadi, A multi-objective imperialist competitive algorithm for integrating intra-cell layout and processing route reliability in a cellular manufacturing system, International Journal of Computer Integrated Manufacturing 30 (8) (2017) 839–855.
[57] F. Torabi Yeganeh, S.H. Zegordi, A multi-objective optimization approach to project scheduling with resiliency criteria under uncertain activity duration, Ann. Oper. Res. 285 (1) (2020) 161–196, https://doi.org/10.1007/s10479-019-03375-z.
[58] A. Tursi, H. Panetto, G. Morel, M. Dassisti, Ontology-based products information interoperability in networked manufacturing enterprises, IFAC Proceedings Volumes 40 (19) (2007) 85–93, https://doi.org/10.3182/20071002-MX-4-3906.00015.
[59] C. Udokwu, A. Kormiltsyn, K. Thangalimodzi, A. Norta, An exploration of blockchain enabled smart-contracts application in the enterprise, Technical Report (2018) 1–29.
[60] L. Varela, A. Araújo, P. Ávila, H. Castro, G. Putnik, Evaluation of the relation between lean manufacturing, Industry 4.0, and sustainability, Sustainability 11 (5) (2019) 1439, https://doi.org/10.3390/su11051439.
[61] W. Viriyasitavat, L.i. Da Xu, Z. Bi, A. Sapsomboon, Blockchain-based business process management (BPM) framework for service composition in industry 4.0, J. Intell. Manuf. 31 (7) (2020) 1737–1748.
[62] H. Wang, C. Guo, S. Cheng, LoC—A new financial loan management system based on smart contracts, Future Generation Comput. Syst. 100 (2019) 648–655, https://doi.org/10.1016/j.future.2019.05.040.
[63] Z. Wang, H. Jin, W. Dai, K.K.R. Choo, D. Zou, Ethereum smart contract security research: survey and future research opportunities, Front. Comput. Sci., 15(2) (2021) 1-18. DOI https://doi.org/10.1007/s11704-020-9284-9.
[64] M. Westerkamp, F. Victor, A. Küpper, Tracing manufacturing processes using blockchain-based token compositions, Digital Communications and Networks 6 (2) (2020) 167–176, https://doi.org/10.1016/j.dcan.2019.01.007.
[65] Y. Wu, Y. Zhang, An integrated framework for blockchain-enabled supply chain trust management towards smart manufacturing, Adv. Eng. Inf. 51 (2022), 101522, https://doi.org/10.1016/j.aei.2021.101522.
[66] Z. Xu, Y. Liu, J. Zhang, Z. Song, J. Li, J. Zhou, Manufacturing industry supply chain management based on the ethereum blockchain. In 2019 IEEE International Conferences on Ubiquitous Computing & Communications (IUCC) and Data Science and Computational Intelligence (DSCI) and Smart Computing, Networking and Services (SmartCNS) (pp. 592-596). IEEE, 2019, October. DOI Bookmark: 10.1109/IUCC/DSCI/SmartCNS.2019.00124.
[67] J. Yan, L. Li, F. Zhao, F. Zhang, Q. Zhao, A multi-level optimization approach for energy-efficient flexible flow shop scheduling, J. Cleaner Prod. 137 (2016) 1543–1552, https://doi.org/10.1016/j.jclepro.2016.06.161.
[68] R. Yang, F.R. Yu, P. Si, Z. Yang, Y. Zhang, Integrated blockchain and edge computing systems: A survey, some research issues and challenges. IEEE Communications Surveys & Tutorials, 21(2) (2019) 1508-1532. doi: 10.1109/COMST. 2019.2894727.
[69] K.-H. Yeh, C. Su, J.-L. Hou, W. Chiu, C.-M. Chen, A Robust Mobile Payment Scheme With Smart Contract-Based Transaction Repository, IEEE Access 6 (2018) 59394–59404.
[70] K.C. Ying, S.W. Lin, S.Y. Wan, Bi-objective reentrant hybrid flowshop scheduling: an iterated Pareto greedy algorithm, Int. J. Prod. Res. 52 (19) (2014) 5735–5747, https://doi.org/10.1080/00207543.2014.910627.
[71] M.A. Zafar, F. Sher, M.U. Janjua, S. Baset, Sol2js: translating solidity contracts into javascript for hyperledger fabric. In Proceedings of the 2nd Workshop on Scalable and Resilient Infrastructures for Distributed Ledgers (pp. 19-24), 2018, December, DOI: 10.1145/3284764.3284768.
[72] Z. Zheng, S. Xie, H.N. Dai, W. Chen, X. Chen, J. Weng, M. Imran, An overview on smart contracts: Challenges, advances and platforms, Future Generation Comput. Syst. 105 (2020) 475–491, https://doi.org/10.1016/j.future.2019.12.019.
[73] Y. Zuo, Y. Wang, Y. Laili, T.W. Liao, F. Tao, An evolutionary algorithm recommendation method with a case study in flow shop scheduling, Int. J. Adv. Manuf. Technol. 109 (3) (2020) 781–796, https://doi.org/10.1007/s00170-020-05471-y.